Automatic Orthographic Alignment of Speech

by

Jerome S. Khohayting
S.B. Mathematics, Massachusetts Institute of Technology, 1993

Submitted to
the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degrees of

Master of Engineering and Bachelor of Science

at the

Massachusetts Institute of Technology
May, 1994

Signature of Author .....
Department of Electrical Engineering and Computer Science
May 12, 1994

Certified by .....
James R. Glass
Research Scientist
Department of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by .....
Frederic R. Morgenthaler
Chairman, Department Committee on Graduate Students

Automatic Orthographic Alignment of Speech

by

Jerome S. Khohayting

Submitted to the Department of Electrical Engineering and Computer Science
in May, 1994 in partial fulfillment of the requirements for the Degree of
Master of Engineering and Bachelor of Science.

Abstract

The objective of the research is to develop a procedure for automatically aligning an orthographic transcription with the speech waveform. The alignment process was performed at the phonetic level. The words were converted into a sequence of phonemes using an on-line dictionary and a network of phones was created using phonological rules.

The phonetic alignment was accomplished using a microsegment-based approach which utilized a probabilistic framework. Acoustic boundaries were proposed at points of maximal spectral change, and were aligned with the phone network using a time-warping algorithm incorporating both acoustic and durational weights to the score. This approach was then compared to the traditional frame-based technique, which can be interpreted as proposing boundaries at every frame before the process of aligning with the network.

The acoustic as well as the durational models were trained on the TIMIT corpus, and the algorithm was tested on a different subset of the corpus. To investigate the robustness of the procedure, the resulting algorithm was trained and evaluated on the same subsets, respectively, of the NTIMIT corpus, which is the telephone version of the TIMIT utterances. A word overlap ratio of 93.2 % and a word absolute error of 20.4 ms were achieved on the TIMIT corpus. This corresponds to a phone overlap ratio of 81.7 % and a phone absolute error of 12.9 ms.

Thesis Supervisor: James R. Glass
Title: Research Scientist

Acknowledgments

I wish to express my deepest gratitude to my thesis advisor, Jim Glass, for all the knowledge, support, patience and understanding he has given me for the last two years. He has devoted a significant amount of his precious time to guide me through my work, and I have learned so much from such occasions. Specifically, I thank him for his help on programming and spectrogram reading.

I also wish to thank all the members of the Spoken Language Systems Group for their friendship. In particular, I would like to thank:

Victor Zue, for letting me join the group, for providing an environment conducive to learning, and for his sense of humor,
Lee Hetherington and Dave Goddeau, for answering a lot of my questions on UNIX, C and LATEX,
Mike Phillips, for creating a lot of the tools I have used,
Mike McCandless, for all his help on C and MATLAB,
Nancy Daly, for all her help on phonetics,
Bill Goldenthal, for all his interesting insights on interview techniques, sports and machine allocation,
Eric Brill, for all his views on the stock market,
Jane Chang, for being a great officemate and for putting up with me,
Christine Pao and Joe Polifroni, for keeping our systems running,
Vicky Palay and Sally Lee, for everything they've done to maintain and organize the group.

Special thanks goes to Tony Eng for reading a draft of my thesis and providing helpful comments. I also thank Mike Ong Hai, Bobby Desai and Daniel Coore for all the good times.

Finally, I wish to thank my parents for all their love and encouragement.
This research was supported by ARPA under Contract N00014-89-J-1332 monitored through the Office of Naval Research, and in part by a research contract from the Linguistic Data Consortium.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Automatic speech recognition has been a tantalizing research topic for many years. To be successful, speech recognizers must be able to cope with many kinds of variability: contextual variability in the realization of phonemes (i.e., coarticulation), variability due to vocal tract differences between speakers and speaking styles, and acoustic variability due to changes in microphone or recording environment. In order to be able to model such variability, researchers use large corpora of speech to train and test acoustic-phonetic models. Corpora are especially useful when orthographically and phonetically aligned transcriptions are available along with the speech [13]. Such corpora are also valuable sources of data for basic speech research. An example of an aligned speech utterance is shown in Figure 1.1. In the figure, the spectrogram for the sentence "Two plus seven is less than ten" is shown, as well as its aligned phonetic and orthographic transcriptions.

Manual transcription of large corpora, however, is extremely time consuming, so automatic methods are desired. There are different degrees of automating this process. One method would require a phonetic transcription of an utterance before align-

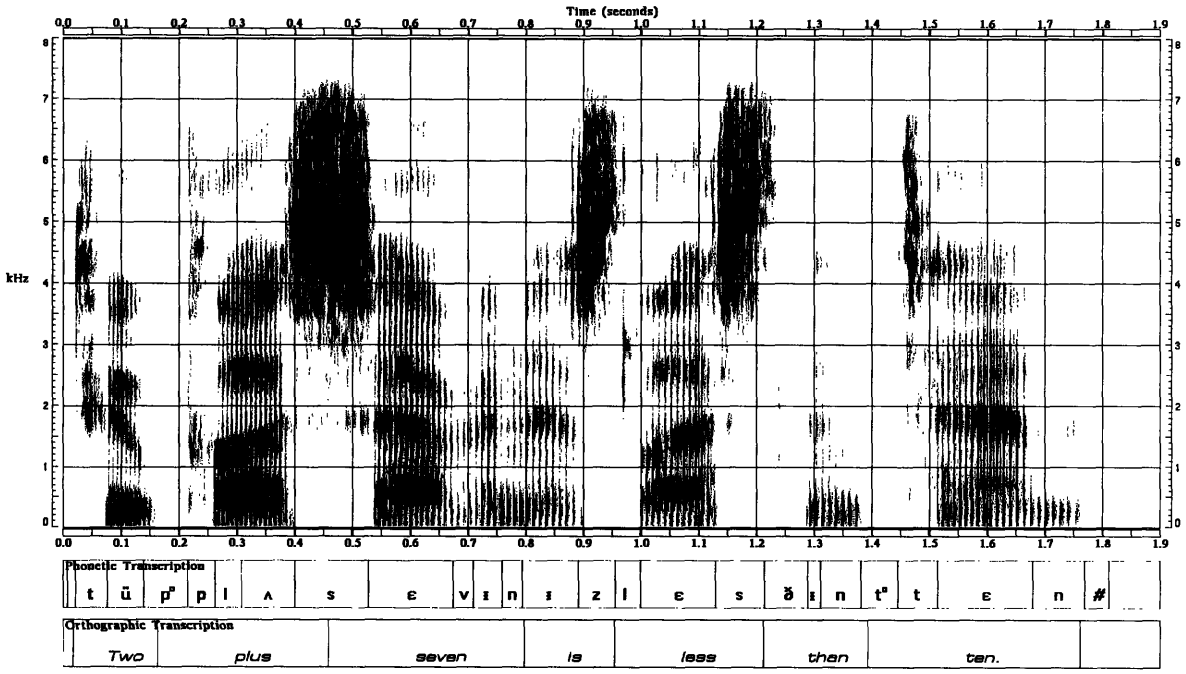

Figure 1.1: Automatic Orthographic Alignment of Speech
Digital spectrogram of the utterance "Two plus seven is less than ten" spoken by a male speaker. A time-aligned phonetic and orthographic transcription of the utterance are shown below the spectrogram.

ing it with the speech signal [14]. A fully automated system would require only the orthography of the utterance, and could generate a phonetic network automatically before aligning it with the waveform. This research focuses on the latter approach.

In the past, researchers have typically used a frame-based approach for phonetic alignment. This research investigates a segment-based approach utilizing a stochastic framework. An on-line dictionary is used to expand words into a baseform phoneme sequence. This sequence is then converted to a phone network by applying a set of phonological rules. The rules are intended to account for contextual variations in the realization of phonemes. The phone sequence which best aligns with the speech signal is determined using the phonetic alignment algorithm.

The aligment procedure is evaluated primarily on the TIMIT [13] and NTIMIT [11] acoustic-phonetic corpus. Comparisons to results reported in the literature are made whenever possible.

1.2 Previous Work

There has been considerable work done on the problem of orthographic transcription by speech researchers in the past. The method of orthographic transcription is normally accomplished by converting the text into possible phone sequences and performing phonetic alignments with the speech signal. In this section, the work done on phonetic alignment, as well as the techniques developed for orthographic alignment, are described.

1.2.1 Phonetic Alignment

Frame-based Techniques

The most common approach to phonetic alignment is the use of frame-based probabilistic methods. These methods use acoustic models to perform time-alignment based on the frame by frame statistics of the utterance. Dalsgaard [7] used a self-organizing neural network to transform cepstral coefficients into a set of features, which are then aligned using a Viterbi algorithm. Ljolje and Riley [15] used three types of independent models to perform the alignment algorithm–trigram phonotactics models which only depended on the phoneme sequence on the sentence, acoustic models which used three full covariance gaussian probability density functions, and phone duration models. The three types of models were interrelated through a structure similar to a second order ergodic continuous variable duration HMM (CVDHMM). They reported an 80% accuracy for placing the boundaries within 17 milliseconds (ms) of the manually placed ones. In a follow-up work [16], they utilized a similar CVDHMM structure but performed training and testing on separate utterances spoken by the *same* speaker. The results were naturally better–an 80% accuracy within a tolerance of 11.5 ms was achieved instead. Brugnara [4] and Angelini [2] also used HMM's to model the acoustic characteristics of the frames and the Viterbi algorithm to perform segmentation. Blomberg and Carlson [3] used gaussian statistical modelling for the parametric as well as spectral distributions, and for the phone and subphone durations. Fujiwara [9] incorporated spectrogram reading knowledge in his HMM's and performed the segmentation algorithm.

Segment-based Techniques

An alternative approach was taken by Andersson and Brown [1] and by Farhat [8]. Both initially divided the speech waveform into short segments composed of several

frames having similar acoustic properties. Anderson and Brown classified the signal into voiced/unvoiced segments using a pitch-detection algorithm. Corresponding segments of voiced/unvoiced events were generated from the text, and a warping algorithm was used to match the segments. The speech signals used were typically several minutes long and reasonable results were achieved. Farhat and his colleagues compared performing time-alignment using a segmental model to a centisecond one. In the latter approach, mel-frequency cepstral coefficients (MFCC's) were computed every 10 ms and the Viterbi algorithm was used to do the matching. In the former, the speech signal was first segmented with a temporal method. A similar vector of MFCC's was computed every segment, and a similar Viterbi search was employed to perform the alignment. The segmental approach achieved better results; using context independent models, it produced a 25% disagreement with manual labelling allowing a tolerance of 20 ms, as opposed to 35% for the centisecond approach. These works suggest that the segmental methods are at least as successful as their frame-based counterparts.

Other Techniques

The work of Leung [14] consists of three modules. The signal is first segmented into broad classes. Then these broad classes are aligned with the phonetic transcription. A more detailed segmentation and refinement of the boundaries is then executed. On a test set of 100 speakers, the boundaries proposed by the alignment algorithm were within 10 ms of the correct ones 75% of the time.

1.2.2 Orthographic Alignment

On the more general problem of orthographic transcription, Ljolje and Riley [15] used a classification tree based prediction of the most likely phone realizations as input

for the phone recognizer. The most likely phone sequence was then treated as the true phone sequence and its segment boundaries were compared with the reference boundaries. Wheatley et al [18] automatically generated a finite-state grammar from the orthographic transcription uniquely characterizing the observed word sequence. The pronunciations were obtained from a 240,000-entry on-line dictionary. A separate path through the word-level grammar was generated for each alternate pronunciation represented in the dictionary. The word pronunciations were realized in terms of a set of context-independent phone models, which were continuous-density HMM's. With these phone models, the path with the best score was chosen and an orthographic transcription was obtained.

1.3 Corpus Description

The TIMIT and NTIMIT acoustic-phonetic corpora are used in this thesis. The TIMIT corpus was recorded with a closed-talking, noise-cancelling sennheiser microphone, producing relatively good quality speech [13]. It is a set of 6,300 utterances spoken by 630 native American speakers of 8 dialects, each speaking a total of ten sentences, two of which are the same across the training set. The NTIMIT corpus is formed by passing the TIMIT utterances over the telephone line, producing speech which is noisier with a more limited bandwidth [13]. Evaluating alignment algorithms on NTIMIT gives an indication of the robustness of the algorithm to different microphones or acoustic environments.

In the training of the TIMIT sentences, a set of 4536 sentences uttered by 567 speakers each speaking eight sentences is used. The training speakers are listed in Appendix D.1. In the evaluation of the TIMIT utterances, a set of 250 sentences uttered by 50 speakers each speaking five sentences is used. The test speakers are listed in Appendix D.2. No test speaker was part of the training set. For the NTIMIT corpus,

the corresponding subsets of the database are used, respectively, for the training and testing procedures.

These corpora come with their phonetic and orthographic sequences, together with the time boundaries for each of these sequences, i.e. the phonetic and orthographic transcriptions. This creates the possibility of a supervised training of the acoustic and durational models used. Moreover, this aids in the testing of the algorithm, because the supposedly correct answer is known and hence can be compared to the transcription derived from the alignment algorithm.

1.4 Transcription Components

The ultimate objective of this thesis is to present a segmental approach to the problem of aligning in time the speech waveform with its orthographic transcription. A schematic diagram for the alignment algorithm is shown in Figure 1.2.

First, acoustic models for each of forty-two phone classes are trained, using mel-frequency cepstral coefficients, described in Section 2.2.2, as the acoustic parameters. Durational models are likewise created.

The second step is to propose phone boundaries from the acoustic data. The idea is that the more boundaries proposed, the chances of the exact boundaries each being located near a proposed boundary is higher. Then it remains to identify these "closest" boundaries among all the ones proposed. Proposing too many boundaries makes this latter problem harder.

The next step is to create a network of possible phone sequence given the orthographic transcription. An on-line dictionary is used to create a baseform phoneme sequence. The phonemes of the words are concatenated together to form the phoneme sequence of the sentence. Then, by applying a set of phonological rules, a network

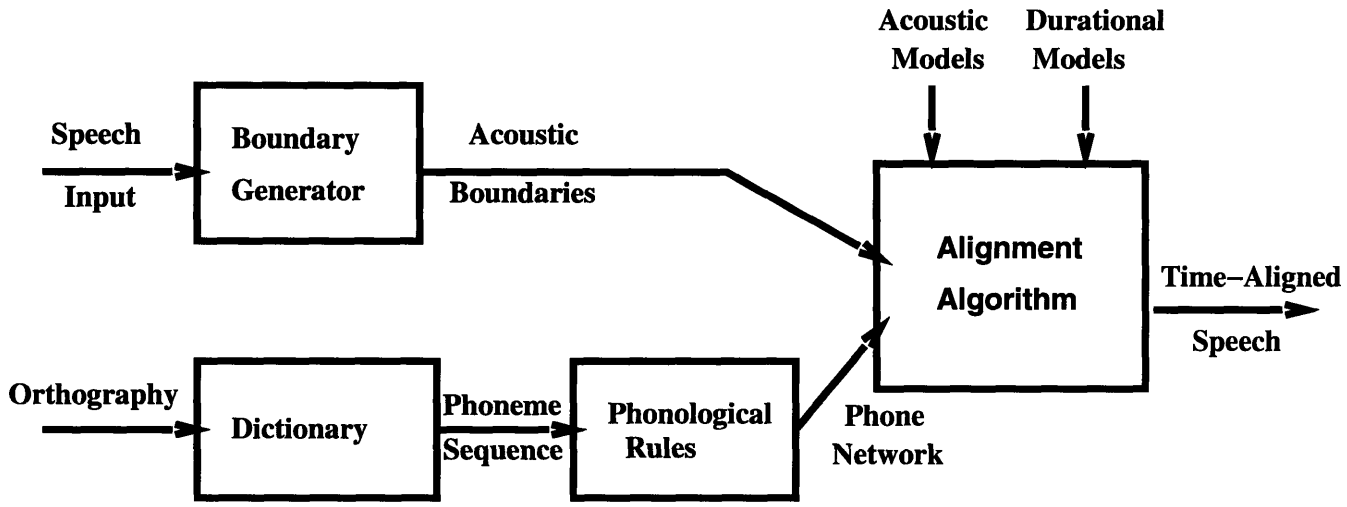

Figure 1.2: Schematic Diagram for the Alignment System

of possible phone sequences is formed from this phoneme sequence. The rules are intended to account for contextual variations in the realization of phonemes.

Then, a Viterbi algorithm is used to traverse through the paths in the network of phone sequences. For each path, a phonetic alignment is performed and a probability is determined. Such a score includes acoustic as well as durational components. The path with the best likelihood is then chosen and consequently an orthographically aligned transcription is achieved.

A frame-based technique in which each frame is proposed as a boundary is also developed. The network creation and alignment procedure is the same as the segment-based approach, and the results are compared. Finally, this whole alignment procedure is trained and tested on the TIMIT and the NTIMIT databases.

1.5 Thesis Outline

A brief outline of the thesis is presented here. In Chapter 2, the probabilistic framework is presented, and the development and training of acoustic and durational models are described. Classification experiments at the frame, microsegment and phone levels are also carried out.

The boundary generation algorithm is discussed in Chapter 3. The various criteria and parameters are enumerated, and the results are discussed. A set of optimal boundary parameters is selected in the last section.

In Chapter 4, a discussion of the phonological variations is presented, and the various phonological rules used for the network creation are described.

The alignment procedure is described at length in Chapter 5. Two different search methods, the two-pass method and the full segment method, are presented and evaluated. The two-pass method is observation-based and experiments using frames and microsegments as observations are separately performed.

Finally, a summary of the results of the alignment procedure is presented in Chapter 6. Some possible future directions for the problem of time-alignment are also described here.

Chapter 2

Modelling and Classification

The acoustic modelling for the alignment process is presented in this chapter. This includes a discussion of the probabilistic framework as well as the different acoustic parameters. To test the accuracy of the models, classification experiments are performed, and the results are analyzed.

2.1 Probabilistic Framework

In this section the acoustic framework is described. A durational component shall be added in Chapter 5 when the actual search is performed. Let $\{\mathbf{S_i}\}$ be the set of possible transcriptions of the speech signal given the word sequence, let $\vec{\mathbf{a}}$ be the acoustic representation of the speech signal, and let $\mathbf{T}$ be the observations. These observations could range from speech frames to bigger intervals like microsegments or full segments. Then, $S^* = S_j$, where

$$j = \arg\max_i \Pr(\mathbf{S_i} \mid \vec{\mathbf{a}}, \mathbf{T}). \tag{2.1}$$

By Bayes' Law,

$$j = \arg\max_i \frac{\Pr(\vec{\mathbf{a}}, \mathbf{T} \mid \mathbf{S_i}) \cdot \Pr(\mathbf{S_i})}{\Pr(\vec{\mathbf{a}}, \mathbf{T})}. \tag{2.2}$$

The factor $\Pr(\vec{\mathbf{a}}, \mathbf{T})$ is constant over all $\mathbf{S_i}$, and can be factored out. Moreover, there is no language modelling techniques involved in this work. The factor $\Pr(\mathbf{S_i})$ is assumed to be equally probable over all i. Hence it remains to find

$$\Pr(\vec{\mathbf{a}}, \mathbf{T} \mid \mathbf{S_i}). \tag{2.3}$$

Let N be the number of phones in $\mathbf{S_i}$, and M be the number of observations in $\mathbf{T}$. We can then express the phones as $S_{i_l}, 1 \leq l \leq N$, the observations as $T_k, 1 \leq k \leq M$, and the acoustic information in the observation k as $\vec{\mathbf{x}}_{T_k}$. Equation 2.3 is then equivalent to:

$$\Pr(\vec{\mathbf{x}}_{T_1}\vec{\mathbf{x}}_{T_2}...\vec{\mathbf{x}}_{T_M} \mid S_{i_1}S_{i_2}...S_{i_N}). \tag{2.4}$$

For facility in computation, we approximate Equation 2.4 as:

$$\prod_f \Pr(\vec{\mathbf{x}}_{T_k} \mid S_{i_l}), \tag{2.5}$$

where f is the set of mappings from the set of phones to the set of observations, subject to the following conditions:

1. Each phone S_i is mapped to at least one observation T_k,
2. Each observation T_k maps to exactly one phone S_i,
3. The mapping preserves the order of the phone sequence with respect to the observation sequence.

Furthermore, S_{i_l} shall be replaced by its corresponding phone class as described in Section 2.2.1, since models were only trained for each class.

In practice, speech researchers have actually maximized the log of Equation 2.5. This does not change the optimal path, since the log function is one-to-one and strictly increasing. Its advantage is that the floating point errors are minimized since the log of the product in Equation 2.5 becomes a sum of logarithms:

$$log \prod_{f} \Pr(\vec{\mathbf{x}}_{T_k} \mid S_{i_l}) = \sum_{f} log \Pr(\vec{\mathbf{x}}_{T_k} \mid S_{i_l}). \tag{2.6}$$

A multivariate normal distribution is trained for each phone class α, and the following formula holds for the probability of an acoustic vector given a phone class.

$$\Pr(\vec{\mathbf{x}} \mid \alpha) = \frac{1}{(2\pi)^{p/2} det(\sum_{\alpha})^{1/2}} e^{-\frac{1}{2}(\vec{\mathbf{x}}-\vec{\mu}_{\alpha})' \sum_{\alpha}^{-1} (\vec{\mathbf{x}}-\vec{\mu}_{\alpha})}, \tag{2.7}$$

where $\vec{\mathbf{x}}$ is the characteristic MFCC vector of the observation, p is the number of dimensions of the acoustic vector, and $\vec{\mu}_{\alpha}$ and $\sum_{\alpha}$ are the mean MFCC vector and the the covariance matrix of phone class α, respectively [12]. Taking logarithms, we get:

$$log \Pr(\vec{\mathbf{x}} \mid \alpha) = -\frac{p}{2} \cdot log(2\pi) - \frac{1}{2} \cdot det(\sum_{\alpha}) - \frac{1}{2}(\vec{\mathbf{x}} - \vec{\mu}_{\alpha})' \sum_{\alpha}^{-1} (\vec{\mathbf{x}} - \vec{\mu}_{\alpha}). \tag{2.8}$$

The maximization of Equation 2.6 will be described in Chapter 5. Depending on the nature of the speech observations, different procedural variations are used.

2.2 Modelling

2.2.1 Phone Classes

There are forty-two phone classes used in this research. The sixty-three TIMIT labels are distributed among the phone classes according to their acoustic realization.

For instance, the phones [m] and [m̩] (syllabic m) are combined into a phone class. Likewise, the stop closures of the voiced stops, [b□], [d□], and [g□], are grouped together, as well as the stop closures of the unvoiced ones, [p□],[t□], and [k□]. The complete list of the phone classes is shown in Appendix A in IPA and ARPABET form.

There are several advantages of grouping the phones in this manner. First, the number of classes is one-third less than the number of phones, hence there will be some savings in terms of memory storage. Second, there will be computational savings in parts of the alignment program where the classes are trained or accessed. Third, some phones very rarely occur and hence the number of their observations in the training set is very few if any. Creating separate models for such phones will not make the training robust, and so they are grouped with other phones having similar acoustic properties. Finally, since the phones within a class have very similar acoustic properties, there will not be too much degradation in terms of performance. This is because the problem at hand is alignment, hence not too much concern is put on the actual phone recognition, but rather on predicting the time boundaries.

2.2.2 Signal Representation

The speech signal is sampled at the rate of 16 KHz. Then, every five ms, a frame is created by windowing the signal with a 25.6 ms Hamming window centered on it. The Discrete Fourier Transform (DFT) of the windowed signal is performed, the DFT coefficients are squared, and the magnitude squared spectrum is passed through forty triangular filterbanks [17]. The log energy output of each filter form the forty mel-frequency spectral coefficients (MFSC), $X_k, 1 \leq k \leq 40$, at that frame. Then, fifteen mel-frequency cepstral coefficients (MFCC), $Y_i, 1 \leq i \leq 15$, are computed from the MFSC's by a cosine transformation:

$$Y_i = \sum_{k=1}^{40} X_k \cdot \cos[i(k-1/2)\pi/40], \qquad 1 \leq i \leq 15 \tag{2.9}$$

The delta MFCC's are computed by taking the first differences between the MFCC's of two frames at the same distance from the present frame. This distance can be varied. If the present frame is labelled n, a delta of N means that the first difference D_i is given by

$$D_i[n] = Y_i[n+N] - Y_i[n-N], \qquad 1 \leq i \leq 15. \tag{2.10}$$

The delta values used in this research varied from one to seven, corresponding to a difference of 10 to 70 ms.

2.2.3 Acoustic Modelling

The acoustic parameters used in this thesis are primarily the MFCC's and the delta MFCC's. These parameters are computed for each speech observation in the following way. For each such segmental observation, an average for each MFCC dimension is computed. Experiments are conducted with and without the delta MFCC's. Without these, the number of dimensions N is fifteen corresponding to the fifteen MFCC's. With these, delta MFCC's are computed at both boundaries of the observation, and N increases to forty-five. Experiments have also been performed to take advantage of the context. Hence, for each observation, the acoustic parameters for the adjacent observations are used as additional parameters for the present observation, increasing the dimensionality of the mean vectors and covariance matrices.

2.2.4 Durational Modelling

Experiments are also conducted to include a durational component. This is done to add more information to the acoustic modelling, preventing certain phones from

having too long or too short a duration. For each phone, the score for all the observations hypothesized for that phone will be incremented by a durational component corresponding to the total length of the duration of all the observations. Gaussian durational models will be trained for each of the phones, and the logarithm of the probability computed based on the models will be treated as the durational score. The weight of each durational component with respect to the acoustic components is varied and the results compared.

2.3 Training

As noted in Section 1.3, the TIMIT database include the manually-aligned phonetic transcription boundaries. This allows for the possibility of a supervised training of the acoustic and durational models.

2.3.1 Acoustic Models

In the training algorithm for the acoustic models, the observations are used as the basic units for the training. In a frame-based approach, the observations are just the frames themselves, and in a segment-based approach, they are the microsegments, which are the output of the boundary generation algorithm described in Chapter 3. The observations formed are then aligned with the labelled boundaries in the following way. The left and right boundaries of each phone in the *correct* aligned transcription is aligned with the closest proposed boundary. Then, every observation between these two proposed boundaries will be labelled with the phone. There are two possible phonetic transcriptions that will arise. The first, which I call the "boundary transcription", considers each observation as independent, and allows a sequence of microsegments with the same phonetic label. It will have a total of the number of boundaries minus one elements in the transcription. The second, termed "segment

transcription", considers the whole segment between the two closest proposed boundaries, respectively, to the boundaries of the phone as one element in the transcription. It is then appropriately labelled with the phone.

From the above algorithm, it is possible that both boundaries of a phone be matched to the same proposed boundary. In this case the phone is considered *deleted.* This problem will not arise in the case of a frame-based approach since all phones in the TIMIT corpus are longer than 5 ms. In a microsegment-based approach, the boundary generation algorithm should be accurate enough that the deletion rate will be low. This issue will be discussed more in the next chapter.

The acoustic models are trained on the boundary transcription through the following method. An average of the MFCC's is computed per observation and depending on the experiment, some other acoustic parameters such as the delta MFCC's might be computed as well. For each phone class, a multivariate normal distribution is assumed, the occurrences in the training set of all the phones in the class are assembled and the mean vector and covariance matrix for the phone class are computed.

2.3.2 Durational Models

The durational models consist of a mean duration and a standard deviation for each phone, instead of each phone class. The storage costs for such a phone model is cheaper and more accuracy can be attained this way. Once again, the TIMIT database is used for the training, where for each phone, the durations of all the occurrences of the phone in the aligned transcriptions are taken into account when computing the phone duration's mean and standard deviation.

2.4 Classification

Phone classification of a speech segment is not strictly a subproblem of time-alignment. In maximizing the best path in a network and choosing which phone an observation of speech maps to, there is no need to choose the best phone among all the phones. The choices of phones are constrained by the structure and the phones of the pronunciation network. Nevertheless, classification experiments give an indication of how good the acoustic modelling is. If the classification rate improves, this means that the models are able to discriminate between sounds and it is plausible to believe that the alignment performance will likewise improve. In this section, simple classification experiments are performed and the results of these experiments are evaluated. Instead of classifying the phones themselves, the objective is to classify a speech segment from among the forty-two phone classes described in Appendix A.

Given an interval of speech, a feature vector $\vec{\mathbf{x}}$ of speech is computed, and the phone class β is chosen according to:

$$\beta = \arg\max_{k} log \Pr(\beta_{\mathbf{k}} \mid \vec{x}), \qquad 1 \leq k \leq N, \tag{2.11}$$

where the log of the probability is computed using Equation 2.8, using appropriate acoustic models, and N is the number of phone classes. The percentage of the time that the chosen class matches the class of the correct phone is the classification rate.

2.4.1 Frame Classification

The first experiments involved the analysis frames as observations. The models trained from Section 2.3 using the frames as observations and creating a boundary transcription structure from these observations are employed here. The same

procedure for training was used on the test utterances. Hence, in this case, each utterance was divided into frames, and classified into some phone class. Given a frame, a feature vector $\vec{x}$ was computed, using the same set of features used in the training algorithm.

A feature vector of fifteen MFCC's computed from Equation 2.9 was computed on the TIMIT database. When the models were trained on these and then tested, a classification rate of 44.85% was achieved. The same experiment on the NTIMIT database resulted in a 35.60% classification rate.

2.4.2 Microsegment Classification

The second set of experiments involved the microsegments as the observations. Exactly the same procedure was used as in the case of frame classification, except that other feature vectors were considered. The results are summarized in Table 2.1.

Database	*Delta(δ)*	*#Dimensions*	*Classification Rate(%)*
TIMIT	0	15	41.67
TIMIT	1	45	45.21
TIMIT	4	45	49.35
TIMIT	7	45	49.53
NTIMIT	0	15	34.93

Table 2.1: Microsegment Classification Results Using Different Deltas

It should be noted at this point that because the boundaries did not exactly coincide with the actual phone boundaries, there are many instances that part of microsegments labelled with a certain phone is not actually part of the phone segment in the correct transcription and this contributes to the error. Comparing these results with the ones for the frame classification, two conclusions can be made. One is that the classification improves from no delta to positive delta. This is because in considering

the microsegments, there is relatively less noise or randomness from one to the next, and the information carried by the delta parameters is enhanced. The second is that the rate improves more as delta is further incremented. This is probably due to an even less randomness from one phone to several phones away from it, and hence a better representation of change.

The next set of experiments take advantage of the context. Given a microsegment, the acoustic vector was augmented by the vectors of the M microsegments before and after it. The number M was varied, and the results shown in Table 2.2. The deltas in Table 2.2 are still with respect to the original microsegment. From the table, one sees that a similar phenomena occurs as in frame classification, that is, that the addition of deltas does not help the classification rate. In this case, the reason is that the delta MFCC's do not correlate with the additional microsegments, hence adding these parameters hinder the performance. It should also be noted that as more microsegments were added, the rate improves. The best result was when there are three additional microsegments on each side and no delta parameters, where a microsegment classification rate of 54.12% was achieved. A phone on average spans 4.4 microsegments, and as we approach this number more and more information about the phone is represented in the feature vector, and this helps the classification rate. It is also interesting to note that adding the delta parameters does not hurt the classification rate much when trained and tested on the NTIMIT database.

As a final experiment on the microsegments, instead of one set of fifteen MFCC averages, three sets are used. A microsegment is divided into three equal subsegments and a fifteen-dimensional MFCC vector is computed per subsegment, giving a total of forty-five acoustic dimensions. This time, no additional microsegments were added on either side, and as the delta was varied, the rate went down and then back up. The results are summarized in Table 2.3.

In this example we can see the tradeoff of having adding delta parameters. Delta

Database	*Additional μSegments on Both Sides (M)*	*Delta(δ)*	*#Dimensions*	*Classification Rate (%)*
TIMIT	1	0	45	49.38
TIMIT	1	1	75	45.69
TIMIT	2	0	75	52.74
TIMIT	2	1	105	49.84
TIMIT	3	0	105	54.12
TIMIT	3	1	135	51.92
TIMIT	3	4	135	51.62
TIMIT	3	7	135	51.42
NTIMIT	1	0	45	39.76
NTIMIT	1	1	75	39.74
NTIMIT	2	0	75	43.80
NTIMIT	2	1	105	43.59
NTIMIT	3	0	105	46.03
NTIMIT	3	1	135	45.66

Table 2.2: Microsegment Classification Results Using Different Deltas and Additional Microsegments

Database	*Delta(δ)*	*#Dimensions*	*Classification Rate (%)*
TIMIT	0	45	43.99
TIMIT	1	75	39.26
TIMIT	2	75	43.10
TIMIT	3	75	45.41
TIMIT	4	75	46.98
TIMIT	5	75	48.25
TIMIT	6	75	48.98
TIMIT	7	75	49.15

Table 2.3: Microsegment Classification Results Using Different Deltas and Three Averages

parameters are advantageous in that they embody some representation of change characteristic to phones, and this is evident for larger deltas ($\delta = 6, 7$). A small, positive delta, however, incorporates some noisy information and this outweighs the advantages.

2.4.3 Segment Classification

Segment classification is based on the segment phonetic transcription structure described in Section 2.3.1. Acoustic models were trained on this structure and the models were tested on the same structure of the utterances in the test set. Aside from varying the delta parameter, sometimes three MFCC averages per segment were computed. These represent the beginning (onset), middle and end (offset) of the phone, respectively.

The results are shown in Table 2.4. The best result of 69.0% was achieved when three averages are computed and the delta value was 7. In general, the numbers in Table 2.4 are naturally significantly higher than any of the numbers for microsegment or frame classification. This is because the segments in the segment transcription structure roughly correspond to the whole phones themselves except for some relatively small boundary error. Moreover, the results improve again as δ is increased above $\delta = 1$ for the same reasons as before, but tapers off after $\delta = 7$. Likewise, the classification improves as the number of averages computed increases.

Most classification experiments in the literature are segment-based, and don't propose any boundaries to create the segments. Instead, the exact segments are used in the training and testing algorithm. To simulate the same experiments, the same procedure was performed. The results are shown in Table 2.5, with varying number of averages and deltas.

These results are very similar to those of Table 2.4. In fact, they're only slightly

Database	*# Averages*	*Delta(δ)*	*#Dimensions*	*Classification Rate (%)*
TIMIT	1	1	45	57.82
TIMIT	1	4	45	64.62
TIMIT	1	7	45	64.39
TIMIT	3	7	75	68.99
NTIMIT	1	1	45	45.54

Table 2.4: Segment Classification With Automatically Aligned Boundaries

Database	*Delta(δ)*	*#Averages*	*#Dimensions*	*Classification Rate (%)*
TIMIT	0	1	15	49.15
TIMIT	0	3	45	61.52
TIMIT	1	1	45	58.87
TIMIT	1	3	75	65.11
TIMIT	4	1	45	65.34
TIMIT	4	3	75	68.71
TIMIT	7	1	45	65.39
TIMIT	7	3	45	69.12

Table 2.5: Exact Segment Classification

better, due to the fact that there are few errors caused by the boundary generation algorithm. The best result of 69.1% was achieved when delta was 7 and three averages per segment were computed. It is also important to note that the improvement in performance caused by an increase in the number of averages computed is more substantial when δ is lower. The classification rate only counts the number of correctly classified segments, and does not account for near misses. The noisy fluctuations generated from picking small deltas produce a great deal of error where the correct phone classes are very near the top choice, and the increase in the number of averages help increase the probability of the correct phone classes enough to become the top choice. The errors in picking bigger deltas, on the other hand, are probably more serious, where the correct class is relatively farther to the top choice, and the augmentation of acoustic vectors does not help much.

2.5 Summary

Full covariance gaussian acoustic models are trained at the frame, microsegment, and full segment levels, and will be used in implementing the alignment algorithm. In the training of the models, parameters, such as δ and the number of averages computed per observation, were varied. To determine which parameters resulted with the most accurate models, classification experiments were performed. In general, results improved when δ values of 4 or higher were used and when more averages were computed per observation.

Chapter 3

Boundary Generation

The boundary generation procedure is described in this chapter. Different acoustic parameters are experimented with, and the results are evaluated using a set of evaluation criteria.

3.1 Independence of Microsegments

In Chapter 2, the approximation in Equation 2.5 is used for computing the scores of different phone paths. Whenever such a probability is expressed as a product of probabilities, there is an assumption of independence between the different observations in the path. Whether such an independence premise is valid or not greatly determines the accuracy of such an approximation.

For frame-based procedures, the independence assumption is severely violated. The average phone duration is approximately 80 ms [6]. Hence, if the present frame is an [s] say, it is highly likely that the succeeding frame will also correspond to an [s]. By grouping frames together into longer segments, one segment will be more likely to be independent from its adjacent segments. The ideal situation is that the signal be divided into the phones themselves. But this is exactly the problem we are

trying to solve! The method of predicting acoustic-phonetic boundaries for phones generally tends to hypothesize many boundaries at intervals of speech where there is significant acoustic change. However, there is usually at most one actual phone boundary at these regions. This is a good start, however, and is in fact the basis of the microsegment-based approach.

In this chapter, the method of boundary selection is described. Acoustic models are then trained from the microsegments formed out of this procedure. Finally, the alignment algorithm chooses the actual phone boundaries from these boundaries and produces an alignment.

3.2 Criteria

One disadvantage of using a microsegment-based approach is that if the boundaries are chosen poorly, certain microsegments overlap with more than one phonetic segment in the *correctly* aligned transcription, which means that such a microsegment does not correspond to exactly one phone. Such errors can be prevented if the boundaries are proposed more frequently. However, the more boundaries that are proposed, the greater the independence assumption is violated, and the less exact the approximation. Hence, there is a tradeoff between lesser boundary errors and lesser approximation errors. To maximize such a tradeoff, several measures are used to evaluate the *accuracy* of the boundaries.

3.2.1 Deletion Rate

The first measure of boundary accuracy is the deletion rate, which is computed in the following way. The proposed boundaries are aligned with the labelled boundaries. For each of the labelled boundary, the closest proposed boundary is found and a

new phonetic transcription is subsequently formed using these "closest" boundaries as the new phonetic boundaries. Occasionally, two or more labelled boundaries will match the same proposed boundary. The phones between these labelled boundaries are consequently deleted in the new phonetic transcription. Such an event is clearly undesirable. Hence, it is imperative the proposed boundaries be chosen such that these deletions are limited.

We can now then define the deletion rate as the ratio of the number of phones deleted over the total number of phones in the original transcription. This quantity is to be calculated for each boundary generation procedure to evaluate it.

3.2.2 Boundary-to-Phoneme Ratio

A second important measure of choice of boundaries is the boundary-to-phoneme ratio. To achieve more independence between the observations, a small boundary-to-phoneme ratio is desired. However, this should not be at the cost of huge deletions and errors. Another advantage of a small ratio is that with fewer boundaries, the number of observations per path in the network is smaller and hence the computation time is smaller. This becomes a big factor when real-time time-alignment algorithms are desired, or if the utterances are longer than normal.

3.2.3 Errors of Boundary Accuracy

Finally, a third measure is the absolute error of the boundaries when compared to the labelled transcriptions. As described earlier, a segment phonetic transcription can be achieved when the closest hypothesized boundary to each labelled phone boundary is found and used as a phoneme boundary for the new transcription. The average absolute difference between these new phone boundaries and the true boundaries can then be calculated.

Such a result tests the accuracy of predicting the exact boundaries. This is crucial since the training and testing algorithms will depend a lot on these new observations. A large error means that there will be erroneous input to the training algorithm. For instance, if the closest boundary to an [s t] sequence is very far to the left of the actual phone boundary between the [s] and the [t], then a significant part of the [s] will be trained as a [t], and this will prove costly in the testing procedure even if the alignment process is sound. Moreover, this *closest* phonetic transcription symbolizes the best the alignment procedure can do. A large average absolute error implies that a good alignment procedure would not be attainable.

3.3 Parameters

This section describes the parameters used in creating the boundaries, as well as reports the results of the evaluation. The parameters used are mainly acoustic parameters from the input speech signal, as well as some "smoothing" parameters.

3.3.1 Spectral Change and Spectral Derivative

The spectral change of the signal at frame i is computed from the spectrum by taking the euclidean distance of the MFCC vectors between a certain number of offset frames before and after the frame in consideration:

$$SC[i] = \sum_{k=1}^{N} (A_{i+\triangle i}[k] - A_{i-\triangle i}[k])^2, \tag{3.1}$$

where $SC[i]$ is the spectral change at time frame i, the quantity $A_i[k]$ is the value of the kth dimension of the acoustic parameter at time frame i, N is the number of components in the acoustic vector and $\triangle i$ is the offset. The offset used was 2 time

frames (10 ms), so that sudden changes in the spectral change due to noise will not be captured in the process of taking the euclidean distance.

Since the spectral change is a scalar function of time, the usual method of computing the spectral derivative is to just take a first difference on the spectral change:

$$D[i] = SC[i + \triangle i] - SC[i - \triangle i]. \tag{3.2}$$

However, the speech signal is noisy and taking a simple first difference will produce errors. Hence, an alternative method is used [10]. The spectral change function is smoothed by convolving it with a gaussian. The idea is that such a smoothing process will remove much of the noisy variations on the spectral change. Denoting the gaussian function by G, we have

$$SC' \approx (SC * G)' = SC * G' + SC' * G. \tag{3.3}$$

Since the speech signal is slowly time-varying, the second term on the right is small relative to the first, and we are left with the following expression for the derivative:

$$SC' \approx SC * G'. \tag{3.4}$$

In other words, the derivative of gaussian is computed first then is convolved with the spectral change function. The same method is used in computing further derivatives of the spectral change. A plot of the spectral change, spectral derivative and the second derivative of the spectral change for the utterance "How much allowance do you get?" is shown in Figure 3.1. There is generally a peak in the spectral change near phone boundaries. These correspond to positive-to-negative zero crossings of the spectral derivative.

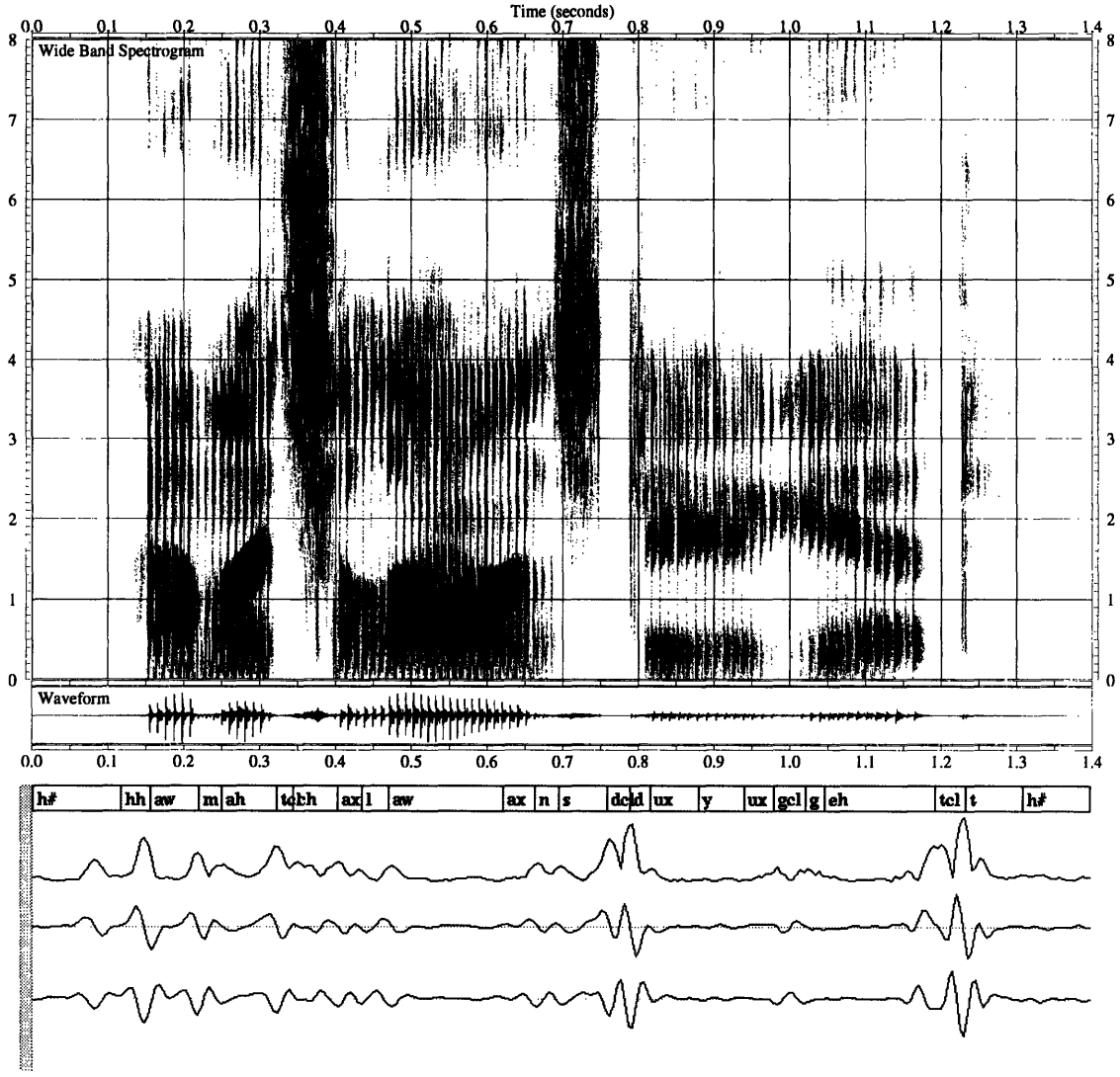

Figure 3.1: Time Plots of Acoustic Parameters

Spectrogram of the utterance "How much allowance do you get?" The plots of the spectral change, spectral derivative and second derivative waveforms are shown below the phonetic transcription.

3.3.2 Peaks in Spectral Change

Phone boundaries are often characterized by a huge spectral change, especially in the case of a closure-stop or a fricative-vowel sequence. Hence, it is natural to look for peaks in the spectral change. In dealing with a continuous waveform, a derivative is taken and the zeroes of this derivative are located and judged as maximas or minimas. Because of the discrete nature of the database, positive-to-negative zero crossings of the spectral derivative are located instead.

3.3.3 Threshold on the Spectral Change

Speech is often accompanied by noise and this noise produces small, random oscillations at various parts of the speech signal. Hence, in addition to the zero-crossing of the first spectral derivative, it is necessary to impose further conditions. Without these, there would be too many boundaries hypothesized at areas of the signal where there is no acoustic-phonetic boundary. This would inflate the boundary-to-phoneme ratio without providing much improvement on the deletion rate.

One possible condition is a threshold on the spectral change. This would exclude most of the tiny oscillations from being proposed as boundaries. However, such a threshold cannot be too small either, because a lot of phoneme boundaries are accompanied by very little spectral change, such as in the case of a vowel-semivowel sequence. Clearly, further modifications shall be needed to be able to identify most of these subtle phoneme boundaries without picking too many random boundaries.

3.3.4 Threshold on Second Derivative of Spectral Change

Another possible additional constraint is a threshold on the second derivative of spectral change. A positive-to-negative zero-crossing in the derivative of the spectral

change implies a peak in the spectral change. Imposing a threshold on the second derivative in addition to this would only pick boundaries at time frames where the slope of the first derivative zero-crossing is bigger, i.e. the transition from positive to negative in the spectral change is faster, implying a more dramatic change in the spectrum.

3.3.5 Associations on Acoustic Parameters

A variant of the threshold on the second derivative constraint is made by comparing first differences of the spectral change. If the previous first difference is above a certain positive value $\triangle d$ and the present is below $-\triangle d$, a boundary can be proposed. This condition corresponds to a significant change in the second derivative, but is more precise on the limits.

3.3.6 Constant Boundaries per Spectral Change

An alternative to having a small threshold above is to impose a larger threshold and then adding various boundaries between the acoustic boundaries. In adding new boundaries, one possibility is to consider the areas where the spectral change is more concentrated. Certainly, phoneme boundaries are more likely to appear at these regions than at areas where the spectral change is close to zero. Moreover, the longer a segment is, the more phone boundaries it probably contains. Taking these two facts into account, it is reasonable to propose adding boundaries at a constant rate per spectral change.

3.4 Results

In this section the results of different boundary generation algorithms are evaluated. The different parameters described in the previous section are varied, and the resulting deletion rate, boundary-to-phoneme ratio and error rates are compared.

3.4.1 Uniform Boundaries

For baseline performance comparison, the results of using uniform boundaries are evaluated. They are summarized in Table 3.1.

rate(ms. per bdy.)	*deletions(%)*	*ratio*	*error(ms.)*
5	0.0	16.16	0
10	0.1	8.08	2
15	0.3	5.40	4
20	1.1	4.06	5
25	1.9	3.25	6
30	3.2	2.72	7
35	4.9	2.34	8
40	6.1	2.05	10
45	8.6	1.83	11
50	11.3	1.65	12

Table 3.1: Results of Uniform Boundaries

The first entry in the table corresponds to a boundary rate of 5 ms per boundary. This is the same as proposing a boundary every frame. As expected, there are no errors or deletions, and the boundary-to-phoneme ratio is high, namely 16.2 boundaries per phone. The goal of the next boundary experiments is to significantly reduce this number without making the deletion rate or error too big. It is important to note that as the boundary rate is lowered, the deletions go up quickly in the case of uniform boundaries.

3.4.2 Acoustic Boundaries

Threshold on Second Derivative

The next experiment involves imposing a second derivative threshold to the spectral change, in addition to a positive-to-negative zero-crossing of the first derivative. The results are summarized in Table 3.2.

Second Der. Threshold	*deletions(%)*	*ratio*	*error(ms.)*
0	8.05	1.74	7.66
10	8.73	1.68	7.82
20	9.59	1.50	8.58
30	10.83	1.38	9.42

Table 3.2: Varying Threshold on Second Derivative of Spectral Change

These boundaries, however, are not satisfactory by themselves. A no-threshold condition on the second derivative, which means that the only constraints are on the first derivative, gives a very high deletion rate. Clearly, some other way of rewarding more boundaries is needed.

Associations

Another method of proposing acoustic boundaries is by using association on acoustic parameters. Tables 3.3 and 3.4 show the results of using associations on the cepstral and spectral coefficients, respectively, where the threshold is on the second order difference.

Threshold on 2nd Ord. Diff.	*deletions(%)*	*ratio*	*error(ms.)*
0.0	3.37	2.781	5.99
6.0	5.05	2.375	7.59
12.0	7.59	2.054	9.51
18.0	10.66	1.800	11.66
24.0	13.94	1.590	13.98

Table 3.3: Results of Associations on Cepstral Coefficients

Threshold on 2nd Ord. Diff.	*deletions(%)*	*ratio*	*error(ms.)*
0.0	3.38	2.808	6.08
3.0	10.30	1.835	11.55
6.0	20.04	1.308	19.39

Table 3.4: Results of Associations on Spectral Coefficients

These results show that associations permit a high deletion rate even though the ratio is low. Furthermore, the tradeoff is bad in relation to uniform boundaries. For instance, a uniform rate of 35 ms produces a deletion rate of 4.9% and a boundary-to-phoneme ratio of 2.34, whereas a 6.0 threshold on the second derivative results in a deletion rate of 5.1% and a ratio of 2.4.

3.4.3 Adding More Boundaries

The above methods by themselves do not produce desired low deletion rates. Clearly, somehow, a new ways of adding boundaries, possibly non-acoustic ones, is needed.

Uniform Boundaries

A simple way of adding boundaries is to put them at a constant rate. For instance, boundaries produced at a constant rate can be added to the set of boundaries derived

by imposing a second derivative threshold. The results of adding uniform boundaries assuming a threshold of 10.0 is presented in Table 3.5.

rate(ms. per bdy.)	*deletions(%)*	*ratio*	*error(ms.)*
10	0.09	9.65	2.71
15	0.16	6.97	3.46
20	0.37	5.63	4.04
25	0.60	4.83	4.53
30	0.69	4.29	4.90
35	0.87	3.91	4.57
40	1.09	3.62	4.90
45	1.42	3.40	5.16
50	1.75	3.22	6.07

Table 3.5: Results of Adding Uniform Boundaries To Second Derivative Threshold

From Table 3.1, a uniform rate of 20 ms per boundary produces a deletion rate of 1.1 percent and a 4.1 boundary-to-phoneme ratio. From Table 3.5, if we add boundaries which are zero-crossings of the first derivative and whose second derivative is above 10, then the deletion rate goes down to a tolerable 0.37 percent, but the boundary-to-phoneme ratio increases only to 5.6. This is clearly a favorable tradeoff. However, the procedure of adding uniform boundaries is a naive way of doing so, since no acoustic information is taken into consideration. Hence there is room for improvement.

Constant Boundaries Per Spectral Change

A more clever way of increasing the boundary ratio is to propose new boundaries where they are more probable to appear. As described before, a way to do this would be to propose constant boundaries per spectral change. Table 3.6 shows the results on adding new boundaries in this manner to those already proposed by imposing a

threshold on spectral change.

Threshold	*Bdy. Per Sp. Change* ($x10^{-4}$)	*deletions(%)*	*ratio*	*error(ms.)*
0.0	0.0	8.17	1.744	8.45
100.0	0.0	9.18	1.570	9.08
175.0	0.0	13.81	1.189	12.26
175.0	3.0	4.70	1.731	8.46
175.0	5.0	1.85	2.399	6.78
175.0	10.0	0.53	3.688	5.05
175.0	14.0	0.27	4.464	4.50
175.0	25.0	0.20	5.874	3.77
200.0	0.0	15.86	1.110	13.77
200.0	3.0	5.05	1.692	8.79
200.0	5.0	1.93	2.366	6.96
200.0	10.0	0.51	3.660	5.08
200.0	14.0	0.29	4.446	4.51
200.0	25.0	0.20	5.879	3.77

Table 3.6: Adding Constant Boundaries Per Spectral Change To Spectral Change Threshold

By comparing Tables 3.5 and 3.6, we see that the results of the latter are indeed much better. For instance, in Table 3.5, a uniform rate of 20 ms per boundary produces a deletion rate of 0.37% and a boundary rate of 5.6, whereas a threshold of 175.0 and a rate of $14 \cdot 10^{-4}$ in Table 3.6 results in a deletion rate of 0.27% and a boundary rate of 4.5, an improvement in both categories.

Similar results occur when we add new boundaries to those imposed by a second derivative threshold. They are shown in Table 3.7. These results are in the same order as that of the results in Table 3.6, in terms of the tradeoffs between boundary ratio and deletion rate.

Threshold	*Bdy. Per Sp. Change* ($x10^{-4}$)	*deletions(%)*	*ratio*	*error(ms.)*
0.0	0.0	8.17	1.744	8.45
0.0	5.0	1.49	2.697	6.06
0.0	10.0	0.40	3.898	4.85
0.0	15.0	0.23	4.788	4.28
15.0	0.0	9.27	1.559	9.07
15.0	5.0	1.60	2.581	6.25
15.0	12.0	0.38	4.229	4.64
15.0	18.0	0.21	5.158	4.09

Table 3.7: Adding Constant Boundaries Per Spectral Change To Second Derivative Threshold

Minimum Distance Constraint

One obvious problem with adding new boundaries by imposing a constant boundaries per spectral change condition is that there will be a lot of segments that will have a huge concentration of spectral change within some small duration of time, say 25 to 30 ms. After adding new boundaries, there will be too many boundaries proposed in this small segment, by the constant boundaries per spectral change criterion. This often is very unrealistic, since even the shortest phones take up a couple of milliseconds. Hence, a minimum distance between boundaries criteria is proposed at this point, with the belief that this will lessen the number of boundaries without affecting the deletion rate too much. Table 3.8 shows the results of adding the minimum distance constraint on the results of Table 3.6, with the spectral change threshold set at 175.0. Table 3.9 shows the effects on putting minimum boundary distance criteria after imposing a threshold on the second derivative of 10.0.

The tradeoff improved somewhat after imposing the minimum boundary condition to add new boundaries to the ones already proposed by putting a threshold on the spectral change, by comparing Tables 3.6 and 3.8. It worsened somewhat if the initial

Minimum Distance (ms)	*Bdy. Per Sp. Change* ($\times 10^{-4}$)	*deletions(%)*	*ratio*	*error(ms)*
8	16.0	0.24	4.714	4.45
8	14.0	0.29	4.402	4.63
9	16.0	0.25	4.661	4.56
9	14.5	0.32	4.428	4.70

Table 3.8: Adding Minimum Distance Criterion to Spectral Change Threshold of 175.0 and Constant Boundaries

Minimum Distance (ms)	*Bdy. Per Sp. Change* ($x10^{-4}$)	*deletions(%)*	*ratio*	*error(ms)*
8	16.0	0.25	4.831	4.36
8	14.0	0.27	4.530	4.55
9	16.0	0.27	4.775	4.47
9	14.5	0.31	4.475	4.66

Table 3.9: Adding Minimum Distance Criterion to Second Derivative Threshold of 10.0 and Constant Boundaries

boundaries were chosen by imposing a threshold on the second derivative.

3.5 Selection

From the results of the previous section, the objective now is to pick a set of criteria which will be used in the orthographic alignment.

Comparing the different results of using different set of parameters in the previous section, the tradeoff seems to be the best when a minimum distance constraint is used to add new boundaries to the ones already created when a threshold on the spectral change is used. Hence, the choice of parameters is narrowed down to the set of parameters listed in Table 3.8.

There is clearly no best set of parameters when comparing the entries in this table.

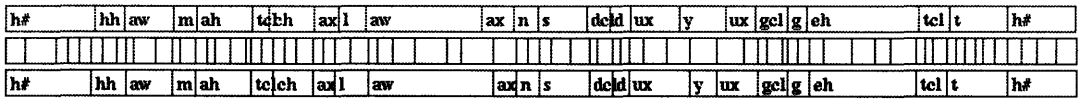

Figure 3.2: Example of Boundary Generation Technique
The top transcription is the correct phonetic transcription of the utterance "How much allowance do you get?" shown in Figure 3.1. The middle transcription is the result of applying the boundary generation procedure using the optimal parameters chosen. The last transcription is the boundary transcription structure described in Section 2.3.1 formed by aligning the top two transcriptions in the figure using a closest boundary criterion.

All of them have low deletion rates, as well as low boundary ratios and absolute errors, and the numbers don't really differ much. In any case, the second entry, with a minimum boundary distance of 8 ms, in addition to a spectral change threshold of 175.0 and additional boundary rate of 14 ms per spectral change, was chosen to be the set of parameters of the boundary selection algorithm. Figure 3.2 shows the result of the boundary selection process when applied to a waveform of the utterance "How much allowance do you get?" The middle transcription is the set of boundaries generated from the waveform having the top transcription as its correct phonetic transcription.

Chapter 4

Network Creation

In this chapter, the process of generating a network of all possible phones given the orthographic transcription is described. Each word in the transcription is transformed into its underlying phoneme sequence through the use of an on-line dictionary. This dictionary is composed of the 6,256 words which make up the TIMIT lexicon [13].

4.1 Phonological Variations

The need for generating a phone network comes from the fact that phonemes are abstract linguistic units. Because of the coarticulation that may take place when different phonemes are produced one after another, the manner in which the phonemes are generated may change. Different phonemes can be realized in different ways, and phones are the realizations of the phonemes. For instance, the phoneme /t/ may be released as in the word "table", or it may be spoken with just a flap of the tongue as in "butter". All these possible realizations of phoneme sequences are embodied in a set of rules which is described in the next section.

Figure 4.1 illustrates the various steps in network generation. The baseform pronunciations are obtained through dictionary lookup. The phonemes of the words

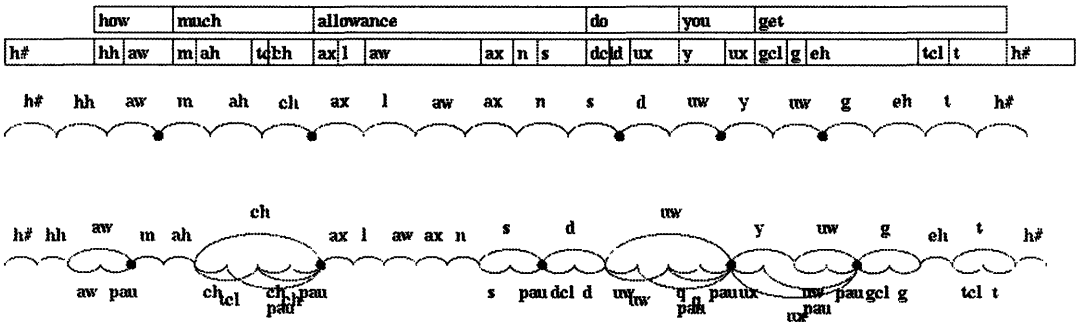

Figure 4.1: Network Generation

The orthographic transcription of the utterance "How much allowance do you get?" is shown on top. Below it is its correct phonetic transcription. Then, a simple network consisting of the phonemes from the dictionary lookup of the orthography is shown. Finally, the whole network of possible phone realizations of the phonemes is depicted, after applying the phonological rules.

are then concatenated together to form a simple network of one path from the first phoneme of the first word to the last phoneme of the last word. This network is represented by arcs and nodes, with the arcs denoting phones and the nodes denoting (possible) phone boundaries. Since there usually are pauses or silence between words, optional pauses are added in the form of additional arcs between words, as well as in the beginning and at the end of the utterance. The first network in Figure 4.1 is an example of the simple network. From this simple network a more complicated one is created after the application of the rules. In the figure, the second network is created from the first one by this process. Later on, the speech signal will be divided into smaller observations in several ways, and the network will be aligned with the observations. The path in the network which aligns best with the observations probabilistically according to Equation 2.2 is then chosen to be the correct one.

4.2 Rules

The set of rules create a network of phones from the phoneme sequence. These rules are meant only to add arcs and nodes to the network, not to replace existing arcs. It is the objective of the alignment algorithm to choose among all the possible paths. A set of fifty-seven rules was used in this transformation, and they can be grouped in the following categories. They are summarized in Appendix B, and applied in the order they appear.

4.2.1 Gemination

When two phonemes with the same identity occur in sequence (usually at a word boundary), there is typically only one phone realized. This rule combines the same phonemes adjacent to one another in the network into one phone. As an example, in Figure 1.1, the phoneme /s/ at the end of the word "plus" and at the beginning of the word "seven" is realized as a single phone in the transcription.

4.2.2 Palatalization

In American English the palatal feature can often be assimilated by a preceding alveolar consonant. Thus, for example, a /sš/ sequence will often be realized as a [š], as shown in Figure 4.2 for the words "gas shortage". Also shown in the figure is the realization of the words "did you", where the palatal feature of the /y/ has spread into the preceding /d/. The result is an affricate [ǰ]. Palatalization can also be produced by /ž/ but this phoneme occurs much more rarely.

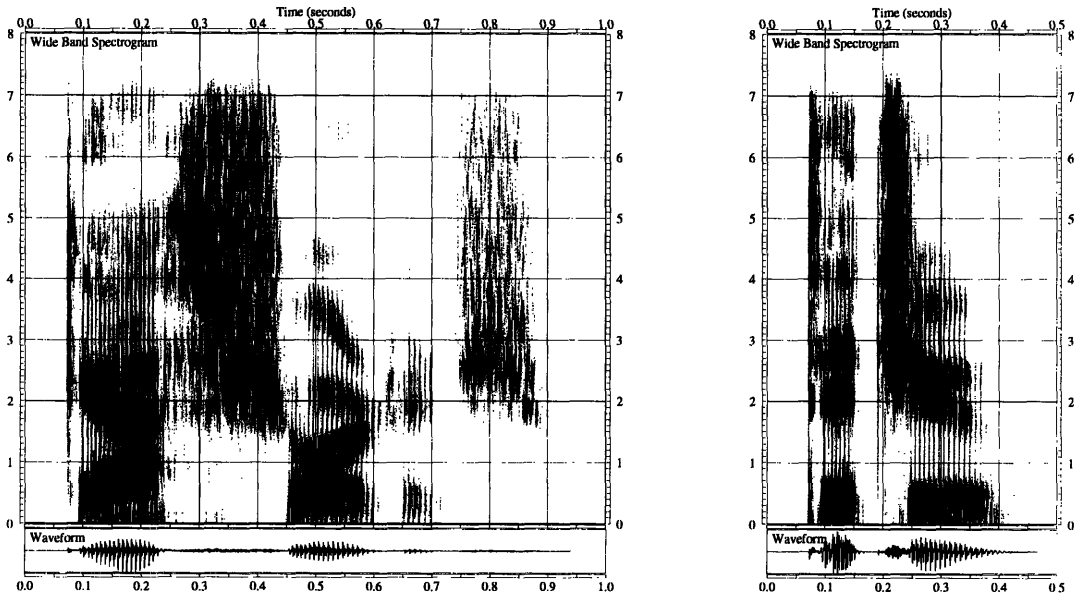

Figure 4.2: Examples of Palatalization

Spectrogram of the words "gas shortage" and "did you." The underlying /sš/ sequence in "gas shortage" is mostly palatalized as a [š] at 0.3 s. In "did you", the /dy/ sequence is realized as a [ǰ].

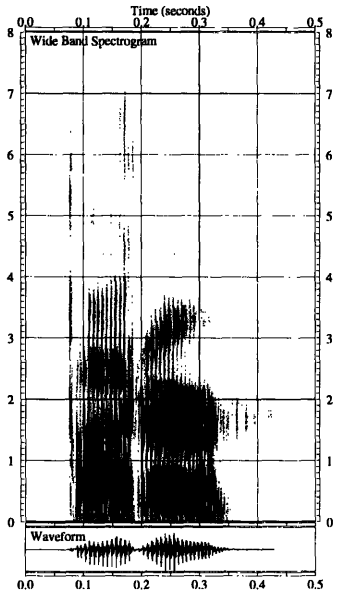

Figure 4.3: Example of Flapping
Spectrogram of the word "butter." The underlying /t/ is realized as a flap [ɾ] at 0.18 s.

4.2.3 Flapping

When alveolar stops or nasal consonants are preceded and followed by a vowel and the following vowel is unstressed, the realization of the consonant can be reduced. In the word "butter," for instance, the phoneme /t/ is not released, but is instead produced by simply flapping the tongue with the upper cavity of the mouth. Fittingly, such a phone is called a flap and is denoted [ɾ]. Figure 4.3 shows a spectrogram of the word "butter" uttered by a male speaker. Notice that at time 0.18 s the phoneme /t/ is realized as a flap. Likewise, a nasal flap occurs occasionally when an /n/ is both followed and preceded by a vowel, such as in the word "winner."

4.2.4 Syllabic Consonants

Syllabic consonants typically occur in reduced syllables. When a schwa, /ə/ occurs just before an /n/, the two phonemes may combine into a syllabic n, [n̩], as in the last syllable of the word "button." The same is true for the phoneme /ə/ preceding the phonemes /m/, /ŋ/, or the glide /l/, as in the word "bottle."

4.2.5 Homorganic Nasal Stop

In a nasal stop consonant sequence, the nasal will often assimilate the place of articulation of the stop. This phonological rule applies to almost all words in a vocabulary (e.g. "bank"), and is captured directly in the dictionary baseform pronunciations. However, it can also occur between words as well. For example, in the /nk/ sequence in the words "one carat," the phoneme /n/ could possibly be realized as the phone [ŋ] by some speakers.

4.2.6 Fronting

In American English, the vowel /u/ can be fronted when it occurs in an alveolar context. As shown in Figure 4.4, which contrasts the words "boom" and "dune", the effect produces a higher than normal second formant. The vowel /u/ is also normally fronted when it occurs in /yu/ sequences as in the word "tuesday".

4.2.7 Voicing

This rule applies to the change in the voicing state of a phone in the context of the phones preceding and following it. The two most common examples are the realization of the phoneme /h/ as a voiced h, [ɦ], when it's followed and preceded by vowels,

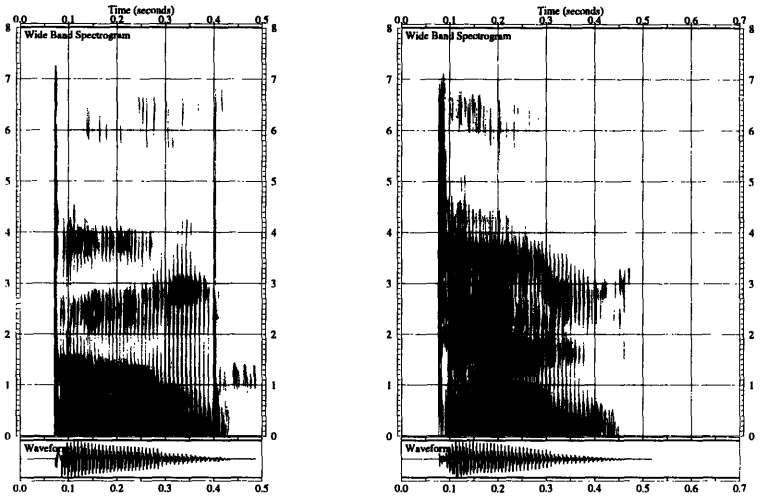

Figure 4.4: A Contrast of a Fronted and a Non-Fronted [u].
Spectrogram of the words "boom" and "dune." The /u/ is not fronted in the word "boom," but is fronted in the word "dune", due to the surrounding alveolar consonants.

and the devoicing of a schwa in the context of two unvoiced consonants. The schwa between the [s] and [č] in the word "Massachusetts" is an example of the latter. When such a word is uttered, this schwa is often reduced and not heard at all, which mainly due to its devoicing.

4.2.8 Epenthetic Silence and Glottal Stops

Epenthetic silences arise due to a mistiming of the articulators during speech production. During the production of the /s n/ sequence in the word "snow" for example, there is often a complete closure in the oral tract before the velum is lowered. The resulting short interval of silence is called epenthetic silence. This effect can also occur for /m/ or /l/, as in the words "small", and "sleep".

Glottal stops often occur at the onset of voicing of a vowel when it is not preceded

by a consonant. They can also be used to mark word boundaries comprised of a vowel vowel sequence as in the case of "to all".

4.2.9 Aspiration

When an utterance begins with a vowel or a /w/, an alternative to the glottal stop is often a short period of aspiration. In addition, many speakers produce "wh" words with an initial [h].

4.2.10 Stop Closures

The rules that cover stop sequences are applied most often. The basic rule is to substitute a stop with its stop closure-stop sequence. However, there are quite a few exceptions. They can divided into several subcases.

Affricates

Affricates are produced by making a complete closure in the oral tract. The nature of the stop closure depends on the affricate it precedes. By convention, a [t□] is used to represent the closure of a [č] and [d□] the closure of a [ǰ].

Stop-Fricative Sequences

In a stop-fricative sequence, the stop released is mixed with the frication noise of the following fricative. Thus, there is no reliable acoustic landmark to locate. The sequence can be in the form of unvoiced stop consonants followed by unvoiced fricatives, or of voiced stops consonants followed by voiced fricatives. They are summarized in Appendix B.

Stop-Stop Sequences

When two stops occur one after the other, it is seldom the case that both are released. A situation similar to that of stop-fricative sequences occur. The first stop is optionally not released and is replaced by its corresponding stop closure.

Chapter 5

Alignment Procedure and Evaluation

In this chapter the alignment procedure is described. Different search methods are explored and their results are compared. Several evaluation criteria are used to perform the comparison, and are described in Section 5.2.1.

5.1 Search

In this section the search process is described in greater detail. A probabilistic framework was presented in Section 2.1 and will be the basis of the search. The models employed in the framework depend on the speech observations used. Separate mean vectors and covariance matrices have to be trained for a frame-based and a microsegment-based approach.

The dynamic time-warping (DTW) algorithm [19] is used to perform the search process throughout. The algorithm finds the best path in the network incorporating both the acoustic and durational scores. The framework in Section 2.1, however, only accounts for the acoustic component of the search. A durational component shall be

added in the maximization process. This is accomplished in two different ways–the observation-based search and the full segment search. They are discussed in turn below.

5.1.1 Observation-based Search

The observation-based search are based solely on the underlying speech observations, which in this research are either the frames or the microsegments. It is inherently different from the full segment search in that it does not hypothesize whole phoneme segments to begin with. Instead, it treats the observations independently and decides which phone the observation is most likely a part of.

A two-pass strategy is employed in the search. A schematic diagram for this method is shown in Figure 5.1. The outer box in the figure will represents the "Alignment Algorithm" component in Figure 1.2. Both passes are modified versions of the dynamic time-warping algorithm. In the first pass, only acoustic scores are taken into account. The output of the first pass is the best path of the network and an alignment of this path with the boundaries, i.e. a phonetic alignment and an orthographic alignment are achieved. For a more detailed description of this method, refer to Appendix C.1.

The second pass takes the best path from the first-pass and performs an independent alignment of this path with the boundaries again. Both acoustic and durational models are used here. Again, refer to Appendix C.1 for a more precise description of this method. The output of the second pass is the final alignment of the phones from the best path with the boundaries. The time complexity of the whole two-pass process is $O(N^2)$.

An example of an alignment using this two-pass algorithm is depicted in Figure 5.2. The phonetic and orthographic transcriptions are the first phonetic and orthographic

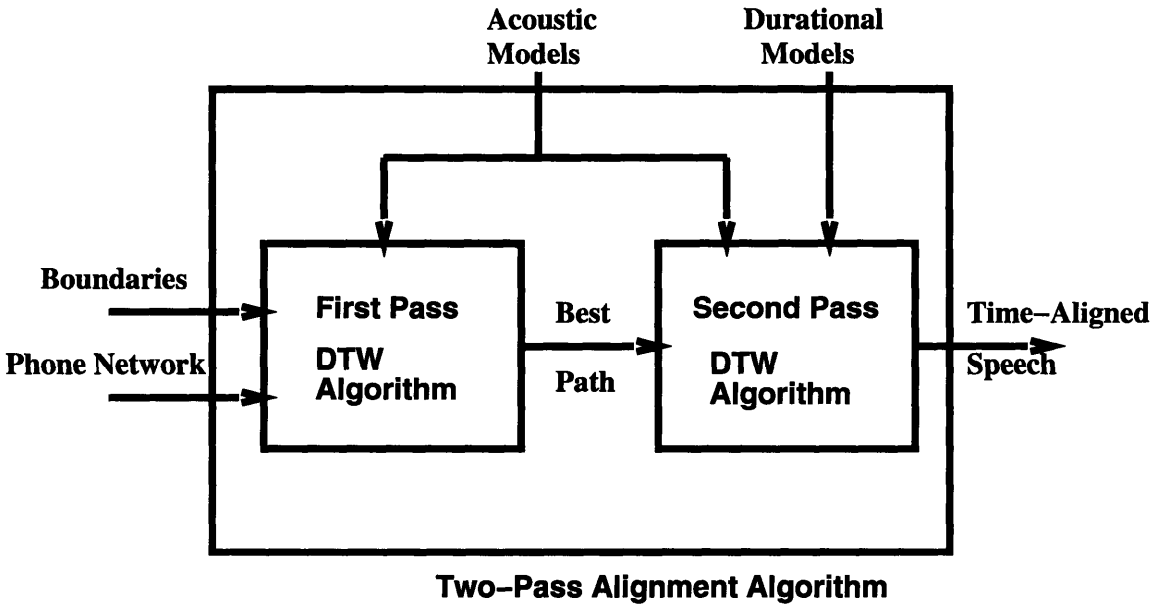

Figure 5.1: Schematic Diagram for the Observation-Based Method
The observation-based search employs a two-pass system to perform the alignment of the boundaries with the phone network. The durational models are used only in the second pass, while the acoustic models are used in both.

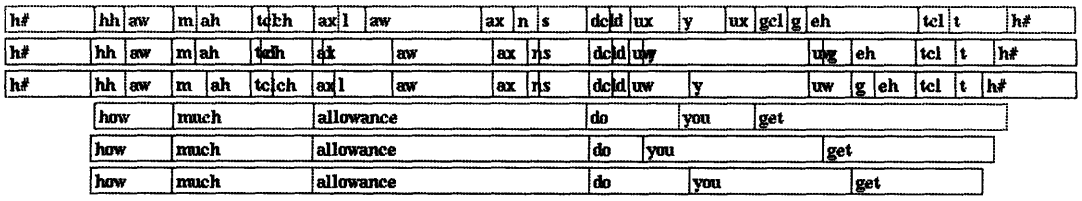

Figure 5.2: Example of Two-Pass Alignment Technique
The top transcription is the correct phonetic transcription of the utterance "How much allowance do you get?" The second phonetic transcription is the result of the first pass alignment. The final phonetic transcription is the result of applying the second pass to the labels of the second phonetic transcription. The corresponding orthographic transcriptions are shown below the phonetic transcriptions.

transcriptions shown, respectively. The next ones correspond to the result of the first pass algorithm matching the phone network with the boundaries. The final ones correspond to the results of the second pass. Note that the second pass improved the word boundary between the words "do" and "you," but not the word boundary between the words "you" and "get." These phenomena occur because of two reasons. First, as can be seen at time 1.0 s in the spectrogram of the utterance in Figure 3.1, the underlying /g/ is poorly articulated. For this reason, the process did not propose a [gʰ], and, since the phonetic segment was a poor match with a normal [g] release, the alignment suffered when compared with the "correct" transcription. Secondly, the durational information for the phone [u] is incorporated in the second pass, which lengthened the duration of the phone in both the words "do" and "you."

5.1.2 Full Segment Search

The full segment search incorporates the acoustic and durational scoring into one search process. Unlike the two-pass process presented above, this method is phoneme-based in the sense that the DTW algorithm matches whole phone segments to the arcs in the network. Refer to Appendix C.2 for a full description of this method. This procedure still has time complexity $O(N^2)$, but because of the more complicated nature of the search relative to the observation-based one, the running time is a lot longer.

5.2 Alignment

In this section, the whole alignment algorithm is evaluated. The results of the observation-based two-pass search as well as the full segment search are examined, and they are discussed in turn below.

5.2.1 Evaluation Criteria

There are two main criteria for evaluating phonetic and word level alignment. The first is the average absolute error between the phone (or word) boundaries in the correct transcription and those in the transcription that results from the algorithm. The second measure involve the amount of overlap between corresponding segments of speech. Let N_S be the total number of phone segments (or words) in the test set, $O_i, 1 \leq i \leq N_S$, be the duration of overlap of phone (or word) i between the corresponding segments in the correct and output phonetic transcription, and $D_i, 1 \leq i \leq N_S$, be the duration of phone (or word) i in the correct transcription. Then overlap measure is given by

$$Overlap = \frac{\sum_{i=1}^{i=N_S} O_i}{\sum_{i=1}^{i=N_S} D_i} \tag{5.1}$$

This value will always be between 0 and 1 since $O_i \leq D_i$ for all i.

5.2.2 Two-Pass Evaluation

Several stages of the two-pass strategy is examined. The first to be explored is the accuracy of the boundary generation algorithm. Such procedure is similar to that of the generation of the boundary transcription structure in Section 2.3. Each of the phone boundaries in the correct phonetic transcription provided by the TIMIT database is matched to the proposed boundaries using a closest time criteria. Hence, a segment transcription structure described in Section 2.3 is created, and is assumed to be the output transcription. The two evaluation criteria described above are then measured, which would give a rough measure of the boundary accuracy.

The second evaluation measures the effectiveness of the search process. Here, the correct phonetic labels are given, but of course not their correct boundaries, and a simple network from the first label to the last label is created. Each stage of the two-pass strategy is then evaluated. The two criteria are measured on both the outputs of the first and second pass.

The final evaluation measures the quality of the whole alignment process which includes network generation. The only additional input is the orthography. The results of both stages are again evaluated by both measures.

The overall error of the system can be thought of as a sum of the errors produced by the boundaries, by the acoustic modelling, by the network generation, and by the search. As described earlier, the microsegment approach has advantages over the frame-based approach in the acoustic modelling and search categories, but obviously will be worse in the boundary generation category, since a frame-based technique will produce no errors here. The hope is that the difference in this category is small enough to be overcome by microsegment approach's advantages in the others.

Frame-Based Evaluation

The first set of alignment evaluations are on the frame-based approach. The main parameters here are delta (δ) and the durational scale γ. There is only one set of averages that can be computed per frame, and δ was allowed to vary only from 0 to 1. The first pass does not involve the durational scale factor, since only the acoustic score is incorporated. The results are summarized in Table 5.1.

An overlap ratio of 82.8% and an average absolute error of 16.7 ms were achieved on the TIMIT database when δ was set to 0 and the correct labels were given. This worsened when δ was changed to 1. These reflect the search procedure. When only the words were given, a two-pass system as described is used, and the evaluations are

Database	*Delta(δ)*	*Given Labels*		*After 1st Pass*	
		Overlap Ratio(%)	*Absolute Error(ms)*	*Overlap Ratio(%)*	*Absolute Error(ms)*
TIMIT	0	82.8	16.7	77.1	20.1
TIMIT	1	82.2	17.2	76.9	19.7
NTIMIT	0	74.6	30.0	67.8	36.2
NTIMIT	1	73.9	30.8	67.5	35.7

Table 5.1: Frame Alignment Phone-Level Ideal and First Pass Results

made after each pass. After the initial pass, the measures slightly improved as delta was switched from 0 to 1, mainly due to the increased information brought about by the delta MFCC parameters. After the second pass, the results of the overall system for different durational scale factor γ are tabulated in Table 5.2.

The best TIMIT results were when $\gamma = 5$ with $\delta = 0$, where the overlap ratio was 79.8% and the average absolute ratio was 15.9 ms. For the NTIMIT corpus, the same parameters achieved the best results, with an overlap ratio of 74.3% and an error of 22.2 ms. The word level evaluation using the same parameters as above are summarized in Tables 5.3 and 5.4.

Similar results were achieved at the word level. Both measures worsened as the delta parameters were added. Moreover, the best results after the second pass were achieved with $\gamma = 5$ and $\delta = 0$, both for the TIMIT and NTIMIT corpora. For the TIMIT corpus, a word overlap ratio of 92.3% and an absolute error of 23.7 ms were achieved with these parameters.

Looking at Tables 5.1, 5.2, 5.3 and 5.4, an interesting observation is that the alignment process did not seem to improve much when parameters that helped the classification process were used. In some cases the results actually worsened, and when they improved, it wasn't by much. For instance, in evaluating frame classification,

Database	*Delta(δ)*	*Durational Scale(γ)*	*After 2nd Pass*	
			Overlap Ratio(%)	*Absolute Error(ms)*
TIMIT	0	0	76.8	20.9
TIMIT	1	0	76.8	19.6
TIMIT	0	5	79.8	15.9
TIMIT	1	5	78.8	16.8
TIMIT	0	10	78.7	17.0
TIMIT	1	10	78.6	16.9
TIMIT	0	15	76.9	19.0
TIMIT	1	15	77.8	17.7
TIMIT	0	20	74.7	22.2
TIMIT	1	20	76.9	18.8
NTIMIT	0	0	67.6	36.2
NTIMIT	1	0	67.3	35.8
NTIMIT	0	5	74.3	22.2
NTIMIT	1	5	72.0	25.6
NTIMIT	0	15	68.7	31.3
NTIMIT	1	15	69.2	30.3
NTIMIT	0	20	66.1	35.8
NTIMIT	1	20	67.4	33.8

Table 5.2: Frame Alignment Phone-Level Second Pass Results

Database	*Delta(δ)*	*After 1st Pass*	
		Overlap Ratio(%)	*Absolute Error(ms)*
TIMIT	0	90.4	29.4
TIMIT	1	90.2	29.3
NTIMIT	0	83.3	51.3
NTIMIT	1	85.8	49.1

Table 5.3: Frame Alignment Word-Level First Pass Results

Database	*Delta(δ)*	*Durational Scale(γ)*	*After 2nd Pass*	
			Overlap Ratio(%)	*Absolute Error(ms)*
TIMIT	0	0	90.2	30.2
TIMIT	1	0	90.2	29.2
TIMIT	0	5	92.3	23.7
TIMIT	1	5	91.6	25.5
TIMIT	0	10	92.3	24.6
TIMIT	1	10	91.7	25.7
TIMIT	0	15	91.9	27.0
TIMIT	1	15	91.5	26.7
TIMIT	0	20	91.3	29.7
TIMIT	1	20	91.3	27.9
TIMIT	0	30	89.7	36.5
NTIMIT	0	0	83.0	51.5
NTIMIT	1	0	85.7	49.2
NTIMIT	0	5	88.9	34.2
NTIMIT	1	5	89.2	37.8
NTIMIT	0	15	87.3	42.4
NTIMIT	1	15	88.7	42.3
NTIMIT	0	20	86.4	47.7
NTIMIT	1	20	88.0	45.8

Table 5.4: Frame Alignment Word-Level Second Pass Results

the classification rate worsened when δ was switched from 0 to 1, but in Table 5.1, swtiching δ from 0 to 1 did not help the first-pass results relative to the ideal results in Columns 3 and 4. The same effect are transferred to the word-level results. This leads to the conclusion that the errors caused by the search process dominate the flaws in the alignment procedure.

Microsegment-Based Evaluation

Assuming the boundary selection parameters chosen in Section 3.5, the evaluation of the generation of boundaries of the microsegment approach produced a phone overlap ratio of 95.15%, and an absolute error of 4.1 ms., a number which is less than the frame duration. Since these numbers are close to perfect, it is safe to conclude that the boundary generation method is sound overall.

An evaluation of the other aspects of the microsegment approach is carried out. Different parameters are varied throughout, including δ, the number of averages computed, the number of additional observations on both side M, and γ, the durational scale factor.

The first set of results on the TIMIT database, summarized in Table 5.5, is on the phone evaluation and assumes $M = 0$, and $\gamma = 5.0$.

There are several observations to be made here. Columns 3 and 4 in Table 5.5 show the results of aligning the boundaries with the correct phonetic transcription. First, the increase in the value of δ or the number of averages, while causing an increase in the classification performance in the previous chapter, does not help that much in the two measures of alignment accuracy. A delta value of $\delta = 1$ with one average produces a result of 85.5 % overlap ratio and 12.6 ms error. Increasing the delta value to $\delta = 7$ makes the results in both categories worse. When the number of averages computed per microsegment is three, the results likewise worsened, to 84.5%

Delta(δ)	*Number of Averages*	*Given Labels*		*After 1st Pass*		*After 2nd Pass*	
		Overlap Ratio(%)	*Absolute Error(ms)*	*Overlap Ratio(%)*	*Absolute Error(ms)*	*Overlap Ratio(%)*	*Absolute Error(ms)*
1	1	85.5	12.6	80.3	14.7	81.7	12.9
4	1	85.3	12.7	80.0	14.5	79.7	14.6
7	1	84.8	13.1	79.7	14.9	79.5	14.9
0	3	85.2	12.9	79.9	14.9	79.7	15.0
1	3	84.5	13.6	78.8	15.7	78.6	15.8
2	3	85.0	13.1	79.6	15.0	79.3	15.1
3	3	85.3	12.9	80.0	14.7	79.8	14.8
4	3	85.5	12.6	80.4	14.5	80.2	14.6
5	3	85.7	12.3	80.5	14.3	80.3	14.3
6	3	85.9	12.2	80.6	14.2	80.4	14.2
7	3	85.9	12.2	80.6	14.1	80.3	14.2

Table 5.5: Microsegment Alignment Phone-Level Results

overlap and 13.6 ms absolute error. However, when the delta was increases all the way up to $\delta = 7$, the results improved somewhat to 85.9% overlap and 12.2 ms error. Looking at the classification results using these same parameters, at Tables 2.1 and 2.3, the set of parameter values which gave good classification did not necessarily do better in alignment, and often did worse. Since the results in Columns 3 and 4 reflect the search algorithm, it is safe to conclude that better acoustic modelling does not help the search process in the experiment.

Columns 5 and 6 test primarily the acoustic modelling, together with the search. The best parameter value for classification, in Tables 2.1 and 2.3, are $\delta = 4$ or 7, with one average. They helped the alignment algorithm a little bit, relative to Columns 3 and 4, especially in the absolute error category. However, the improvements are not significant, and the using $\delta = 1$ still provided the best results, primarily for its better search results.

The last two columns test the whole system, including the durational models.

The main result here is the improvement using the first row parameters. Both the error and overlap measures were enhanced significantly, getting very close to the ideal results given in Columns 3 and 4. As for the rest, they stayed the same relative to the first pass results.

Delta(δ)	*Number of Averages*	*After 1st Pass*		*After 2nd Pass*	
		Overlap Ratio(%)	*Absolute Error(ms)*	*Overlap Ratio(%)*	*Absolute Error(ms)*
1	1	92.7	22.5	93.2	20.4
4	1	92.6	22.7	92.6	22.6
7	1	92.5	22.9	92.6	22.8
0	3	92.0	23.3	92.0	23.1
1	3	91.8	25.1	91.9	24.9
2	3	91.9	24.3	91.9	24.1
3	3	92.1	23.3	92.2	23.1
4	3	92.2	22.7	92.3	22.5
5	3	92.3	22.6	92.4	22.3
6	3	92.3	22.7	92.4	22.4
7	3	92.4	22.5	92.4	22.4

Table 5.6: Microsegment Alignment Word-Level Results

The word level evaluation using the same parameters as above are summarized in Table 5.6. The results are pretty much the same as in Table 5.5. The values after the first pass did not vary significantly, whereas a marked improvement was shown when using the parameter values $\delta = 1$ with one microsegment average in the second pass. The best results are achieved when using these parameters, where the overlap ratio was at 93.2 % and the absolute error is 20.4 ms.

When M was increased, the results did not improve much, even though the classification rate was enhanced. For $M = 3$ with a delta value of 1, a 92.5% first-pass word overlap ratio was achieved, with a word error of 24.2 ms. The second pass improved the outcome very slighly, to 92.6% word overlap ratio and 24.0 ms error.

Comparing the corresponding results for the microsegment-based two-pass alignment system and the frame-based two-pass system, we see that the former generally achieved better results. This can be attributed mainly to the former satisfying the independence assumption better, as was first mentioned in Section 3.1. Each observation in the microsegment-based approach is more independent of its adjacent observations, and hence the Equation 2.5 is more accurate.

Another observation is that as with the frame-based approach, the errors due to the search algorithm are the most dominant ones. Using parameters which improved the acoustic modelling (i.e., classification) did not help much in the overall alignment procedure. For the microsegment-based approach, this is evident, for instance, in Table 5.6, where increasing δ, which significantly improved classification, did not help much in the alignment procedure.

5.3 Full-Segment Search Evaluation

In this section the full-segment search method is evaluated. The main purpose of this evaluation is to compare the results with the best ones of the two-pass system. Both methods used the same parameters in performing the alignment. The best two-pass system is the microsegment-based system with $\delta = 1$ and one microsegment average. Hence, in the full-segment evaluations, to calculate the scores of each phone, the microsegments within the phone are used to calculate its score. δ is set to 1 and the duration scale factor γ is allowed to vary. The phone-level results for the TIMIT database are shown in Table 5.7.

The alignment results were progressively worse as γ was increased, both in the evaluations when the correct phone labels are given and when only the words are given. The word-level results are given in Table 5.8, where a similar occurrence happens. The best word-level results were achieved with using $\gamma = 10$, and with

Durational Scale(γ)	*Given Labels*		*Overall System*	
	Overlap Ratio(%)	*Absolute Error(ms)*	*Overlap Ratio(%)*	*Absolute Error(ms)*
10	78.4	21.2	68.5	27.8
20	74.2	25.6	65.1	31.1
30	70.8	30.0	62.5	34.9

Table 5.7: Full-Segment Search Phone-Level Ideal and Overall Results

these parameters an word overlap ratio of 91.5% and an average absolute error of 36.6 ms were achieved.

Durational Scale(γ)	*Overall System*	
	Overlap Ratio(%)	*Absolute Error(ms)*
10	91.5	36.3
20	90.6	39.7
30	89.6	44.4

Table 5.8: Full-Segment Search Word-Level Overall Results

5.4 Summary

Comparing the results of the full-search method with those of the two-pass system, we see that the latter performed the alignment process better. The overlap ratio is comparable to the best results of both the frame- and microsegment-based two-pass systems. However, the average absolute error is much worse. Overall, the best system is the microsegment-based two-pass system with $\delta = 1$ and one average computed per microsegment. This results in a word overlap ratio of 93.2% and word error of 20.4 ms, which corresponds to a phone overlap ratio of 81.7% and a phone error of 12.9 ms.

Chapter 6

Conclusions

6.1 Summary

There are three major results. First, the two-pass system generally performed a more accurate alignment procedure than the full segment search system. The word overlap ratio of the best results of methods are both over ninety percent, and is generally comparable. However, the average word absolute error for the best result of the full search is around 16 ms more than that of the two-pass search. Moreover, the two-pass system is computationally more efficient.

Second, between the two two-pass systems evaluated, the microsegment-based approach worked better than the frame-based approach. This can be attributed mainly to the independence assumption that the former satisfies better.

Finally, the errors in the search algorithm dominated the alignment procedure. An increase in the classification rate did not guarantee an improvement in the alignment algorithm. For instance, the best result was achieved using the two-pass system with the microsegments as the observations. The parameter values that generated this result were $\delta = 1$ with one microsegment average per observation. A breakdown of the errors using this set of parameters is shown in Table 6.1.

Word Level	*Overlap Ratio(%)*	*Absolute Error(ms)*
after 1st pass	92.7	22.5
after 2nd pass	93.2	20.4
Phoneme Level		
using correct boundaries	95.2	4.1
using correct labels	85.5	12.6
after 1st pass	80.3	14.7
after 2nd pass	81.7	12.9

Table 6.1: Full-Segment Search Word-Level Overall Results

We can see from the table that the biggest source of error is when going from being given the correct boundaries to being given the correct labels. This is indeed the search process. The boundary generation, as well as the acoustic and durational modelling, are relatively more sound.

6.2 Comparison To Other Work

Although a lot of speech researchers have worked on the problem of alignment in the past, only Riley and Ljolje [15] have tried to perform orthographic alignment on the TIMIT database. They achieved a phone-level result of 80% of the boundaries within 17 ms of the TIMIT-provided boundary, which roughly corresponds to my phone-level average error of 12.9 ms. The work of Farhat [8] were slightly worse than that of Riley's. They achieved a phone-level result of 75% of the boundaries within 20 ms of the manually-labelled ones. The other relevant work was that of Leung [14]. He achieved a phone-level result of 75% of the proposed boundaries within 10 ms of the correct ones.

6.3 Future Work

From the previous section, the most promising technique is microsegment-based two-pass system. It is therefore natural to explore this technique in future research. Four main sources of improvements are as follows. First, since the search algorithm provided the biggest cause of error, different methods of performing the search process can be explored. Perhaps there is a better way to incorporate both passes into one that can improve the alignment without sacrificing too much computation time.

Second, even though the boundary generation algorithm seems sound, one cannot rule out the possibility that the errors attributed to other sources might actually be caused in part by the boundary creation scheme. After all, these boundaries determine the microsegments which are the basis for all the work after that. Hence, it might be a good idea to explore other ways of generating boundaries.

Third, the acoustic modelling can also be enhanced. Other methods of representing the speech signal can be explored, such as the wavelet function. This takes advantage of the tradeoff in time and frequency, and is most promising in the classification and alignment problems.

Finally, it would be interesting to see how the alignment method would perform on a spontaneous speech corpus such as the SWITCHBOARD corpus. The occurrence of non-lexical items not part of the given orthographic transcription could pose some problems to the alignment procedure outlined above. Additional modelling would have to be incorporated to the existing one to handle disfluencies (e.g., filled pauses, restarts) which do not typically occur in read corpora. For instance, simple inter-word models such as word spotting techniques might be used to account for this variability.

Appendix A

Phone Classes

The forty-two phone classes in IPA and ARPABET forms are shown in Tables A.1 and A.2, respectively.

Category	*Phones*	*Category*	*Phones*	*Category*	*Phones*
Vowels:	[i]	Glides:	[l], [l̥]	Fricatives:	[z]
	[ɪ], [ɨ]		[r]		[š], [ž]
	[ɛ]		[y]		[v]
	[æ]		[w]		[f]
	[ə], [ʌ], [ə^{h}]	Voiced	[b$^{\square}$], [d$^{\square}$],		[ð]
	[u], [ü]	Closure:	[g$^{\square}$]		[θ]
	[ʊ]	Unvoiced	[p$^{\square}$], [t$^{\square}$],		[s]
	[ɔ], [ɑ]	Closure:	[k$^{\square}$]	Nasals:	[m], [m̥]
	[e]	Voiced	[b]		[n], [n̥], [ɾ̃]
	[ɑ^{y}]	Stops:	[d]		[ŋ], [ŋ̊]
	[ɔ^{y}]		[ɾ]	Affricates:	[č]
	[ɑ^{w}]		[g]		[ǰ]
	[o]	Unvoiced	[p]	External Sil.:	[h#],[h#1],[h#2]
	[ɝ], [ə^{l}]	Stops:	[t]	Internal	[$\square$],[$\square$],
Aspiration:	[h], [ɦ]		[k]	Silence:	[ʔ]

Table A.1: Forty-Two Phone Classes in IPA form

Category	*Phones*	*Category*	*Phones*	*Category*	*Phones*
Vowels:	iy	Glides:	l, el	Fricatives:	z
	ih, ix		r		sh, zh
	eh		y		v
	ae		w		f
	ax, ah, axh	Voiced	bcl, dcl,		dh
	uw, ux	Closure:	gcl		th
	uh	Unvoiced	pcl, tcl,		s
	ao, aa	Closure:	kcl	Nasals:	m, em
	ey	Voiced	b		n, en, nx
	ay	Stops:	d		ng, eng
	oy		dx	Affricates:	ch
	aw		g		jh
	ow	Unvoiced	p	External Sil.:	h#,h#1,h#2
	er, axr	Stops:	t	Internal	epi,pau
Aspiration:	hh, hv		k	Silence:	q

Table A.2: Forty-Two Phone Classes in ARPABET form

Appendix B

Phonological Rules

Listed below in the table are the fifty-seven phonological rules.

Rule Heading	*Left Context*	*Present Phone*	*Right Context*	*Possible Phone Subs.*
Gemination:		[n] [n]		[n]
		[m] [m]		[m]
		[ŋ] [ŋ]		[ŋ]
		[l] [l]		[l]
		[f] [f]		[f]
		[v] [v]		[v]
		[θ] [θ]		[θ]
		[ð] [ð]		[ð]
		[s] [s]		[s]
		[z] [z]		[z]
		[š] [š]		[š]
		[ž] [ž]		[ž]
Palatalization:		[s] [y]		[š]
		[t] [y]		[č]
		[d] [y]		[j]
		AF	[š]	[š]
		AF	[ž]	[ž]

Rule Heading	*Left Context*	*Present Phone*	*Right Context*	*Possible Phone Subs.*
Flapping:	VOWEL	AS	VOWEL	[ɾ]
	VOWEL	[n]	VOWEL	[ɾ̃]
Syllabic		SCHWA [n]		[n̩]
Consonants:		SCHWA [m]		[m̩]
		SCHWA [ŋ]		[ŋ̩]
		SCHWA [l]		[l̩]
		[r] SCHWA		[ə^{ɹ}]
		SCHWA [r]		[ə^{ɹ}]
Homorganic		NVN	VS	[ŋ]
Nasal		NLN	LS	[m]
Stop:		NAN	AS	[n]
Fronting:		[y] [u]		[ü]
	ALVEOLAR	[u]	ALVEOLAR	[ü]
		Voicing		
	VOICED	[h]	VOICED	[ɦ]
	UNVOICED	SCHWA	UNVOICED	[ə^{h}]
Epenthetic	[s]		NASAL-OR-L	[▯]
Silence:				
Aspiration:	PAU #		[w]	[h]
Glottal	PAU #		VOWEL	[ʔ]
Stops	VOWEL #		VOWEL	[ʔ]
Stop			[č]	[t$^{▫}$]
Closures			[ǰ]	[d$^{▫}$]
		[t]	[s]	[t$^{▫}$]
		[d]	[z]	[d$^{▫}$]
		[k]	[s]	[k$^{▫}$]
		[g]	[z]	[g$^{▫}$]
		[p]	[s]	[p$^{▫}$]
		[b]	[z]	[b$^{▫}$]
		[b]	STOP	[b$^{▫}$]
		[d]	STOP	[d$^{▫}$]
		[g]	STOP	[b$^{▫}$]
		[p]	STOP	[p$^{▫}$]
		[t]	STOP	[t$^{▫}$]
		[k]	STOP	[k$^{▫}$]
		[b]		[b$^{▫}$] [b]
		[d]		[d$^{▫}$] [d]
		[g]		[g$^{▫}$] [g]
		[p]		[p$^{▫}$] [p]
		[t]		[t$^{▫}$] [t]
		[k]		[k$^{▫}$] [k]

These rules are used to generate the network from the orthographic transcription. The code for some of the abbreviations used in the table are as follows:

1. ALVEOLAR - [s] [z] [t] [d] [n] [ɳ]
2. LS (labial stop) - [b] [p]
3. AS (alveolar stop) - [d] [t]
4. VS (velar stop) - [g] [k]
5. UNVOICED - [p] [t] [k] [s] [č] [θ] [f] [š] [h]
6. NASAL - [m] [n] [ŋ]
7. NVN (non-velar nasal) - [m] [n]
8. NLN (non-labial nasal) - [ŋ] [n]
9. NAN (non-alveolar nasal) - [m] [ŋ]
10. SCHWA - [ə] [ɨ]
11. AF (alveolar fricative) - [s] [z]
12. NASAL-OR-L - [m] [n] [ŋ] [l]
13. PAU - [ʔ] [□] [h#]
14. # - word boundary

Appendix C

Alignment Algorithms

The two modified DTW algorithms are described in greater detail in this section. Both algorithms take as input the phone network and the acoustic boundaries, and output the time-aligned speech signal.

C.1 Two-Pass Observation-Based Method

The first method is the two-pass method. A schematic is shown in Figure 5.1. In the first pass, only acoustic scores are taken into account. The scores are calculated by means of Equation 2.8. The DTW algorithm treats the boundaries from the boundary generation algorithm on one axis and the arcs of the network which represent the phones on another. A score matching each observation and each arc in the network is computed and stored. Then, given a ith arc and the jth boundary, the algorithm determines the best path through the network that ends with the matching of the observation whose right boundary is the ith arc and the jth boundary. The best path through the network will be the one that is determined at the final step, when the the last boundary and the last arc are matched. Assuming that there is a limit on the number of arcs that arrive at any given node, this algorithm has a time

complexity of $O(N^2)$. The output of the algorithm is the best path of the network and an alignment of this path with the boundaries, i.e. a phonetic alignment and an orthographic alignment are achieved.

The second pass takes this best path and performs an independent alignment of this path with the boundaries again, this time taking into account durational scores. It throws away any score or matching derived from the first pass. Once again, the DTW algorithm for the second pass is observation-based, having the phones in the best path on one axis and the boundaries on the other. However, since durational scores are phoneme-based, a conjecture is made at each step on how long the present segment will be. Given the ith phone and the jth boundary, the best matching between the observation O whose boundaries are the jth and (j+1)th and the ith phone is determined by the following method. There are two possible choices at this point. There could be a phone transition, where a matching occurs between the observation O_l left of O to the (i-1)th phone, or a non-transition, where a matching occurs between the observation O_l and the ith phone. Let $TotS_{ob}$ be the array of the best scores at each matching of a phone and boundary; hence the values $TotS_{ob}(k,l)$ are known at this point for $k \leq i$ and $l \leq j$ except for $k = i$ and $l = j$, which is to be determined now. This score incorporates all acoustic scores and durational scores up to that point in the path. The scores S_{trans} and $S_{non-trans}$ for each choice, respectively, are likewise composed of an acoustic score and a durational score. The first choice involves a transition of phoneme, and we have:

$$S_{trans} = AS_{ob}(i,j) + \gamma \cdot DS_{ob,trans}, \tag{C.1}$$

where $AS_{ob}(i,j)$ and $DS_{ob,trans}$ are the acoustic component and durational components, respectively, and γ is the durational scaling factor. Experiments are conducted for different values of γ. $AS_{ob}(i,j)$ is computed by means of Equation 2.8, and $DS_{ob,trans}$ is computed using the durational model. The formula for determining the

durational score for a phone α is the following:

$$DS_\alpha = -log(2\pi\sigma) - \frac{1}{2} \cdot ((\delta\alpha - \mu_\alpha)/\sigma_\alpha)^2, \tag{C.2}$$

where μ_α and σ_α are the mean and standard deviations of the duration of the phone α, respectively, and $\delta\alpha$ is the duration of phone α. For $S_{non-trans}$, we have:

$$S_{non-trans} = AS_{ob}(i,j) + \gamma \cdot DS_{ob,non-trans}. \tag{C.3}$$

But since there is no transition, the exact phone duration of the phone i is not known. We can only conjecture at this point. A good guess would be to assume that the duration is an average one, and hence the second part of the RHS of Equation C.2 is assumed zero:

$$DS_{ob,non-trans} = -log(2\pi\sigma). \tag{C.4}$$

Then S(i,j) can now be calculated by the following rule:

If $S_{trans} + TotS_{ob}(i-1,j-1) \geq S_{non-trans} + TotS_{ob}(i,j-1)$,
$TotS_{ob}(i,j) = S_{trans} + TotS_{ob}(i-1,j-1)$;
else
$TotS_{ob}(i,j) = TotS_{ob}(i,j-1) + AS_{ob}(i,j)$

Note that for the second case only the acoustic score is added to the total score at that point. Durational scores are only added when there is a transition. The time complexity of the second pass is likewise $O(N^2)$, and hence the time complexity of the whole two-pass process is $O(N^2)$. The output of the second pass is the final alignment of the phones from the best path with the boundaries.

C.2 Full Segment Search Method

The full segment search is described in detail in this section. Here, the DTW algorithm matches whole phone segments to the arcs in the network. Given the ith arc which represents phone α and the jth boundary, the best score $TotS_{fs}(i,j)$ at this point is the score of the best path which ends with the phone α terminating at the jth boundary. All the acoustic and durational scores are incorporated into $TotS_{fs}(i,j)$. A maximum number of observations M that a phone can match to is preset. M was chosen to be 150 for a frame-based search and 15 for a microsegment-based search. To describe the full search algorithm, we label the arcs which have the starting node of the ith arc as their end node as a_l, $1 \leq l \leq L$. Now, given some boundary k, j-M $\leq$ k $\leq$ j-1, we look at the best path which matches all the observations between boundaries k and j, to the phone α. The weight WS(i,j,k) of this matching has an acoustic score $AS_{fs}(i,j,k)$ and a durational score $DS_{fs}(i,j,k)$. They are calculated as follows:

$$AS_{fs}(i,j,k) = \sum_{m=k}^{j-1} log \Pr(\vec{\mathbf{x}}_\mathbf{m} \mid \alpha), \tag{C.5}$$

where Equation 2.8 is used to calculate the log of the probability. Equation C.2 is used to compute $DS_{fs}(i,j,k)$, with the time difference between boundaries k and j substituted for $\delta\alpha$. Then,

$$WS(i,j,k) = AS_{fs}(i,j,k) + \gamma \cdot DS_{fs}(i,j,k), \tag{C.6}$$

where γ is the durational scaling factor as before. This weight is added to the quantity $B_{i,j,k}$:

$$B(i,j,k) = \max_l TotS_{fs}(a_l,k), \tag{C.7}$$

which is the best path matching any of the arcs a_l with the boundary k to get the score C(i,j,k) of the best path matching the all the observations between boundaries k and j to the phone α. A final maximization is processed to compute for $TotS_{fs}(i,j)$:

$$TotS_{fs}(i,j) = \max_{j-M \leq k \leq j-1} B(i,j,k). \tag{C.8}$$

The output of this algorithm when i and j have reached their final respective values gives an alignment of the speech signal with the proposed boundaries. Because there is a limit set of the number of arcs that can arrive at a node, as well as on M, the number of observations that can match with a phone, the above procedure still has time complexity $O(N^2)$. If there was no limit on M, which is to say a phone can occupy everything till the first boundary, then the time complexity goes up to $O(N^4)$, which gives a good measure on the complexity of the algorithm.

Appendix D

Train and Test Speakers

D.1 Train Speakers

mwac0 mrwa0 mjai0 fmpg0 mwgr0 msdh0 mkag0 mjfr0 mhit0 mdlc2
fljg0 fkfb0 fcjs0 fbcg1 mmwb0 mmdb0 mjlb0 maeb0 fklc0 fgdp0
mwsb0 mwch0 mtrc0 mtdt0 mmdb1 fsma0 mesj0 mdls0 mdac2 fawf0
msdb0 mjjg0 mjbr0 mfxv0 mesd0 mdbb1 mchh0 mthc0 mrav0 mjsw0
mjlg1 mdrm0 mdns0 mbom0 makb0 ftlh0 mkrg0 mrpc0 mtmt0 mwjg0
mjfh0 mmsm0 mtat0 fexm0 fskp0 mjvw0 mkjl0 mpgl0 mree0 mrws0
mefg0 mrtk0 mprk0 mmwh0 mjls0 mhpg0 mcmj0 fedw0 fcrh0 fcmm0
fmld0 fsgf0 ftbr0 mctt0 mpgh0 mvjh0 mwbt0 fdjh0 fdtd0 fhlm0
fjkl0 fmgd0 fsjg0 mafm0 mdef0 mdvc0 mkdd0 msvs0 faks0 fgwr0
fisb0 fjdm2 fpas0 mbpm0 mhbs0 mklw0 mnjm0 mpar0 mrmb0 mtab0
fbch0 mfgk0 fpaz0 fpls0 mbcg0 mbma1 mdlf0 mejl0 mhrm0 mrem0
mrld0 mrxb0 mses0 mkln0 mmam0 mrdm0 mreh1 ftlg0 fsxa0 fsdj0
fksr0 fhes0 feeh0 fcmr0 fcke0 fdhc0 fbjl0 fasw0 mtrr0 mrcz0
mpdf0 mkls1 mgwt0 meal0 mdwm0 mdwk0 mdld0 mcsh0 fpkt0 fmmh0
fdrd1 mjmp0 milb0 mglb0 mtlb0 fmkf0 mkah0 mdhs0 ftmg0 fskl0
flac0 mtjg0 mtdp0 msat0 mrjh0 mmws0 mmlm0 mkjo0 mkdt0 mjwg0
mjae0 fnkl0 mtwh1 mtpp0 mrhl0 mmxs0 mjxa0 mjrk0 mcdd0 frll0
mteb0 fmah1 fljd0 fdxw0 mrjt0 mrew1 mpmb0 mpgr1 mkdb0 mjma0
mgrl0 mdpb0 fsms1 feme0 fcag0 mrms1 mrjm4 falk0 mmab1 mfer0
flas0 fkms0 fcmh1 mwvw0 mvrw0 mtlc0 mfmc0 mrpc1 mrab1 mnls0
mmaa0 mges0 mdcm0 mbth0 fdac1 mjtc0 mjjm0 mdsj0 mbdg0 majc0
mcef0 fbmh0 flag0 fdas1 fcmh0 mclm0 mdaw1 mdjm0 mjrf0 mmdm2
mrjm3 mnet0 msjk0 mtpr0 mbwm0 mddc0 mrgs0 mrgm0 fdaw0 fdmy0
mmvp0 fsmm0 mgsl0 mwem0 fpmy0 fmjb0 mdhl0 fsjs0 mtqc0 mppc0
mjdm0 mgaw0 mmbs0 fadg0 mjes0 mtpg0 mcxm0 mrcg0 fsdc0 fjmg0

mlns0 mmgk0 mahh0 mjwt0 fscn0 fjcs0 mjth0 faem0 fear0 ftaj0
mejs0 mrlj0 mjmm0 mjjj0 fjsa0 mrml0 mcss0 mmcc0 fecd0 mjbg0
mtjm0 fjlg0 mrbc0 mgrt0 mkam0 fntb0 mkls0 mrtj0 mjsr0 fnlp0
flhd0 flkd0 mplb0 mrjm0 mded0 mtjs0 fsem0 mjln0 mjdc0 mmds0
fjhk0 mmdm0 fsah0 fram1 mcpm0 mdrd0 fceg0 mmgc0 mjac0 mrlr0
fdrw0 fskc0 mpeb0 mprt0 mdrb0 medr0 mjpm0 mljc0 mdac0 fvkb0
mrko0 fmju0 frjb0 mers0 fkaa0 mvlo0 mjkr0 mjxl0 mrfl0 mtcs0
mcth0 mpgr0 mlbc0 feac0 mmjb1 flma0 fjwb1 fcyl0 mkaj0 mtkd0
mpam0 mdlb0 felc0 mtbc0 fjxm0 mprb0 mpam1 mhjb0 mbma0 mbwp0
mctw0 mdlm0 fslb1 mrcs0 mcew0 mdtb0 mesg0 mabc0 fcft0 mmea0
mbsb0 mtrt0 fjas0 fklh0 mtwh0 mmar0 flkm0 mdbb0 mdmt0 mfwk0
fcrz0 mmjr0 mrjs0 fmcm0 mdma0 mdwh0 mmrp0 mdlh0 mdlr0 mtat1
mlll0 mjar0 mmdg0 msem1 fjlr0 mdwa0 mjee0 mdem0 fmjf0 mrjb1
mgsh0 mdcd0 mgrp0 mhmg0 mlel0 mljb0 mrab0 mbjk0 mdss1 fpad0
mkxl0 mtpf0 mpfu0 futb0 mjra0 msas0 mbml0 mkes0 mrsp0 mrjo0
mtdb0 mrmh0 mjhi0 megj0 fhew0 fnmr0 msms0 fcdr1 mmeb0 mgag0
mgar0 fgmd0 fjsp0 mrai0 mwrp0 mrrk0 fjrp1 mmab0 mreb0 mrre0
mpcs0 flmc0 mrcw0 mbgt0 ftbw0 fcmg0 mstk0 msmr0 madc0 mslb0
fmbg0 mlih0 mbjv0 mdps0 mdss0 makr0 marc0 mjpm1 mrpp0 mcrc0
mctm0 mtls0 mmdm1 mwsh0 mlsh0 mewm0 mtkp0 fmem0 mter0 mrjr0
fsag0 mcdc0 mcdr0 mwew0 fkdw0 mstf0 fgjd0 mprd0 fbas0 mrds0
mkch0 mwdk0 mcae0 mchl0 fcjf0 mfrm0 mjdm1 fclt0 mljh0 mdbp0
frng0 mtaa0 mzmb0 fgrw0 mdlc0 mtas0 mjpg0 mrgg0 falr0 mgjc0
mtju0 madd0 mjrg0 fcal1 mcem0 fsrh0 fspm0 fsjk1 fvfb0 mclk0
mjda0 mtas1 mrws1 mjmd0 mmdh0 mrvg0 mrso0 flmk0 mrtc0 mabw0
mjfc0 mrms0 mrlk0 fcau0 fmml0 mpwm0 mdab0 mdsc0 mgak0 mpab0
fjen0 fjlm0 fpjf0 mfxs0 mmag0 mjdg0 mrlj1 mjws0 mwar0 fjsk0
mjrh1 mjrh0 fklc1 fleh0 mjjb0 mrdd0 msah1 mjeb1 mjde0 mhxl0
fblv0 fpaf0 mhmr0 mgmm0 fsls0 mbbr0 mdpk0 mcre0 mklr0 mmpm0
mccs0 mcal0 mtxs0 mwre0 fpab1 flet0 fkde0 fjxp0 fjsj0 fjre0
fgmb0 fgcs0 fapb0 mtmn0 fetb0 mwad0 fltm0 fdml0 fsbk0 mdks0
fcaj0 fmaf0 fbmj0 mntw0 mpsw0 fvmh0 mtml0

D.2 Test Speakers

msjs1 fjem0 mjrp0 mdwd0 mjeb0 mrjm1 mtmr0 fjwb0 fajw0 fdnc0
msfv0 mgjf0 mbef0 mgaf0 mlnt0 mapv0 fdfb0 fsjw0 msrg0 msfh0
mgxp0 mbns0 marw0 msmc0 mkcl0 mroa0 flbw0 fdms0 frew0 fsak0
mklt0 msfh1 mrkm0 mdas0 mcmb0 mram0 fkkh0 flod0 fmah0 mjdh0
flnh0 mkdr0 mdlr1 mbar0 maeo0 mdlc1 fpac0 freh0 mres0 fjrb0

Bibliography

[1] A. Andersson, and H. Broman. Towards Automatic Speech-To-Text Alignment. In *Proceedings of 3rd European Conference on Speech Communication and Technology*, 301–304, September 1993.

[2] B. Angelini, F. Brugnara, D. Falavigna, D. Giuliani, R. Gretter, and M. Omologo. Automatic Segmentation and Labelling of English and Italian Speech Databases. In *Proceedings of 3rd European Conference on Speech Communication and Technology*, 653–656, September 1993.

[3] Mats Blomberg and Rolf Carlson. Labelling of Speech Given Its Text Representation. In *Proceedings of 3rd European Conference on Speech Communication and Technology*, 1775–1778, September 1993.

[4] F. Brugnara, D. Falavigna, and M. Omologo. An HMM-Bases System for Automatic Segmentation and Labelling of Speech. In *Proceedings of the 1992 International Conference on Spoken Language Processing*, 803–806, October 1992.

[5] D. Crystal. *A Dictionary of Linguistics and Phonetics*, 1980.

[6] T.H. Crystal and A.S. House. Characterization and Modelling of Speech-segment Durations. In *Proceedings of the 1986 International Conference on Acoustics, Speech, and Signal Processing*, 2791–2794, April 1986.

[7] P. Dalsgaard, O. Andersen, W. Barry, and R. Jorgensen. On the Use of Acoustic-Phonetic Features in Interactive Labelling of Multi-Lingual Speech Corpora. In *Proceedings of the 1992 International Conference on Acoustics, Speech, and Signal Processing*, 549–552, October 1992.

[8] A. Farhat, G. Perennou, and R. Andre-Obrecht. A Segmental Approach Versus a Centisecond One For Automatic Phonetic Time-Alignment. In *Proceedings of 3rd European Conference on Speech Communication and Technology*, 657–660, September 1993.

[9] S. Fujiwara, Y. Komori, M. Sugiyama. A Phoneme Labelling Workbench using HMM and Spectrogram Reading Knowledge. In *Proceedings of the 1992 International Conference on Spoken Language Processing*, 791–794, October 1992.

[10] J. Glass. *Finding Acoustic Regularities in Speech: Applications to Phonetic Recognition.* Ph.D. thesis, Massachusetts Institute of Technology, May 1988.

[11] C. Jankowski, A. Kalyanswamy, S. Basson, and J. Spitz. "NTIMIT: A phonetically balanced, continuous speech, telephone bandwidth speech database." In *Proceedings of the 1990 International Conference on Acoustics, Speech and Signal Processing*, 109–112, April 1990.

[12] Richard A. Johnson, Dean W. Wichern. *Applied Multivariate Statistical Analysis*, 1982.

[13] L. Lamel, R. Kassel, S. Seneff. Speech Database Development: Design and Development of the Acoustic-Phonetic Corpus. In *Proceedings DARPA Speech Recognition Workshop*, 100–109, 1986.

[14] Hong C. Leung. *A Procedure for Automatic Alignment of Phonetic Transcriptions with Continuous Speech.* Master's thesis, Massachusetts Institute of Technology, January 1985.

[15] A. Ljolje and M. Riley. Automatic Segmentation and Labelling of Speech. In *Proceedings of the 1991 International Conference on Acoustics, Speech, and Signal Processing*, 473–476, May 1991.

[16] A. Ljolje and M. Riley. Automatic Segmentation of Speech for TTS In *Proceedings of 3rd European Conference on Speech Communication and Technology*, 1445–1448, September 1993.

[17] P. Mermelstein and S. Davis. Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. In *Transactions on Acoustics, Speech and Signal Processing*, 1980.

[18] B. Wheatley, G. Doddington, C. Hemphill, J. Godfrey, E. Holliman, J. McDaniel, and D. Fisher. Robust Automatic Time Alignment of Orthographic Transcriptions with Unconstrained Speech. In *Proceedings of the 1992 International Conference on Acoustics, Speech, and Signal Processing*, 533–536, March 1992.

[19] Patrick Henry Winston. *Artificial Intelligence, 3rd Edition*, 1992.