Speech Segmentation and phone classification for regional pronunciation variants

Martin Schickbichler, 0330366

Signal Processing and Speech Communications Laboratory
Graz University of Technology, Austria

SYNVO GmbH
8700 Leoben, Austria

Assessor:
Dr.sc.ETH Harald Romsdorfer

Graz, November 29, 2013

Abstract

Automatic phonetic segmentation and labeling of recorded speech corpora has several applications in natural language processing. Under optimal conditions, state-of-the-art systems achieve an accuracy in this task that can be compared with manual segmentation and labeling. Nevertheless there are many cases in which these optimal conditions are not met. Non-standard pronunciation is one scenario that poses a challenge to segmentation systems. This thesis investigates segmentation and labeling of speech that has non-standard pronunciation. Automatic segmentation of a corpus containing three regional varieties of German is performed. Specific rules are developed to cope with pronunciation variation. Several improvements are added to the standard segmentation framework. Further, the pronunciation of the vowels by six speakers is analyzed. In particular, a method for vowel analysis and classification based on artificial neural networks is proposed and applied to the corpus. A method for integrating the resulting phone classifier into an existing HMM/GMM-based segmentation system is presented and implemented, resulting in a higher segmentation accuracy.

Zusammenfassung

Viele Anwendungen in der Maschinellen Sprachverarbeitung basieren auf der automatischen phonetischen Segmentierung und Annotation von Sprachkorpora. Unter optimalen Bedingungen erreichen Systeme am Stand der Technik für diese Aufgabe heutzutage eine beachtliche Genauigkeit, die vergleichbar mit der manuellen Segmentierung und Annotation ist. Jedoch gibt es viele Fälle, in denen diese optimalen Bedingungen nicht erfüllt sind. Eine vom Standard abweichende Aussprache ist ein Szenario, welches Segmentierungssysteme vor Probleme stellt. Diese Masterarbeit untersucht die Segmentierung und Annotation von Sprache bei vom Standard abweichender Aussprache. Eine automatische Segmentierung eines Korpus mit drei regionalen Sprachvarianten des Deutschen wird durchgeführt. Spezielle Regeln werden entwickelt, um mit der vom Standard abweichenden Aussprache zurechtzukommen und einige dieser Verbesserungen werden im vorhandenen Segmentierungssystem implementiert. Weiters wird die Aussprache von Vokalen der sechs Sprecher analysiert. Konkret wird eine Methode für die Analyse und Klassifikation von Vokalen basierend auf Neuronalen Netzwerken vorgeschlagen und auf den Korpus angewandt. Weiters wird eine Methode, um diesen Lautklassifikator in ein existierendes Segmentierungssystem zu integrieren, vorgestellt und implementiert, wodurch eine Verbesserung der Segmentierungsgenauigkeit erzielt wird.

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

....
(Place, Date) (Signature)

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

....
(Ort, Datum) (Unterschrift)

Acknowledgements

I want to thank my advisor Dipl.-Ing. Dr.sc.ETH Harald Romsdorfer who always found some time when I needed support, even on weekends or during busy periods, who gave me the opportunity to apply my knowledge in practice at the company SYNVO, and who encouraged me to finally finish this thesis even though I found it hard to do so after starting my professional career. My thanks also go to Graz University of Technology, which provided me a good education and, even more important, the ability to learn how to educate myself.

Special thanks go to the Institute of Signal Processing and Speech Communication, which provides, beyond excellent education, their students with professional resources and a workplace in the laboratory to work on their theses and projects. I want to specifically mention Univ.-Prof. Dipl.-Ing. Dr.techn. Gernot Kubin, Ass.-Prof. Mag. Dr.phil. Rudolf Muhr and Dipl.-Ing. Dr.techn. Stefan Petrik. They significally contributed to my interest in speech signal processing and linguistics.

Furthermore I would like to thank my family as well as the Austrian state for supporting me financially, especially during the beginning of my studies. Without this support I probably would not have started any higher education. And finally I thank my close friends who were very supportive during my whole studies.

Contents

Chapter 1

Introduction

Speech is the main form of human communication and can be described on several levels of abstraction. From a physical perspective, speaking causes air pressure changes over time in front of the mouth and produces a sound waveform. The sound waveform can be represented as an acoustic signal, containing the sampled air pressure changes over time. From a linguistic perspective, the acoustic speech signal contains a lot of information. The spoken text, the identity of the speaker, or the pronunciation and intonation of the different words can be derived from this signal – just as humans do during the process of hearing. The spoken text is usually of particular interest. It can also be considered as an abstract level of description of the speech signal. The same text can be realized by many different speech signals, even if spoken by the same speaker.

In natural language processing (NLP), human speech is processed by computers and thus different levels of abstraction are involved. The text is among the highest levels of abstraction of a speech signal. Going down the abstraction hierarchy there is a level of abstraction that is of particular importance in NLP and that deals with the pronunciation of words: the phonetic description. The field of phonetics describes the sounds that occur in the languages of the world [Lad75]. The sounds that compose the smallest significant units of utterances, i.e. that are able to distinguish one word from another in a language, are called *phonemes*. Different acoustic realizations of a phoneme that do not alter the meaning of a word are called *allophones* of that phoneme. The term *phone* is generally used for the smallest phonetically identifiable segmental unit of a speech signal. Each utterance can be phonetically described by a sequence of phones. If this phone sequence consists of phonemes only, it is called *phonological transcription* or *phonemic transcription*. In that case, the transcription describes only the underlying sounds but not the details of their realization. If the transcription accounts for all those details, it is called *systematic phonetic transcription*. As an example for the difference, [Lad75] mentions the words *cat* and *catty*. The first word is pronounced with a voiceless alveolar plosive (/t/) at the end. Despite the relation of the second word to the first one, the consonant near the end changes: it is usually pronounced voiced (as /d/). Thus the two words have the phonological transcriptions /kæt/ and /kæti/ and the phonetic transcriptions [kæt] and [kædi], respectively. For *cat*, the transcriptions are the same whereas they differ for *catty*. The usage of the term *phonetic transcription* is often ambiguous. It is a transcription that accounts for some level of

allophonic detail, in contrast to the *phonological transcription*. If the level of details is high, it is called a *narrow phonetic transcription*, if only few details are accounted for, the term *broad phonetic transcription* is used [Lad75]. In this thesis, when the level of detail is not important, the term *phonetic transcription* will be used. Whenever it is important, one of the aforementioned terms will be used. The common style of writing phonological transcriptions enclosed in slashes (like /kæti/) and phonetic transcriptions in brackets (like [kædi]) will also be followed in this thesis.

The phonetic segmentation of an utterance reveals the position and the identity of its phones. It is a time alignment of the phonetic transcription with the acoustic signal. In this thesis, the term *segmentation* is used for *phonetic segmentation of a speech signal* unless otherwise noted.

The manual segmentation of speech is an elaborate task that becomes quite time-consuming for large corpora. An automatic procedure alleviates the segmentation task. A general advantage of automatic methods for speech segmentation, apart from the cost and time needed for manual segmentation, is that their segment boundaries tend to be more consistent than the ones produced by human labelers.

Speech segmentation has various applications in speech processing, among them the creation of the segmental units that are needed to train acoustic models for automatic speech recognition (ASR). Training with a more accurate segmentation leads to better models and to a higher performance of a speech recognizer that uses these models. Another application is the accurate annotation of corpora that should subsequently be used for unit-selection speech synthesis. In this speech synthesis approach, the synthetic utterances are created by selecting and combining already recorded units in a corpus.

Many automatic procedures for speech segmentation rely on the canonical phonetic transcription of utterances. The canonical phonetic transcription is the standard pronunciation and can be looked up in a pronunciation dictionary for the language, inferred from phonological rules or both. In many cases, however, the actual pronunciation can differ significantly from the canonical phonetic transcription. Examples are spontaneous speech, dialects and regional language variants.

An ideal automatic segmentation and labeling system for regional language variants should be able to derive an automatic phonetic segmentation as accurate as possible, with respect to phone boundaries and phone classification. It should be based on the acoustic signal representation and its orthographic transcription only. It should not be dependent on the canonical transcription and perform an accurate segmentation for each realized pronunciation variant. Ideally, it creates a segmentation even if only few data of a certain speaker of a certain language variant is available and no accurate statistical modeling with the data can be done. Such a system should account for pronunciation variation that is not covered by explicit rules known in advance. If enough data is available, it should be able to derive new pronunciation rules.

In this thesis, research is done on speech segmentation and phone classification. The focus is on aspects that can help improving a segmentation system in a way so that it approaches some features of the described ideal system. The starting point is the existing segmentation system of the company SYNVO that is already able to perform a high-quality segmentation on other resources. It should be adapted to be applicable to a corpus with regional language variants.

The existing system uses a quite limited phone set that is not suitable for the ADABA corpus used in this thesis. The ADABA corpus comes with recordings of three regional variants of German (from Austria, Germany and Switzerland) and very narrow transcriptions, i.e. it uses a comprehensive phone set. The pronunciation rules used by the segmentation system are not optimal for the ADABA corpus and thus should be extended. In parallel to the work on this thesis, I developed an application program with a graphical user interface (GUI) that facilitates the whole segmentation process at the company SYNVO. The application serves as a tool for manual and automatic segmentation of speech corpora. It is used to perform the automatic and manual segmentations done in this thesis.

In addition, a method to extract articulatory features for phones is applied. Especially for the vowels, the extracted features are analyzed in detail. A research question is whether specific aspects of regional pronunciation variation can be detected with this method. It is also investigated whether vowel classification can be improved by using the machine learning technique *multilayer perceptrons* (MLPs) that learns the *articulatory features* tongue position and lip rounding. Ideas on how these features can help to improve a segmentation and labeling system are given. The features are compared to the well-known formants. Phone classification with this method is compared to a classification based on hidden Markov models (HMMs) and Gaussian mixture models (GMMs). A combination of the HMM/GMM forced-alignment in the original segmentation system with the MLP-based phone classifier is implemented and evaluated.

The remainder of this thesis is organized as follows: Chapter 2 gives the motivation for and some background information relevant to this thesis. In chapter 3, a review of the literature of some existing systems is done. Chapter 4 introduces the used speech corpus. Extensions and improvements of the initial segmentation framework are presented in chapter 5. The research on the vowel classification task is treated in chapter 6. Evaluation and experiments are done in chapter 7 and finally conclusions are drawn and an outlook for future work is given in chapter 8.

Chapter 2

Motivation and background

From a phonetic point of view, a spoken utterance can be described as a sequence of phones – its phonetic transcription. The goal of speech segmentation is to segment and label an utterance into smaller segmental units. For this thesis, these segmental units are the phones and thus the segmentation finds the boundaries of the phone sequence. To split the utterance into these phone segments, an accurate alignment of the speech signal and its phonetic transcription must be done. If the phonetic transcription is known in advance, this task is often referred to as *phonetic alignment*. The exact phonetic transcription is not always known before the segmentation is performed. In many cases, when segmenting a single utterance or a whole corpus, all information previously known is the speech signal and the text, i.e. the orthographic transcription of the utterance. Using this orthographic transcription, a so called canonical phonetic transcription can be created by looking it up in a pronunciation dictionary that contains per-word transcriptions or by applying pronunciation rules. The task of inferring a phonetic transcription from the text and subsequently aligning it with the speech signal is also called *text-dependent alignment*.

The task of automatic phonetic segmentation of speech is a complex process and its details can vary depending on the application. For this thesis, we deal with speech corpora, a scenario where recordings of a speaker are done in a controlled environment with a given text. Thus it is necessary to convert the text to a phonetic transcription automatically. This transcription must then be aligned with the acoustic signal. The resulting alignment partitions the signal into a phone sequence: its phonetic segmentation. Design goals of the whole process are accurate segment boundaries and a transcription that closely matches the phone sequence actually produced by the speaker.

Automatic speech segmentation has been addressed many times in the past and several systems have been proposed. Nevertheless there are various situations where segmentation systems run into problems. Such problems can be caused by a lack of resources for the recorded language, e.g. missing pronunciation rules or pre-built statistical models. An important factor that causes problems is pronunciation variation. Pronunciation variation can e.g. occur when a speaker uses a dialect or a regional language variant that differs from the standard pronunciation. Regional pronunciation variation is often determined by the realization of the vowels. The principal dialects of English, for example, do not have many differences in their consonants [Lad05]. Thus vowels

are of particular interest for this thesis.

Often, speech segmentation systems assume that the pronunciation follows a standard that can be looked up in a pronunciation dictionary. Sometimes the systems model variation of the standard pronunciation. If so, they do this by applying pronunciation variation rules that are already known. Therefore, to achieve good segmentation accuracy, it is of particular importance to gather knowledge about the systematic pronunciation variation of a certain language variant and to integrate this knowledge into the segmentation system.

2.1 Phonetic transcription of speech sounds

The phonetic transcription of speech is a sequence of symbols where each symbol represents a phone. There are numerous different phones that humans can realize, but individual languages don't make use of all possible phones. Different languages have different phone inventories. The phones that are actually used by speakers of a certain language are a language-specific subset of all possible phones that are realizable by humans.

Nevertheless it is desirable to have a universally applicable symbol set for phonetic transcriptions that covers all the phones that can occur. A standardized symbol set that meets this criteria is defined by the International Phonetic Association (IPA). They publish the International Phonetic Alphabet (also abbreviated IPA) [AC99]. It contains a comprehensive set of phone symbols. Suprasegmentals and diacritics are used to modify nuances of the phones (such as stress, nasalization, etc.). The IPA is commonly used by phoneticians all over the world.

Due to the large symbol set, the IPA contains a variety of special characters which can, unfortunately, still cause problems in many applications of computer-aided speech processing. Due to this, alternative phonetic alphabets were proposed. Their main goal is to provide a symbol set encoded in 7 bit ASCII characters. In many cases, only a subset of the IPA symbols is represented in such alphabets, depending on the specific application. A famous example for such an alphabet is the Speech Assessment Methods Phonetic Alphabet (SAMPA).

In this thesis, transcriptions are given in IPA unless otherwise noted. For the work carried out, two machine-friendly phonetic alphabets are used: The first one is an adapted version of SAMPA for Austrian German (SAMPA Austria) that is provided with the analyzed corpus (ADABA). The second alternative phonetic alphabet used is the SYNVOPA, developed by the company SYNVO and thus used in the segmentation tool (described in section 5.4). The alternative phonetic alphabets and their mappings to IPA symbols are listed appendix A.

2.1.1 Articulatory phonetics

The phonetic transcription described above is based on a phone-level representation of speech. Although phones are used as smallest base units in many state of the art speech processing applications, it has been argued that the phone may not be the optimal smallest base unit [Ost99]. Articulatory features are a representation of speech sounds at a sub-phone level. Each phone can be described by

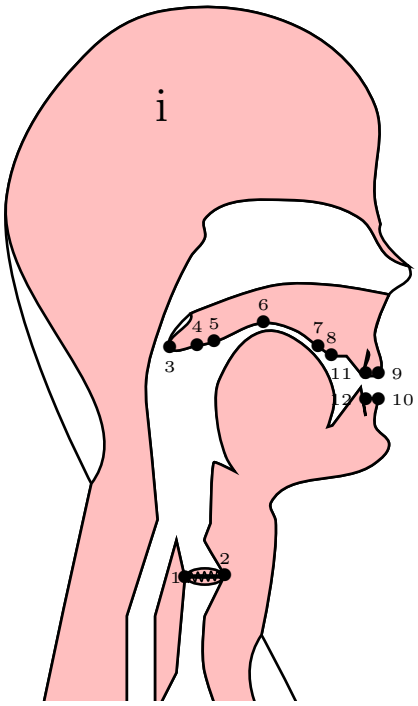


Figure 2.1: The human vocal tract. The figure is realized with the Vocal Tract LaTeX package [VSK12].

articulatory features. They are directly related to the speech production process by the human vocal tract and describe the evolution of different articulators in time during speaking. Figure 2.1 shows the articulators in a human vocal tract. A more detailed explanation of the articulators in speech production and their relevance to phonetics can be found e.g. in [Lad75] and [AC99]. In this section, only a brief description is given to provide the necessary background for later chapters. The relevant locations are marked with bullets: The vocal folds and the glottis (1, 2), the lips (9, 10), the teeth (11, 12), the alveolar ridge (8), the hard palate (6), the velum (5) and the uvula (4). Position (7) is the place of articulation for post-alveolar consonants. If position (3) touches the back of the pharynx, air only passes through the oral cavity, otherwise also through the nasal cavity. The latter is the case for nasal vowels and consonants.

Each phone can be described as a vector of such articulatory features that represents a specific vocal tract configuration necessary to produce the phone. However, articulators may change asynchronously and not only at the phone boundaries. Phones can be distinguished in two major categories, consonants and vowels. For consonants, the airflow during speech production is disturbed somehow, either blocked (as for plosives) or restricted. Vowels are voiced sounds where the airflow is basically undisturbed. The vocal tract, however, has a different shape for the different vowels. Especially the tongue position and the mouth shape influence the resonance frequencies of the vocal tract during vowel production. These are the articulators that are responsible for producing the

different vowels.

Consonants can be described by the articulatory features *manner* and *place* of articulation. They describe the manner of airflow disturbance and the position where this disturbance takes place. Consonants are classified by the manner of articulation into (see [JMK00]):

plosives where the airflow is entirely blocked for some time followed by a burst, i.e. a sudden release of air

nasals where air is also passing the nasal cavity

fricatives where a narrowing is done by the articulators and causes turbulences in the airflow

approximates are also done by a narrowing of articulators but without turbulences in the airflow

taps or flaps are articulated in a similar way as plosives, however, there is no airflow pressure that is blocked and thus now burst follows. They are realized by movements of the tongue against the alveolar ridge.

Fricatives with a higher pitch are called *sibilants*, fricatives that follow plosives are called *affricates*.

The following places of articulation are distinguished [JMK00]:

bilabial where the disturbance (in fact a blocking) of the airflow is done by the lips

labiodental where the constriction is made by teeth and one lip

alveolar where the constriction takes place just behind the teeth

palatoalveolar where the constriction takes place at the end of the alveolar ridge

palatal where the constriction takes place at the palate

velar where the constriction takes place at the velum (the soft part of the palate)

uvular where the constriction takes place further back than the velum

glottal the constriction takes place at the glottis

Using this information, it is possible to describe phones by a few articulatory features (place and manner for consonants, lip rounding and tongue position for vowels). Another sub-phone phonetic description of speech that is similar to this one is the concept of binary *distinctive phonetic features* and was described in [CH68]. The individual features can be present or absent for each phone. The set of distinctive phonetic features is used to distinguish individual phones.

2.2 Regional varieties and dialects

There are several thousand languages in the world. For example, [Lew09] counts, at the time of this writing, 6910 different languages. Nevertheless an exact number is hard to give due to definition problems of the term *language*. Some languages, such as German, are pluricentric languages and thus have different standard varieties (e.g. for German, there are the German spoken in Germany, Austrian German, Swiss Standard German, and the German spoken in certain regions of other European countries) [Cly92]. These standard varieties differ from each other. Nevertheless each variety is considered to be a correct standard in the respective country. Further, most languages have several dialects, i.e. varieties that typically do not have a standardized written form and are only used in colloquial language. The boundary between the terms *language*, *variety* and *dialect* is not always clear. A famous quote with unknown origin answers the question concerning the difference between a language and a dialect with "*A language is a dialect with an army and a navy.*", indicating that the definition and usage of the terms is influenced by politics and power too, and not only by scientifically funded factors.

For speech processing applications, however, the exact boundaries of the linguistic definitions are of less importance. In theory, if the necessary resources such as corpora and pronunciation dictionaries are available, a speech processing application can be build for any language, variety or dialect. Nevertheless, if there are no or few resources available for a certain variety or dialect, only the ones of the related standard language or a related variety can be used – resulting in performance drawbacks.

An important linguistic aspects in which regional variants of a particular language differ from each other, is the pronunciation of words. Often, deletions of certain phones in specific contexts or substitutions of phones or phone sequences with other phone sequences can be observed. This pronunciation variation causes specific challenges to automatic segmentation systems.

2.2.1 Standard varieties of German

German can be considered as a pluricentric language [Cly92]. It is the official language in Germany, Austria and Liechtenstein and one of the official languages in Switzerland and in Luxembourg. Further it is a co-official language in some parts of Italy and Belgium. In several additional countries it is recognized as a minority language.

The national varieties have a lot in common, however, there are differences that are considered as standard in their respective country. Differences can be found in the vocabulary, in some grammatical concepts in spoken standard language, in the accentuation of syllables and in the pronunciation of words.

Differences in the vocabulary denote words or phrases that do not occur in all standard varieties of the language. Austrian specifics include many words for food (e.g. *Erdapfel* in Austria versus *Kartoffel* in Germany) or vocabulary related to state institutions, but the differences are not restricted to these two areas.

Grammatical differences include the gender for some nouns and different auxiliary verbs when composing the perfect for some verbs (e.g. English *I sat* in German: *Ich bin gesessen* vs. *Ich habe gesessen*).

Some words show a different accentuation pattern in different standard varieties. For example, the word *Kaffee* is stressed at the first syllable in most parts of Germany and on the second syllable in Austria.

And finally, there are phonetic differences, that is, variety-specific pronunciation of words. For this thesis, the differences in pronunciation are the main scope of interest. An analysis of pronunciation variants in Austrian German was done e.g. in the thesis of Michael Baum [Bau03]. In his thesis, narrow transcriptions for a large telephone database (Speech-Dat, see [BEK00]) were performed. He identified various context-dependent systematic variations which were subsequently used to improve the accuracy of a speech recognizer for Austrian German. An example for a phonetic difference is the realization of the phoneme /z/: In many regions of Germany it is actually pronounced as the phone [z] whereas in Austrian German it is often realized as the unvoiced phone [s].

2.3 Segmentation and labeling of speech

In many situations, the canonical phonetic transcription does not equal the actual phonetic content of the utterance spoken. This phenomenon is called *pronunciation variation*. A segmentation system has to take this variation into account. When modeling pronunciation variation, not only the canonical phonetic transcription of a word is considered, but also alternative pronunciation variants. A segmentation system then tries to align all possible phonetic transcriptions with the speech signal. The best matching alignment then determines the segmentation, which now not only identified the phone boundaries, but also the best matching pronunciation variant. The main tasks of the underlying algorithm thus are to

- perform a local acoustic-phonetic matching, i.e. determine how well the speech signal at a given time matches a certain phone and identify the best matching phone respectively
- perform a time alignment of the best matching phone sequence and the speech signal

The latter is a well-known problem of aligning time series and can efficiently be handled by dynamic programming algorithms. It has been addressed many times, also in speech research, and the resulting concepts are similar in the time alignment step and differ only in the way how the local acoustic-phonetic matching is done. If the acoustic-phonetic matching is done using phonetically labeled reference-templates, an alignment algorithm called Dynamic Time Warping (DTW) can be used. If the acoustic-phonetic matching is done by comparing the speech signal to a statistical model of each phone, as usually done in speech technology with Hidden Markov Models (HMM), an alignment algorithm called Viterbi search can be used.

The raw speech signal is redundant from a phonetic point of view. Thus, in speech technology, it is usually represented as a stream of acoustic features. These features provide a more compact representation of the perceptually and phonetically relevant parts of the signal. Alignment is then done between this feature stream and the phone sequence. Figure 2.2 illustrates the phonetic alignment.

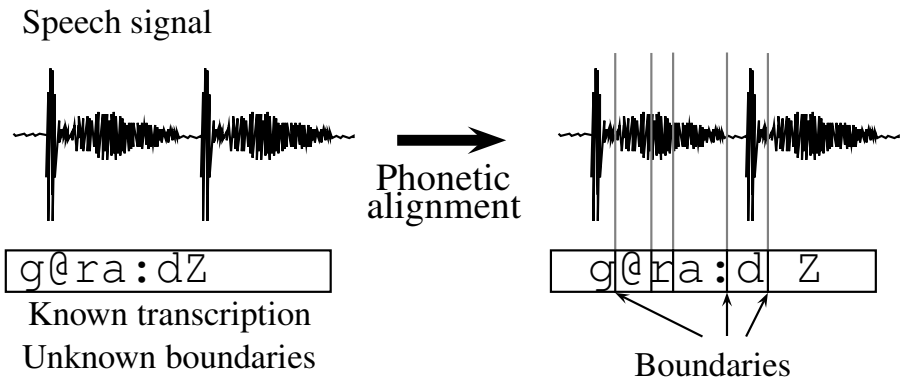


Figure 2.2: Alignment of a speech signal with the phonetic transcription of the word `garage`. The transcription is given in SYNVOPA notation.

In the past, several systems have been proposed for automatic speech segmentation. Often they are based on HMMs (see e.g. [Rab89]). The HMMs are used to acoustically model the individual phones, often by using Gaussian Mixture Models (GMMs) to model the acoustic observation sequence (the feature vector stream) produced by the hidden discrete state sequence that represents the phones. In the simplest case, the alignment is done by restricting a Viterbi-search (see [For73]) to the phone model sequence of the known transcription. In contrast to Viterbi-decoding in speech recognition, the best inter-model path is assumed to be known, and therefore there is only one possible sequence of models (assuming no pronunciation variants are considered). The algorithm finds the minimum-cost alignment between the feature vectors and the states of the model sequence and is thus able to estimate the location of the phone boundaries. The phonetic transcription needed for the alignment is derived from the orthographic representation which is assumed to be known for our speech segmentation application. If this transcription is a single canonical phonetic transcription, there is a single model sequence that is aligned with the feature stream. Alternatively, several pronunciation variants can be accounted for. Then the search finds the boundaries as well as the model sequence path that best matches the acoustic content. In figure 2.3, the input and output of this operation is shown.

2.4 The speech signal

A speech signal is the sound waveform produced during the speaking process. This sound waveform is highly redundant in a linguistic sense. Hardly ever two realizations of the same utterance will result in the same sound waveform, even if produced by the same speaker.

The sounds that can be produced in the speaking process are restricted by the physiological characteristics of the human vocal tract. The physical process involved in human speech production allows for specialized representations and features when analyzing a speech signal. Some basic concepts that are exploited are described in the following sections.

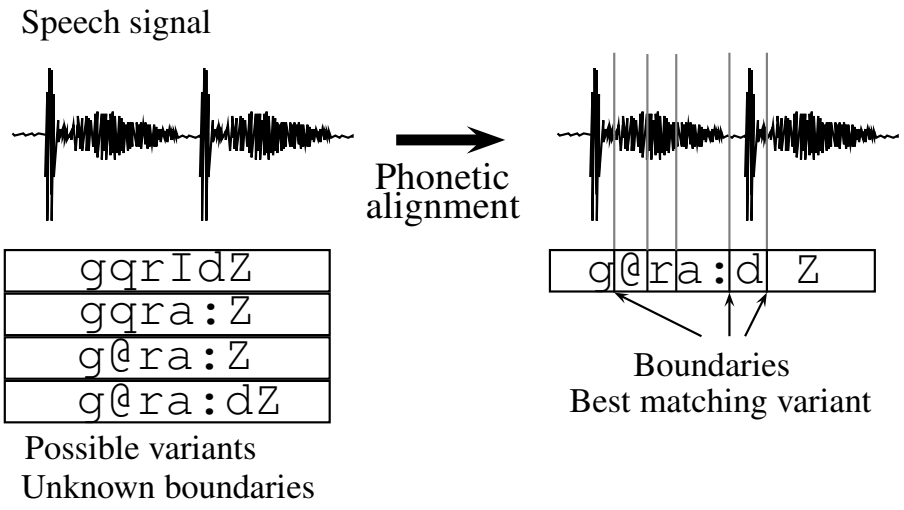


Figure 2.3: Phonetic alignment of the speech signal with several possible transcriptions of the word `garage`. The transcriptions are given in SYNVOPA notation.

2.4.1 The source-filter model of speech

The source-filter model is a model for the speech production process. In this model, the speech signal is generated by an excitation source and subsequently passes a filter that models the vocal tract. The source and the filter are assumed to be independent of each other. This model is of high importance in speech processing as it approximates important aspects of the speech production process. For voiced sounds, the physical equivalent of the source is the vibrating glottis which produces a periodic signal. For unvoiced sounds, the physical equivalent of the source is assumed to be approximately white noise caused by the turbulent airflow coming out of the lung. The filter is realized by the shape of the human vocal tract.

The source-filter model is also of interesting from a perceptional point of view. Source and filter are of different importance for the perception of speech. A vowel, for example, is produced by a periodic excitation source and a vocal tract shape that does not substantially constrict the airflow. This shape, however, influences the resonance frequencies of the vocal tract, which are defining the vowel's identity. The source, on the other hand, does not significantly influence the identity of the vowel as long as it is a periodic signal. This becomes apparent when considering the fact that the same vowel can be pronounced with different pitches. Even if the excitation is not periodic, the vowel's identity may still be perceived correctly (e.g. when whispering, there is no periodic excitation as the vocal folds don't vibrate).

Thus, in speech processing applications, the separation of source and filter is exploited. It is tried to separate parts of the signal that equal the filter in this model from the parts that equal the excitation source.

2.4.2 Formants

Formants are local energy maxima in the spectrum of a vowel. Their physical interpretation are the resonance frequencies of the vocal tract. Therefore they are of great importance for the vowel identity. The first few formants (defined as the *F-pattern* in [Fan60]) are in general sufficient to distinguish between different vowels of a speaker. The first and the second formant (F1 and F2) even have a direct relation to physical properties of the vocal tract shape (see e.g. [Joo48]): F1 is related to the backness of the highest point of the tongue and F2 is related to the openness of the mouth in the vowel production process (sometimes F2-F1 is used instead of F2). Thus, when using appropriate scaling, a chart of the first two formants resembles the vowel quadrilateral.

In [LHGR78], an algorithm was presented that is able to infer vocal tract shapes for English vowels from the first three formant frequencies only. The authors analyzed the tongue's position at 18 points in the vocal tract from x-ray images. They discovered that the tongue's shape can be determined by only two principal components, which they identified to be *front raising* and *back raising* components of the tongue and that these two components can be derived from the first three formant frequencies.

Nevertheless, the vocal tract shape and thus the F-pattern of a specific vowel differs among speakers due to different anatomical properties. This makes them speaker-dependent and hence some kind of normalization is needed if formants are used in a speaker-independent speech processing application. Further, reliable formant estimation from the speech signal is difficult [BSH07].

In this thesis the formants are retrieved using the program Praat [Boe01]. This software estimates the formants by using Linear Predictive Coding (LPC) with the Burg-algorithm (see e.g. [PTVF07]).

2.4.3 Feature representation of a speech signal

It has already been mentioned that the speech signal is redundant from a linguistic point of view. The variation introduced by different vocal tract shapes, ambient noise, intensity, pitch, speech rate, etc. usually has no influence on the phonetic content of an utterance. Therefore, it is not a good choice to use the sound waveform directly in speech processing applications that aim to extract phonetic content of an utterance. Instead, the goal is to use only relevant information. During decades of research in speech processing, sophisticated features were proposed that emphasize the perceptually and phonetically relevant characteristics of the speech signal and that are highly invariant to the variation mentioned before. Among the most popular ones used today, there are the Mel Frequency Cepstral Coefficients (MFCCs) [DM80]. To calculate them, the speech signal is partitioned into overlapping windowed segments and this segments are transformed to the frequency domain. The result is called spectrogram and provides a time-frequency representation of the signal. Then a filterbank analysis is performed on the spectrogram. The filters are overlapping and not equidistantly distributed on the frequency-axis. Instead they are aligned on the Mel-scale to reflect the perceptual characteristics of the human ear. The result is a sequence that contains the amplitude of the energy in each filter-bank channel respectively. In the last step, this resulting sequence is transformed to the cepstral domain by performing the discrete cosine transformation

of the logarithm of the Mel-filterbank output. The representation of speech in the cepstral domain is motivated by the source-filter model that sees speech as a combination of an excitation source (the vocal-folds for voiced sounds and random noise for unvoiced ones) and a filter behind (the vocal tract). Speech sounds can be characterized better if explicit knowledge about the source and the filter is available than from the raw waveform. In the model, the signal emitted from the source undergoes the mathematical operation of convolution with the filter's impulse response in the time domain. In the frequency domain, this operation becomes a multiplication. By using the logarithm, the multiplication becomes an addition, or superposition. The separation of superpositioned sequences is approximately possible if they do not overlap too much. The discrete cosine transformation ensures a good decorrelation of the individual features.

There are other types of features than MFCCs used in speech processing. A frequently used example are Perceptual Linear Prediction (PLP) coefficients [Her90]. They share several characteristics with MFCCs (e.g. spectral analysis and the use of a filterbank that reflects properties of the human auditory system) and show comparable performance in various applications. For details, refer to the literature. With a proper representation of a speech signal by features as MFCCs or PLP coefficients, the foundation for various methods in speech processing is built. Relevant approaches for automatic phonetic alignment are descibed in the following chapter.

Chapter 3

Segmentation of speech signals

This chapter presents an examination of the methods used in automatic speech segmentation. The two alternative concepts behind the most common approaches are explained and existing systems based on these concepts are mentioned. Further, different approaches for dealing with pronunciation variation are described.

3.1 Phonetic alignment

The alignment of a phonetic transcription with a speech signal can be done in various ways. Most state-of-the art systems use hidden Markov models (HMMs) for this tasks. As an alternative, the technique of dynamic time warping (DTW) has also been used in the past. In the following subsections, the two methods are explained briefly. For details, refer to the literature mentioned in the text.

3.1.1 Hidden Markov Models

Hidden Markov models have been applied successfully as statistical models to various applications in speech processing. The basics concepts behind these models are reproduced here briefly. For a detailed explanation of HMMs, please refer to the literature, e.g. [Rab89].

A HMM is a statistical model for sequential data. The underlying process is modeled by a state sequence which is not directly observable (hidden). The transition between the hidden states of the sequence is assumed to follow a Markov process. Thus, a future state depends only on the present state and not on other states in the past. Even though the state sequence is hidden, each state produces an observation that is visible. These observations are modeled by a second random process. If the observation space is discrete, the model is called discrete density HMM (DDHMM), if the observation space is continuous, it is called continuous density HMM (CDHMM). For CDHMMs, Gaussian mixture models (GMMs) are often used to model the observation probabilities.

A HMM is defined by the following properties:

- The set of *states*

- The transition probabilities between the individual states, usually represented as a *transition matrix*
- The *observation probabilities* for each individual state. The observation space can be discrete or continuous.
- A *starting state* for the state sequence

A HMM thus is useful if observed sequential data (the observation sequence) is somehow connected to an unknown underlying second sequence (the hidden state sequence). Frequently, the sequential data is time-series data.

In speech processing applications, the observed data is the acoustic signal and thus a time-series signal. As mentioned in section 2.4.3, instead of using the samples of the waveform directly, a feature representation is used. The observation sequence is thus a time-series of feature vectors, extracted from the acoustic signal at equidistant intervals.

The usual phonetic representation of speech is the phonetic transcription and thus a sequence of phones. Therefore it is a good idea to do the acoustic-phonetic modeling of speech by HMMs where the hidden state sequence represents the phone sequence and the observations represent the feature sequence extracted from the signal. As commonly used features have a continuous range, the observation space is continuous. Therefore, CDHMMs are used for the modeling.

It is common to use multiple states to model a single phone. This allows the individual states to represent different parts of the phone. For example, with a three state HMM the states can model the transition region from the previous phone, a more or less stationary part and the transition region to the proceeding phone. In this example, the three states of a phone are expected to be in a defined order (the transition region to the proceeding phone is never before the stationary part). Hence it is useful to restrict the possible transitions between the states to be in forward direction. Often, the skipping of individual states is forbidden too, leading to a so-called left-to-right HMM.

Usually, three common problems are described when applying HMMs in practice [Rab89]:

Evaluation of the probability that a specific HMM produces an given observation sequence

Decoding of the most probable hidden state sequence that produces a given observation sequence

Parameter estimation of an HMM by supervised learning. Numerous observation sequences belonging to a specific HMM are given as examples during training.

The *evaluation problem* can be solved by the *Forward-algorithm*, the *decoding problem* by the *Viterbi-algorithm* and the *parameter estimation problem* by the *Baum-Welch-algorithm*. The algorithms are not be reproduced here, please refer to [Rab89] for an explanation of them.

When using HMMs as models in speech processing, the Forward-algorithm gives the probability that a particular model (for a phone, subphone, or word) produced the given acoustic feature sequence. The Viterbi-algorithm is used to find the best alignment between the feature sequence and the phonetic transcription of an utterance. Alternative pronunciation variants can be considered

too. The Viterbi-algorithm is applied when doing speech segmentations with HMMs. The parameter estimation is used to create the desired models, e.g. phone models that are used for further processing (such as speech recognition or speech segmentation).

Frequently, each phone is modeled by a separate HMM. These HMMs are trained via the Baum-Welch algorithm using data that is already labeled. When the models are used for a decoding task such as speech recognition or speech segmentation, they must be applied to larger units such as words or sentences. To achieve this, the individual phone HMMs are connected together according to the phonetic transcription of the utterance. A lattice of possible hypotheses can also be produced this way. This is useful when multiple pronunciation alternatives should be searched. A decoder then searches for the most probable path through this lattice of HMMs. The resulting path equals the best matching phone and state sequence for the utterance, given the lattice and the models. When dealing with the lattice structure during alignment, extensions to the plain Viterbi algorithm are necessary. An efficient solution to the decoding problem when using such a lattice is to use an algorithm called token-passing (explained e.g. in [YRT89]).

When using a HMM-based forced alignment procedure to segment a speech corpus, the phonetic transcription of the utterance and its alternative pronunciation variants define the search graph. To successfully apply the algorithm, a precondition is, that models for all the phones in the search graph are available. If no phone models for the speaker of the utterance exist, speaker-independent models can be used instead. If speaker-independent models are unavailable too, there are two options to get started: Either some data must be manually labeled and used to *bootstrap* the HMMs, or a so-called *flat start* approach can be used. For the latter, a uniform segmentation is assumed to train the initial acoustic models. With this acoustic models, a new forced-alignment can be performed that will probably result in a better segmentation. The new segmentation can then again be used to re-train the acoustic models and so on. This iterative procedure is called embedded segmentation. Bootstraping of models, the flat start approach and embedded segmentation are explained e.g in the HTK book [YEG+06].

3.1.2 Dynamic time warping

Another possibility to align the speech feature vectors with a transcription is to use reference templates instead of statistical models. When comparing a phone to a reference template, the possibility of length differences must be taken into account. An efficient way to solve this is dynamic time warping (DTW). DTW performs the alignment of two time series while allowing a non-linear warping of the time-axis. Basically this is done by:

- allowing a feature vector frame of the analyzed time-series to match with several frames of the reference time-series
- allowing the algorithm to skip frames of one time-series

Using DTW for automatic segmentation, no statistical models of phones are needed. Instead, reference templates of them are used for alignment with the

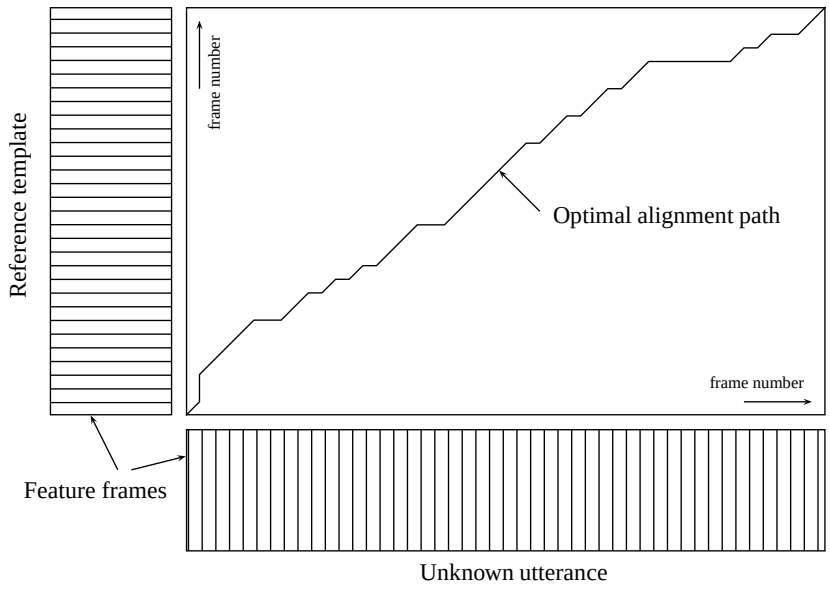


Figure 3.1: DTW alignment of two feature streams.

feature stream of the utterance to segment. Individual templates can be concatenated to create a template for a whole utterance.

Figure 3.1 shows an example alignment path of two feature streams. It can be seen that the two time-series may have different lengths. For a more detailed discussion of the DTW algorithm and its properties, please consult the literature, e.g. [SK83].

3.2 Existing systems for speech segmentation

Many existing speech segmentation systems are based on the alignment concepts either DTW or Viterbi-alignment with HMMs, respectively. Another classification of the systems can be made by the a-priori knowledge that the systems require. Some systems expect nothing but the acoustical signal to be known. They don't need the text or the transcription of the utterances. Several examples for such systems are described in [EA05]. As these systems need to perform unconstrained phone recognition (explained below in section 3.3), one has to accept some performance drawbacks. Other systems expect the actual phonetic realization of the utterance (a narrow phonetic transcription) to be known. Given the phonetic transcription, a phonetic alignment method using reference templates or HMMs can be applied. Frequently, segmentation systems expect only the orthographic transcription to be known. In that case they infer the canonical phonetic transcription from the text. Sometimes they assume that the speaker follows the standard pronunciation and use the canonical transcription directly for the phonetic alignment. More sophisticated systems try to consider variation of the standard pronunciation in the alignment.

For this thesis, variation of the standard pronunciation is of particular importance. Many regional language variants are under-resourced. They lack

available speech corpora, models and pronunciation dictionaries. It is thus more difficult to deal with regional language variants and their pronunciation variation using available frameworks for speech segmentation.

In the following sections, some existing systems for automatic segmentation based on DTW and HMMs are presented briefly.

3.2.1 DTW-based speech segmentation systems

When segmenting a new utterance with the DTW algorithm, a reference template for this utterance is needed. If the segmentation task has a constrained vocabulary (e.g. single digits or a small set of words), it is possible to maintain a database of pre-segmented reference templates for the whole vocabulary. But in general it is infeasible to have reference templates for all possible utterances. Nevertheless the templates can be generated synthetically by concatenating smaller units such as phone templates. Using concatenated phone templates has an additional advantage: The phone boundaries of the synthetic template are defined by the concatenation locations. Thus no manual segmentation on the template database is necessary (of course, to get the phone templates, a segmentation has to be performed).

An early DTW-based segmentation system, however, works without reference templates. It is described in [Wag81]. A two-stage algorithm based on DTW is used to segment an utterance, given its transcription. DTW is used in both stages. The system uses no reference templates but a set of acoustic-phonetic rules along with the features energy, voicing, the fundamental frequency and the linear prediction coefficients (LPC). In the first stage, an expanded version of the phonetic transcription is mapped to a rough acoustic feature stream (voiced/unvoiced/silence features, and formants for the voiced segments) by DTW. The local acoustic-phonetic matching is done by a table-lookup of the distance between the acoustic categories (voiced, unvoiced and silence) and the phonetic categories of the phone symbols (e.g. *vowel*, *voiced fricative*, *stop gap*, etc.). Consider the example of the phonetic category *stop gap*. In the cited work it is expected that the stop gaps are realized as *silence* or as an *unvoiced* acoustic segment. As the silcence variant is considered to be more likely, the distance of a stop gap to *silence* is set lower than its distance to *unvoiced*. As the author does not expect the stop gap to be voiced, the distance to this acoustic segment is set to infinity. In the second stage, the transcription is expanded again to include transitions to and from the neighbouring segments. This new expanded transcription is then mapped to the acoustic frames by using a second DTW algorithm. This time, local matching is done by comparison of the derivatives of energy and formants to their expected values, that are stored in a table and depend on the phone transition.

[SS87] experimented with, among others, a template-based method. It uses speaker-independent single-speech-frame reference templates for each phone. The templates are concatenated according to the given phonetic transcription and aligned with the unknown utterance by a dynamic programing algorithm. Their results show that 92% of the boundaries lie within a 45 ms interval of a manual reference. Gong and Haton [GH93] use DTW and perform speaker adaptation to achieve a better match with the reference templates. [Cam96] uses a speech synthesizer to produce reference templates for the automatic segmentation of a Japanese corpus using DTW.

In [MD97], the authors use a high-quality speech synthesizer that creates an acoustic reference utterance from the given text. The phone boundaries in the synthesized speech signal are known. The created reference utterance is then aligned with the unknown utterance via DTW. The phone boundaries of the unknown utterance can be determined by the mapping to the synthesized utterance with boundaries. Several years later, the same authors compared their DTW and speech synthesis-based approach with one that is based on HMMs and Artificial Neural Networks (ANNs) [MDDR03]. Their results showed that the DTW-based approach was inferior. Nevertheless, they emphasized the advantage that the DTW-approach neither requires an initial segmentation of the corpus, nor already trained models to perform such an initial alignment. As a consequence they proposed to use the DTW-approach to perform the initial alignment and to perform a refinement with the HMM/ANN approach.

Another interesting approach published recently that uses DTW is presented in [GSCB10]. This segmentation system is based on unsupervised acoustical clustering via DTW and Gaussian mixture models. The authors further apply boundary refinement techniques after the core segmentation. The refinement method applied depends on the phonetic classes of the adjacent phones. For subsequent vowels, the acoustic similarity of consecutive frames is used to refine the boundary in an iterative procedure. For boundaries with plosive or silence phones the differential energy is used (for the plosives, a shorter window length is applied than for the silences). For other phones, the differential energy and the differential zero crossing rate is used. With their method, the authors achieve 84.7% of the boundaries to be within 20ms tolerance on the TIMIT corpus.

Several DTW-based segmentation systems have been proposed in the past. However, none of them was able to outperform the best HMM-based approaches discussed in the next section.

3.2.2 HMM-based speech segmentation systems

In [TW94], an automatic segmentation system called *Aligner*, that uses the HMM-based forced alignment approach, was introduced. A phonetic transcription and pronunciation variants are derived from the text via a pronunciation dictionary containing more than 113000 entries. Segmentation is done via HMM-based forced alignment. The extracted features are MFCCs with a 10ms frame shift: 12 cepstral coefficients and their derivatives as well as the delta energy (the energy itself is not included). The authors evaluated their approach on the TIMIT corpus. Comparisons of the boundaries with the manually labeled TIMIT test set was not straightforward as the resulting phone sequences of the aligner differed from then manual TIMIT transcriptions in one out of three times. With doing a symbol mapping based on a heuristic using phonetic features, however, they achieved around 72% agreement with the test set for a tolerance of 16ms and about 91% agreement for a tolerance of 32ms.

The Munich Automatic Segmentation system [Sch99] (MAUS) is another example for a HMM-based segmentation system. The framework generates a lattice of pronunciation variants before doing the forced alignment with speaker-independent models of the target-language. An iterative version of MAUS (see [Sch04]) is also available. In this version, the first segmentation is used to re-train the acoustical models. The resulting models better fit the target

speaker and be used segment the corpus again. The re-training and segmentation is repeated until convergence of the boundaries is achieved.

The system in [Hos09] provides, to my knowledge, the best-reported results on the TIMIT database (93.36% agreement within 20ms). In this approach, transition-dependent states as well as additional features are used to improve a baseline system. This baseline system consists of an HMM/ANN hybrid that is trained on the TIMIT training set. Nevertheless this system required the exact realized phoneme sequence to be known to achieve its considerable results (it includes the ability to work with pronunciation variation based on a dictionary and on rules, but no results are reported for this case).

In [MGF08], embedded training with a flat start approach is used as a first iteration. Subsequently, isolated unit training is performed with the initial segmentation result. Notable is that they use no hand-labeled bootstrap information for training the models. In isolated unit training, each HMM is trained separately using the labeled data corresponding to the trained model, in contrast to embedded training, where all HMMs are trained simultaneously using all the data and a segmentation is done implicitly during training. They achieve an accuracy of 83.56% on the TIMIT test set (for the 20ms interval).

The framework presented in [OCB10] is a segmentation system that aims at being applicable to under-resourced languages too. The authors use acoustic models based on articulatory features and argue that these models generalise better across languages. In a first step, a forced-alignment of the given transcription and given speaker-independent models is done. No phone models are used in this step. Instead the authors use models trained on the place and manner of articulation (see section 2.1.1). When using speaker-independent models, the utterance to segment and the models do not necessarily use the same phone set. Thus, when using phone models, some phones in the unknown utterance may lack a corresponding phone model. This problem is usually solved by mapping such phones to similar models. In the mentioned work, however, this problem does not occur. The given phonetic transcription of the utterances to segment is converted to two streams containing their articulatory feature transcription first. By doing so, an implicit mapping between the phone set of the given speaker-independent models and the phone set of the utterances to segment is done. Of course, models for all possible articulatory feature values for the place and manner of articulation are needed in that case, but this is easier to achieve, as the space for each of the two features is quite limited. Further, the use of models based on articulatory features ensures that more data per model is available in the subsequent embedded re-estimation step than in the case of phone-based models. The authors evaluate their system in two ways. They compare a segmentation based on the articulatory feature models with one based on phone models. Both model sets are trained and applied on the TIMIT corpus. The phone based models perform better in this task. The second task is the segmentation of a corpus with a different target language and a different phone set. Monophone models for the target language are bootstrapped using a segmentation based on manner and place models trained on the source language. They perform better than monophones that resulted from training using a flat-start approach. This work indicates that the approach of mapping to broader phonetic categories like place and manner of articulation can help when dealing with under-resourced languages. The result that phone-based models produce more accurate segmentations than articulatory feature-based models indicate

that the mapping may be improved further.

A typical state-of-the-art HMM-based speech segmentation system includes the following modules:

Feature extraction. Features like MFCCs or PLPs are usually used as basic input data for further processing as known from ASR applications. These features are frequently combined with their delta- and acceleration coefficients. In ASR systems, it is common to extract them at a period of 10ms using a window size of 25ms. The choices for ASR systems may, however, not match the needs of segmentation systems perfectly. For example, the time resolution of the forced alignment is limited by the frame shift of the extracted features. The usage of additional features may improve the accuracy too. Both adaptations are implemented in good systems like [Hos09].

Pronunciation variation modeling. The application of the speech segmentation system is not limited to cases where the realized phone sequence of the utterances is known exactly in advance. Often, linguistic resources provide the text along with the recorded utterances. In some cases, they are created by giving the speakers the text they should read during the recordings in advance. Such situations are, for example, the recordings of corpora like the one used in this thesis (ADABA). Another example is the creation of audio books. In other cases, recorded utterances are transcribed textually at a later time. An example are recordings of telephone conversations. The process of transcribing the recordings orthographically is much faster than performing a manual phonetic transcription. In the described scenarios the realized phone sequence is not known explicitly and some means for dealing with variation from the canonical pronunciation is needed. This can be done by a comprehensive pronunciation dictionary containing multiple variants, by the usage of variation rules, or by relaxing the constraints of the forced alignment in a way so that the detection of additional, previously unknown, variants is possible (as explained in section 3.3 and 3.4).

Initial segmentation. A HMM-based segmentation system uses forced alignment to determine the boundaries. The way in which the initial HMMs are obtained is crucial for the segmentation performance. An improvement over a simple flat-start approach either requires an existing segmentation of a part of the utterances to bootstrap the acoustic models, or existing models that can be used to perform an initial segmentation (e.g. speaker-independent models). It should be noted here, that, while for ASR applications context-dependent models are preferred, for automatic segmentation also context-independent monophone models are used frequently [TGG03].

HMM-based forced alignment. Once HMMs that fit the data to segment are available, forced alignment can be performed by the Viterbi-algorithm. The search space is constrained to the expected pronunciation variants. During decoding, the variant and the boundaries that result in the maximum likelihood alignment are inferred.

Boundary correction. Additional algorithms to correct the boundaries after forced alignment can further improve the segmentation result. These algorithms assume that the boundaries found by the forced alignment are somewhere in the region around the true phone boundaries. They try to move the boundaries closer to their optimum, often by exploiting additional features. See, for example,[KC02] for such a system.

Iterative retraining After a forced alignment and further corrections a reasonably good segmentation should be the result. This segmentation, however, can now be used to train new HMMs that could subsequently be used for another segmentation iteration which should now be better and can again serve as basis for training new HMMs. Additional steps, such as pronunciation variation generation or boundary correction are also repeated if desired. See the approaches in [RP05], [KC02], and [MGF08] as examples where this concept is applied.

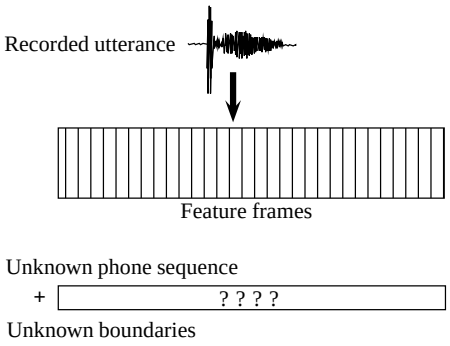
Apart from HMM- and template-based approaches, other segmentation systems have been proposed as well. An example is [vSS99], which explicitly detects the boundaries between phones by using edge detectors on various features. The detection is optimized for each diphone or alternatively for each diphone class. The list of segmentation systems mentioned in this chapter is not exhaustive, however, the described methods should give a good insight into the various ideas that have been developed so far. For a more detailed analysis of different segmentation systems, refer to [TGG03] and [Hos00].

3.3 Phone classification

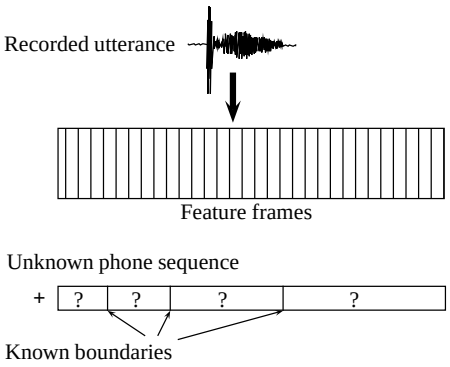
In this thesis, the term *phone recognition* is used for the task of phonetically labeling all feature vector frames belonging to an utterance when no prior information is given and no constraints are applied. Sometimes this is also called *frame classification*. The term *phone classification* will be used for the task of labeling phones, assuming that the boundaries are known in advance. The figures 3.2(a) and 3.2(b) illustrate the difference.

The input for a phone classifier is a sequence of acoustic feature vectors that belong to one particular phone and the output is the phonetic label of this phone. A phone recognizer, on the other hand, labels all of its input feature vectors and thus finds the labeling as well as the boundaries. Due to this, an unconstrained phone recognizer's performance is typically worse than that of a phone classifier. A reasonably good speaker-independent phone recognizer would imply a good labeling of all frames and thus a good segmentation. However, the best speaker-independent phone recognizers still have error rates of about 25%. Additional knowledge helps to improve accuracy. Usually, for ASR and segmentation applications, constraints are applied. A language model is used in typical ASR systems to restricts the search. For segmentation systems, the given phonetic transcription (optionally with alternative pronunciation variants) facilitates the automatic alignment.

Unconstrained phone recognition can directly support a speech segmentation and labeling system in finding the label identities as well as the boundaries between the phones. Phone classification, in contrast, could serve as an additional step after finding the boundaries to perform fine label classification. In



(a) Phone recognition



(b) Phone classification

Figure 3.2: Phone recognition and phone classification.

this thesis, phone recognition and phone classification based on ANNs, with a focus on vowels, is analysed. It is further investigated how this classification may support an existing HMM-based segmentation system.

3.4 Existing approaches for modeling pronunciation variation

The actual pronunciation of an utterance often differs from the canonical phonetic transcription. As mentioned in section 3.3, the phonetic transcription of realized pronunciation in speech signal should be known to achieve reasonable performance in a phonetic alignment task, due to imperfect recognition results of unconstrained phone recognizers.

The modeling of pronunciation variation has been addressed in several contexts. In ASR, the Word Error Rate (WER) can be reduced by incorporating models of pronunciation variation. Modeling pronunciation variation is usually done at word or at phone level, but can also be done with articulatory features (e.g. [BOW07]). A good survey of literature to pronunciation variation was done several years ago in [SC99]. The authors decide to classify the approaches along four characteristics:

- the type of pronunciation variation
- the information sources used
- the information representation
- the level of modeling

For the *type of pronunciation variation*, they distinguish between intra-speaker and inter-speaker variation. Intra-speaker variation covers alternative pronunciations of words by the same speaker, e.g. due to co-articulation effects. Inter-speaker variation covers differences between several speakers of the same language, e.g. based on their dialect or sociolinguistic factors. A speech segmentation system for regional variants has to deal with intra- and inter-speaker variation as the canonical transcription is commonly based on a single standard variant.

The authors further mention *information sources* as a distinctive characteristic of different approaches. In ASR, a distinction between knowledge-based and data-driven methods for modeling pronunciation variation can be made. Their main distinction is the starting point for handling pronunciation variation, whether this is the speech signal (data-driven) or linguistic knowledge (knowledge-based). In knowledge-based approaches, phonological rules are used to determine possible pronunciations. In the phonetic alignment step, all the generated variants are then considered. In data-based approaches, a manual or an automatic phonetic transcription is done and this transcription is considered in the phonetic alignment in addition to the canonical phonetic transcription. As manual transcriptions are time-consuming, the automatic solution is widely used. The automatic phonetic transcription may be smoothed in advance to limit the creation of incorrect variants [Wes03]. The whole approach could appear a little unorthodox as a phone recognizer is first used to generate alternative pronunciation variants and subsequently used to chose from all variants.

However, in [Wes03], it is argued that this approach is able to actually model pronunciation variation. The author achieves improvements when smoothing the automatic phonetic transcription with D-trees before adding the variants to the recognizer, even if this recognizer is not the same one that was used for the automatic phonetic transcription.

As the third feature to classify methods for pronunciation variation modeling, the authors of [SC99] use the *information representation* in the systems. They distinguish between systems that formalize the knowledge and subsequently generate the pronunciation variants and systems that directly use the variants without an explicit formalization. The information extracted from the data may be formalized in a more abstract way e.g. as rewrite rules. This can be done by aligning the canonical transcription with the transcription extracted from the data by a dynamic programming algorithm and deriving abstract rules from this alignment.

As the last criteria, [SC99] distinguishes at which level in an ASR system the modeling the pronunciation variation is done: in the lexicon, the acoustic models or the language model.

Most approaches that model pronunciation variation aim to improve the performance of an ASR system. However, variation modeling that is able to predict the realized phonetic transcription clearly can help to improve the performance of an automatic segmentation system too, especially when applied to data that may contain non-canonical pronunciations. For this thesis, dealing with pronunciation variation is of particular importance as the analyzed corpus contains regional language variants. The next chapter describes the used speech database in detail.

Chapter 4

The pronunciation dictionary ÖAWB and the phonetic database ADABA

The corpus that is analyzed in this thesis is the Austrian Phonetic Database (Österreichische Aussprachedatenbank, ADABA) that is provided along with the Pronouncing Dictionary of Austrian German (Österreichisches Aussprachewörterbuch ÖAWB/AGPD) [Muh07, Muh08]. The focus of the dictionary is the Austrian variety of German. The dictionary contains around 42000 transcriptions of common words for Austrian German. For almost 13000 entries, transcriptions for the Austrian, the German and the Swiss standard variety are included. All these entries are isolated words (with a few exceptions). Audio recordings of six speakers are provided along with the transcriptions: two speakers from Austria, two from Germany and two from Switzerland, for each country a male and a female one. The Austrian speakers were determined in an extensive selection process, the Swiss and the German speakers were professional speakers of the radio stations Südwestfunk in Germany and DRS in Switzerland.

The transcriptions in the database are manually obtained narrow phonetic transcriptions for each of the six spoken versions of a recorded word. Even though the dictionary itself claims that the transcriptions are not narrow, from a technical point of view they are, due to the large phonetic alphabet used in comparison with the the phonetic alphabet used in typical speech processing applications for German. This means that the transcriptions, in theory, are close to the actual realization of the words by the selected speakers. They do not necessarily comply with some standard transcription. However, as no officially acknowledged pronunciation standard for Austria exists, the approach of transcribing representative speakers is practical. The dictionary claims to represent a *media presentation standard* which is reflected in the selection of the six speakers.

17 transcribers were involved in the transcription process. When taking diacritics into account, more than 140 different phones can be found in the phonetic transcriptions. Some of them, however, occur very rarely. A reason for the large phonetic alphabet could be that the transcribers of the database did not restrict

Speaker	Different phones	Different phones (min. occurrence=5)
AT_M	143	86
AT_W	149	92
DE_M	140	90
DE_W	146	92
CH_M	139	81
CH_W	134	79

Table 4.1: Number of different symbols for each speaker in the ADABA and the number of phones that occur at least 5 times.

themselves to a particular phone set but used the unrestricted IPA symbol set (containing many diacritics and suprasegmentials) instead. Another reason may be that insufficient consistency was achieved among the 17 transcribers. This and other drawbacks have lead to some criticism in the literature after publication. In [Ehr09], a summary on this issue is given. The author's main points are inconsistencies, the used diacritics, wrong transcriptions and the usage of multiple correct transcriptions for the Austrian variant. She also criticizes the selection process for the Austrian speakers as being intransparent and heavily biased as the female speaker for the Austrian variant was a main supporter of the project too. The selection process of the Austrian speakers, however, *is* explained in the dictionary, where it is further mentioned, in which stages the female speaker was involved (in the pre-selection process). The final selection from 8 male and 9 female candidates was done by a listening survey via internet with 480 participants. As the selected female speaker is the head speaker of the Austrian public broadcaster and thus has many years of professional speaking experience, it is not unlikely that she is elected via such a survey.

Despite the criticized points, the project can be considered as the first attempt to provide a pronunciation dictionary for regional varieties of German from a pluricentric perspective and for Austrian German in particular on a large scale. In addition, the audio recordings provided in the database make it a useful resource for speech processing applications.

If only the phones that occur at least 5 times in the corpus are counted, the total number of different phones is 128, considering the transcriptions of all six speakers. Table 4.1 shows the number of different symbols for each speaker in the ADABA. `AT` is used for the Austrian, `DE` for the German and `CH` for the Swiss speakers, `M` denotes the male and `F` the female speakers. Some selected phone frequency differences between the ADABA and the Duden transcriptions are shown in table 4.2. One can observe major differences between the varieties in the usage of phonetic symbols. Further, several phonetic symbols not present in the Duden are used to emphasize certain pronunciation differences between the varieties.

A problem encountered with the transcriptions in the database was that they were not explicitly given for all words of all six speakers. Frequently, only the male Austrian transcription was complete while the others contained wildcards referring to another transcription. Table 4.3 shows some examples of word transcriptions that use wildcards for the Austrian, German and the Swiss variety. The symbols for the wildcards and their usage are inconsistent and

Phone	Duden	ÖAWB AT	ÖAWB DE
IPA	%	%	%
[ɛ]	4.02	0.07	2.79
[e]	1.97	2.80	2.65
[e̞]	0.00	2.81	0.26
[ə]	5.33	1.94	4.64
[ɘ]	0.00	1.89	0.93
[ɜ]	0.00	0.77	0.07
[ɪ]	3.18	0.02	0.65
[i]	3.15	6.47	5.53
[z]	1.54	0.02	1.36

Table 4.2: Relative frequency of selected phones in the phonetic transcriptions of the Duden and the ÖAWB. Differences in length and stress are ignored for these numbers.

Variety	ADABA line (SAMPA)
AT	abbauen [0];[[ap"ba_(o.@\n]]/[[...ba_(on=]]
AT	Abenteuer [0];[[a:b@\nto_(e6]]/[[a:bn=...]]
AT	zylindrig [0];[[tsi"lindRik]], [[-iC]], [[tsy"lin-]]
DE	dagegen [0];[["da:ge:gN=]]-[[da:"..]]/[[+]]-[[da:"...]]
DE	dabei [0];[["daba_(e]]-[[da"ba_(e]]/[[-]]-[[+]]
DE	Gelegenheit [0];[[g@"le:g@nha_(et_>]]/[[..gN..]]
DE	Hubschrauber [0];[["hupSRa_(oba]]/[["hub_0...b6]]
DE	Weltmeisterschaft [0];[["vEltma_(estaSaft]]/[[..t6..]]
CH	Scheibenwischer [0];[["Sa_(eb@\n%viSa]]/[[..bm=..]]

Table 4.3: Some examples how different wildcards are used in the ADABA transcriptions.

ambiguous. Thus an automatic completion of the transcriptions was impossible without incorporating further knowledge on the syllable and on the language level. In table 4.3, some problems are indicated. Symbols for placeholders include hyphens and arbitrary numbers of dots. They can refer to an alternative variant spoken by the same or by a different speaker. They can appear at the beginning, in the middle and at the end of a word or at multiple positions. When placeholders are used, the remaining, explicitly written phone symbols do not provide sufficient context for an automatic script when considering the phone level only. The symbols for separating speakers and variants of a word are not consistent (hyphens, commas, slashes) and sometimes ambiguous.

Nevertheless, some cases were identified where the intended meaning could be identified unambiguously using the wildcard information only and thus an automatic expansion via a script was applicable. For the remaining few thousand words, the transcriptions were manually completed by the author in a quite time-consuming process.

4.1 Characteristics of the Austrian variety in the ÖAWB

The ÖAWB contains transcriptions of three regional varieties of German with a clear focus on the Austrian variety. In chapter D of the ÖAWB several characteristics of the Austrian variety are claimed. The ones that are related to the six speakers of the isolated word corpus are reproduced here:

- the replacement of the vowels [ɪ], [ʏ] and [ʊ] with their closed forms [i], [y] and [u]
- a more closed realization of the vowels [ɛ], [ɔ] and [œ]; therefore the symbols [e̝], [o̝] and [ø̝] are used
- the usage of three different middle-positioned vowels ([ə], [ɵ] and [ɜ]), instead of only using a single schwa-vowel [ə]
- the realization of the German diphthongs as [o͡e] (and sometimes as [ø̝͡e]), [a͡o] and [a͡e] instead of using the Duden symbols [ɔ͡y], [a͡ʊ] and [a͡ɪ]
- reduced voicing or absence of voicing for the phoneme [z]; thus the symbols [s] and [s̬] are used
- a still strong but reduced aspiration of the plosive [t] compared to the other varieties
- numerous differences when pronouncing certain pre- and suffixes and the post-vocalic [r]; some of them can also be found in [Bau03], for a detailed list consider the ÖAWB [Muh07]

The Austrian specifics are reflected in different transcriptions of the ÖAWB word list. An exception are the diphthongs: Even though the different diphthong realization is claimed to be a characteristic of the Austrian variety in chapter D of the ÖAWB, the change of the diphthong symbols is also done for the transcriptions of the other varieties.

Chapter 5

Segmentation of the ADABA corpus

The previous chapter described the ADABA corpus that is used in this thesis. One goal of this thesis is an automatic segmentation of all six speakers in the isolated word corpus of the ADABA. During my employment at the company SYNVO, I worked with the SYNVO segmentation system that is also applied in this thesis. In this chapter, the segmentation system and necessary adaptations for the ADABA corpus are explained.

5.1 Choosing a segmentation system

The chosen approach for speech segmentation with pronunciation variants requires the speech signal and the text of the utterance as main input. The text is then converted to the canonical phonetic transcription by a text analysis module. This module analyzes the text as a whole and outputs the phonetic transcription including information on phrasing, stress and intonation. The internal details of the module are not relevant for this thesis, however, the generated phonetic transcription is based on the Aussprache-Duden [MI00].

To account for pronunciation variants, rules describing variability of the pronunciation are used. Applied in the segmentation process, this rules generate multiple pronunciation variants of the text. All generated variants are then considered in the alignment process. The method describes possible pronunciation variants as variations to the canonical phonetic transcription of an utterance. An alternative would be to generate the different variants based on the orthographic transcription.

The used speech segmentation system is based on the approach described in [RP05]. In this paper an automatic speech segmentation system for mixed-lingual prosody databases is developed with the goal of replacing a manual segmentation process. This system is HMM-based and uses an iterative forced-alignment approach to successively improve the segmentation quality. Initialization of the HMMs is done with a flat-start approach, using the canonical phonetic transcription of the utterances. Thus no initial segmentation is needed. Subsequently, in each new iteration, embedded re-estimation of the HMMs with the transcription of the last iteration is done. In the first iteration, the canonical

phonetic transcription is used. The rule-based approach that is used to model pronunciation variation is also able to deal with inter-word phenomena.

The segmentation system described in [RP05] was enhanced in the last years. Initialization is not done with the flat-start approach anymore but by using an initial segmentation derived from a forced-alignment with speaker-independent models. This leads to a better initial segmentation. The speaker-independent models used for the initial segmentation step in this thesis are part of the SYNVO segmentation system.

In the segmentation system, after each iteration, the segmentation and labeling is corrected in a post-alignment step by using additional features. A speech type segmentation is done to extract features about the voicing or silence of segments, resulting in the classification of segments as *voiced*, *unvoiced*, *silence* or *irregular*. For the voiced segments, a period segmentation is performed. The details of this period segmentation are explained in [Rom12]. This additional segmential features are then used as input to the correction step after the forced alignment. The performed corrections are configurable and depend on the expected quality of the segmentation after the forced alignment and the expected quality of the additional features. The possible post-alignment corrections in the segmentation system are:

Period alignment: The boundaries of voiced segments are aligned with the period segmentation.

Segment deletion: Very short segments with a small loglikelihood are deleted.

Voicing correction: Voiced plosives can be replaced by their unvoiced counterparts based on the voicing information in the additional features.

Silence correction: Silence segments resulting from the forced alignment are trimmed to the silence boundaries in the speech type segmentation.

Align speech types: The segment boundaries are aligned with the boundaries of the speech type segmentation.

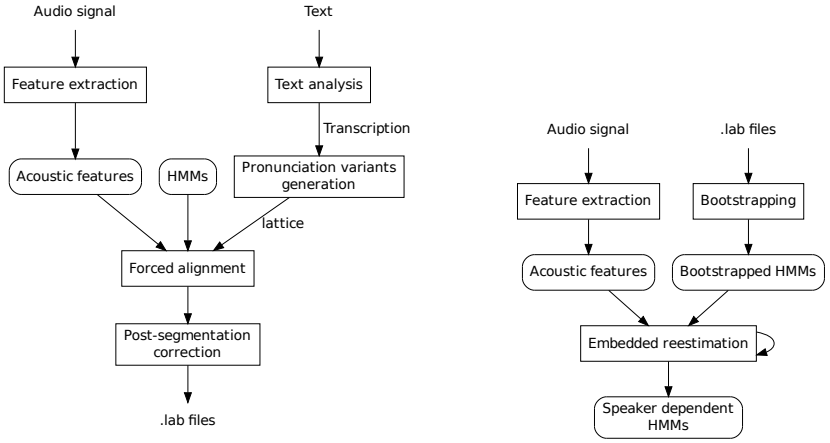
Preplosive pause insertion: Plosives are split into two labels: A closure segment and a burst segment.

Glottal closure correction: Glottal closures resulting from the forced alignment are deleted if the speech type segmentation shows no evidence for them.

Breathing correction: Before and after breath segments, a silence can be inserted based on the speech type segmentation.

Figure 5.1 shows an overview of the segmentation system.

This system works quite well for the automatic segmentation of corpora. Moreover, by using just the initial iteration with speaker-independent models it is possible to segment single utterances for new speakers even if no corpus is available for these speakers. Needless to say, the segmentation accuracy achieved by a forced-alignment procedure with speaker-independent models can not be expected to be as accurate as the iterative procedure that is possible if more data is available. In combination with the post-aligment boundary correction step, however, considerable results can be achieved.



(a) General structure of an iteration.

(b) Estimating HMMs for subsequent iterations via embedded re-estimation.

Figure 5.1: Overview of the segmentation system. In the first iteration, the HMMs are existing speaker-independent HMMs. In subsequent iterations they are estimated via embedded re-estimation.

The segmentation framework makes use of the Hidden Markov Model Toolkit (HTK, see [YEG+06]). The speech signal and the canonical phonetic transcription of the utterance are assumed to be given (the latter is generated from the text by the text analysis module). The additional features mentioned above (the speech type segmentation and the period segmentation) are extracted in advance. In the first iteration step, acoustic features are extracted from the speech signal. The feature extraction is performed in every iteration. In the first iteration, the features have to match the speaker-independent models used. In the subsequent steps, the features have to match the models trained from scratch with the data to segment. For further iterations, however, already extracted features with the same configuration are reused. The acoustic features in the first iteration are MFCCs with 13 cepstral coefficients (including the 0th coefficient), delta and acceleration coefficients, resulting in a total number of 39 features. Cepstral mean normalisation is performed on the features. The window size is 25 milliseconds and the frame shift is 10 milliseconds. For further iterations, user-configured features can be used. The default is to use MFCCs with 24 cepstral coefficents, the log energy and their deltas.

In the next step, pronunciation variants of the utterances are generated according to iteration-specific pronunciation rules. The generated variants are represented in the Standard Lattice Format (SLF) of HTK. Starting from the second iteration, bootstrapping and embedded re-estimation of new models is done. Then, forced alignment is performed using the HTK-tool `HVite` with the variant lattice and the features as input. The forced-alignment results in the maximum-likelihood segmentation and labeling. This segmentation and labeling is passed to the post-alignment correction that corrects the boundaries by incorporating the additional features extracted before the iteration loop.

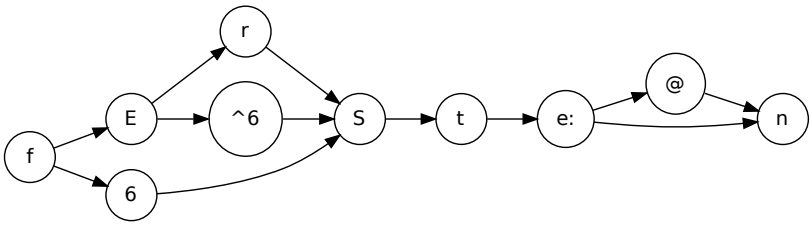


Figure 5.2: Pronunciation lattice of the word *verstehen*.

The algorithm described in this chapter is able to perform automatic speech segmentation of larger corpora as well as of under-resourced material such as a single utterance. During preliminary experiments it was observed that the boundaries for the unvoiced segments were significantly better than for the voiced ones. The speech type features show no new evidence on voiced-voiced boundaries and thus no further correction in the post-alignment step could be performed. When the framework is applied to a corpus with regional pronunciation variants as done in this thesis, further difficulties occur. First, pronunciation variation rules specific for the variants are missing and second, the phone inventory of the segmentation algorithm may not match that of the variants.

5.2 Pronunciation variation modeling

Dealing with variability of speech is essential for a good automatic segmentation and labeling system. The variability caused by regional variants is not covered by a standard system that does not model the actual kind of variation. In section 3.4, two possibilities to deal with pronunciation variation were described: knowledge-based and data-driven. Generic pronunciation rules model some of the variation that appears in regional language variants. However, phenomena that are specific to a variation are not covered by such rules. Therefore, rules for the new variant have to be identified in advance.

In this thesis, a rule-based system is used. The generic rules from the segmentation framework are extended with variant-specific rules from the literature. The rule-based system converts the canonical transcription to a variant lattice which is then used during the forced alignment. The paths though this lattice represent possible realizations of the utterance. As an example, assume that the word *verstehen* (engl. to understand) has the canonical phonetic transcription /fɛrʃteːən/ and several pronunciation variants for the first syllable: [fɛr], [fɛɐ̯] and [fɐ]. Assume further that the schwa vowel in the last syllable is optional. Then the pronunciation variant lattice looks like in figure 5.2.

Before the forced alignment is performed, the phone symbols are mapped to the model names. If a symbol has no corresponding model, a mapping to a similar model (or a sequence of models, e.g. for diphthongs) is done. The mapping is performed by a predefined mapping file. This file has entries for all expected phones and matches them with existing models. If a new phone occurs

during the segmentation, the user is asked to add a new entry in the mapping file.

The conversion from the canonical transcription to the variant lattice is done using so-called two-level rules [Kos83]. Two-level rules build a relation between a lexical form and a surface form. In this context, the lexical form is the canonical pronunciation and the surface form is a realized pronunciation. All phones in two level rules are written in SYNVOPA notation (see appendix A). An example for a two-level rule is:

```
'C'/'k' => 'I' _ '#' ;
```

This rule is interpreted as follows: On the left side of the operator `=>`, the lexical/surface replacement is defined, and on the right side, the context for the replacement is described. The above rule means that a [ç] (in the lexical form) preceded by [ɪ] on a word end (here represented by the symbol '#') can optionally be realized as a [k] (in the surface form). This is a common rule for German pronunciation. For example, the word `wenig` (meaning `few` in English) can be pronounced as [veːniç] or as [veːnik]. The first part of the rule (`'C'/'k'`) specifies the lexical and the surface forms, separated by the slash symbol `/`. In the rule above, this part is followed by the operator `=>` and the context specification (here `'I' _ '#'`). In this context string, the symbol `_` is a placeholder for the considered symbol pair from the left side of the operator. The placeholder is preceded by its left context (here `'I'`) and followed by its right context (here the word boundary symbol `'#'`). The symbols used for the left and right context can also specify the lexical and the surface form, just like it is done on the left side of the rule. It is thus possible to define a left context like `'i:'/'I'` which refers to the symbol `i:` in the lexical and `I` in the surface form. The operator defines how the rule is applied. Possible operators are:

`=>` The modification on the left side implies the specified context. That means that the replacement is only possible in this context. However, the specified context also permits other forms. This operator can be used to define optional replacements.

`<=` The context implies the modification. No other modification is possible in this context. The same modification, however, can be triggered by other contexts.

`<=>` The modification implies the context and the context implies the modification. This is a combination of the above operators. It means that the specified replacement is applied if and only if the given context matches and that the given context must lead to this and only to this replacement.

Another example for a two-level rule is gemination, i.e. the lengthening of consonants when they occur doubled in the orthographic transcription. For example, the rule:

```
's'/@ => _ '#'/? ('s'|'S'|'z'|'Z');
```

means that the phone [s] can optionally be deleted at word ends if it is followed by another word starting with [s], [z], [ʃ] or [ʒ].

In the segmentation framework, all specified two-level rules are compiled into a finite state transducer. The transducer is then used to generate pronunciation variants from the canonical phonetic transcription.

Speaker	Transcription
Austrian female	apˈbiːgn̩
Austrian male	apˈbiːgən
German female	apˈbiːgən
German male	apˈbiːgən
Swiss female	ʔap͡ˈpiːgən
Swiss male	ˈʔaˌpiːgən

Table 5.1: Pronunciation variants of the German word `abbiegen`.

A data-driven extraction of knowledge for a regional language variant means that rules should be generated from available data and subsequently applied to consider new variants. How this is done is highly dependent on which and how much data is available. In the ADABA corpus used for this thesis, for several words, more than one pronunciation variant can be observed per region or even per speaker. For example, the German word `abbiegen` (to turn [left or right]) has the phonetic transcriptions (according to the ADABA) for the six speakers as shown in table 5.1 in IPA notation.

From this variant instances it would be possible to derive new variants for the same word by incorporating for each speaker also the variations that others made. For example, the ending [gn̩] that the Austrian female speaker made could also be considered for the other speakers in an automatic segmentation and labeling process. Thus one variant for the swiss female speaker would become: [ʔap͡ˈpiːgn̩]. Nevertheless, this approach was not applied in this thesis, as the decision was made to use the Aussprache-Duden transcription instead of the ADABA transcription as starting point.

To use the initial segmentation framework described here for the ADABA corpus, some adaptations were necessary. Additionally, in parallel to the work on this thesis, a software-tool that facilitates the segmentation process was developed by the author at the company SYNVO. Its development as well as the adaptations of the initial framework are treated in the next sections.

5.3 Applying the segmenation framework to the ADABA corpus

The initial segmentation framework is already capable of performing automatic phonetic alignment of a corpus. Nevertheless, segmenting the ADABA corpus with this segmentation framework poses some new challenges. The phone set of the framework is smaller than the ADABA phone set. In chapter 4 it was mentioned that the ADABA comes with a very narrow transcription, using more than 140 different phones. Thus, definitions for the additional phones are added. Further, a mapping of all phones to the phoneset of the speaker independent models used for the initial segmentation is defined.

The rules of the segmentation framework for generating German pronunciation variants are not optimally suited for the ADABA. One might think that the narrow phonetic transcriptions of the ADABA corpus make the use of pronunciation variation rules redundant. However, the mentioned drawbacks of

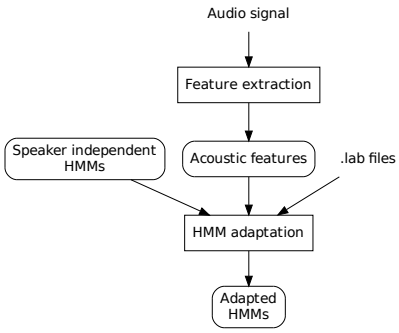


Figure 5.3: HMM adaptation.

the transcriptions in chapter 4 and manual verification of some transcriptions of the corpus indicate that the provided phonetic transcriptions should not be considered as error-free. Consistency among 17 manual transcribers is hard to achieve, especially when using such a large phone inventory as done in the ADABA corpus. An automatic approach that starts from a different canonical phonetic transcription and incorporates pronunciation variation rules can be consulted in a consistency and correctness check of the manual transcriptions. Hence well suited pronunciation rules are desirable.

In [Bau03], several systematic differences of Austrian German to the canonical pronunciation were identified. The differences are used in this thesis to extend the pronunciation rules.

An additional improvement can be achieved when the speaker-independent HMMs of the first iteration step are adapted to the characteristics of the speaker of the current corpus. This can be done via a technique called *Speaker adaptation*. Speaker adaptation uses data of a target speaker and changes the model parameters to better fit this adaptation data. As opposed to the training of speaker-dependent models, fewer data is needed to perform speaker adaptation. Adaptation can be done in supervised or unsupervised mode, depending on the availability of labeled adaptation data. Supervised adaptation uses the labeled adaptation data to adapt the models before using them in the decoder. Unsupervised adaptation can be done directly during the decoding. The unsupervised approach has the advantage that it can be used online during the decoding of a new speaker without the need for previously available material for this speaker. Nevertheless, if applicable, the supervised approach leads to better results. As the automatic phonetic segmentation of corpora is done offline, supervised adaptation is the best choice for this task. The approach is implemented using the HTK, and uses the labeling resulting from the initial segmentation step as a basis. Figure 5.3 shows the model adaptation steps. The `.lab` files denote the segmentation from the initial iteration, using the speaker-independent HMMs. For details on the adaptation process, see e.g. the HTK book [YEG+06].

As mentioned before, in the framework used for segmentation, speech type features are used to correct the segmentation boundaries in a post-correction step. This usually leads to an improvement of the segmentation accuracy. How-

ever, this approach also had some disadvantages. The movement of boundaries based on local information only (as opposed to the forced alignment, which always considers the overall cost of a possible alignment path) can lead to more serious errors when the speech type segmentation is wrong. While in many cases the post-correction improves the quality of the segmentation, in some cases, the segment boundaries are better without this step, after the forced alignment only. Thus an approach that better combines the forced alignment with its overall cost minimization on one hand, and the speech type features on the other hand, is desirable.

Other existing systems, some of them already mentioned in chapter 3, move away from the traditional way of using a GMM for obtaining the observation probabilities of a speech frame given a particular model (as it is implemented in popular speech recognition software, like HTK). Often a HMM/ANN hybrid is used instead of the HMM/GMM approach (e.g. in [Hos09]). This means that ANNs are used to estimate the observation probabilities.

This leads to the idea of directly using the speech type features in the forced alignment step. There these features can be considered as additional observations and therefore contribute to the observation probabilities. As the HTK tool `HVite` is used for the forced alignment, the options to achieve this are to either use the (limited) capabilities of additional feature streams, to change HTK's source code or to re-implement the decoding from scratch. For more flexibility and due to the lack of documentation for the HTK source code, the last approach was chosen. The decoding is re-implemented in Java. A token-passing algorithm (see [YRT89]) was used to allow the forced alignment on a variant lattice. Attention is paid to be compatible with the HTK formats. Thus the inputs to the decoder are still a variant lattice in HTK's SLF format and HMMs in HTK's model format, and the output is a HTK label file. When calculating the observation probabilities for a specific feature frame and model, the speech type features are now taken into account.

First, for each phone, a probability that the speech type is one of its four possible values (*voiced*, *unvoiced*, *irregular* or *silence*) is estimated. During decoding, when considering each HMM, the observation probability resulting from the GMM is multiplied with the probability resulting from the speech type and the phone for the considered HMM. This product can be interpreted as the joint probability of the two when assuming independence.

5.4 The speech segmentation tool

As an employee at the SYNVO company, the author developed for SYNVO a speech segmentation tool with a GUI that simplifies the segmentation process. The tool is also used to perform experiments for this thesis. Therefore, a short summary on it is given here. The requirements for the tool are:

- port the segmentation framework from a shell script based workflow to Java, and use of external calls for tools like HTK if appropriate
- automate and simplify the automatic segmentation process of a corpus
- manage utterances and corpora in an explorer-like navigation view

- display the waveform, spectrogram of utterances; load and display several labelings of segmented utterances
- implement zoom and scroll functions for the shown utterances
- allow the manual editing and saving of segment labels and their boundaries
- integrate an error-log to simplify debugging of segmentation errors
- manage configurations for segmentations and segmentation iterations
- play/pause/continue the playing of utterances and segments of them
- show the variant lattice and the best selected path during the forced alignment
- implement a plugin-mechanism that permits an convenient extension possibility for the tool

It was decided to develop in Java and to use the Eclipse Rich Client Platform (Eclipse RCP) as basic framework for the software. The Eclipse RCP allows developers to base their end-user applications on the same framework as the Eclipse Integrated Development Environment. It contains the following components [ML05]:

- Eclipse Equinox, an implemenation of the Open Services Gateway interfaces (OSGi) specification [WHKL08],
- the eclipse core runtime
- the Standard Widget Toolkit (SWT), a graphical widget toolkit that uses the platform's native GUI compontents and thus provides a native look and feel to the end user
- a plugin-mechanism
- JFace, which adds more abstraction to the SWT and provides components like dialogs, viewers, wizards, etc.

Based on Java and Eclipse RCP, the software can run on multiple operating systems and only requires the Java Runtime Environment to be installed. To use the automatic segmentation functionality, however, HTK and Matlab must be installed.

5.4.1 System architecture

The high-level architecture of the software is shown in figure 5.4. Only the most important modules are shown. Modules of the software roughly correspond with Java packages. Some helper classes are spread across multiple packages.

Segmentation control logic: The control logic implements the segmentation algorithm. Parallel execution of most sub-tasks is possible. The module is divided into the master control logic and the individual segmentation subtasks.

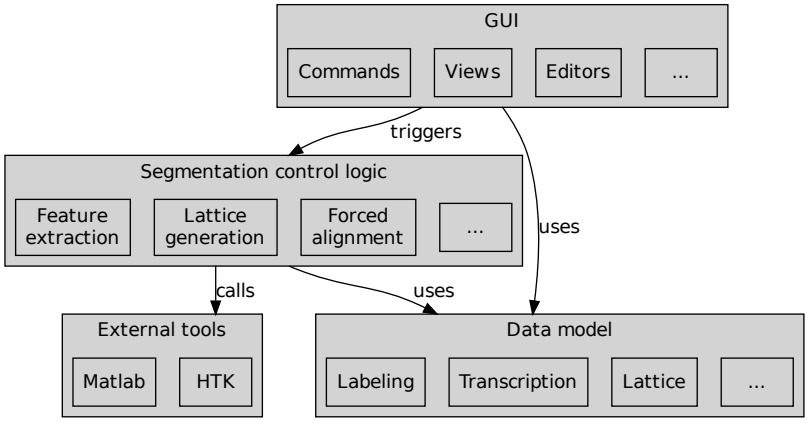


Figure 5.4: The main modules of the segmentation tool.

Commands: Contains classes for the sub-steps of one segmentation iteration. This can be the feature extraction step, the embedded re-estimation step or the post-processing step of the segmentation framework.

Data model: Representation of resources used in the software, like utterances, corpora, etc.

IO: Parsers and writers of different file formats such as HTK label files, HTK model files, feature files, etc.

GUI: Consists of the submodules

Commands: Contains executable Eclipse RCP Command classes that implement actions triggered by the GUI.

Views: Eclipse RCP Views for displaying data without providing editor capabilities. An example is the view for the pronunciation lattice.

Editors: Eclipse RCP Editors such as the Utterance editor that is responsible for displaying the waveform, spectrogram and label annotations an that permits the editing of label boundaries.

Navigations: Eclipse RCP views such as the CorpusExplorer that displays the different corpora and utterances.

Tools and helper classes: Several general purpose classes, e.g. for audio playback or logging functionality.

A central part of the software is the Utterance editor. It displays the acoustical waveform, the spectrogram and different label annotations. Zoom functions for the waveform in horizontal and vertical direction are implemented. Different labelings (i.e. segmentations) can be loaded for an utterance. A user can

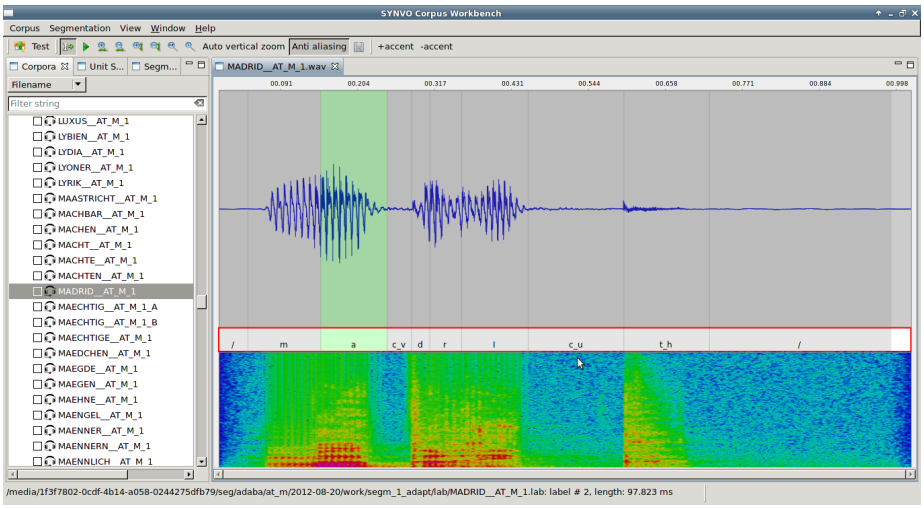


Figure 5.5: The segmentation software SYVNO Corpus Workbench. The corpus `AT_M` is loaded and its utterances are listed on the left. One utterance is shown in the editor along with a phonetic segmentation and the spectrogram. The phonetic labels are in SYNVOPA notation.

select individual segments (phones) and listen to them. The segments and their boundaries can be edited. The user can explore the waveform, the spectrogram and the fundamental frequency curve and compare different segmentations.

5.4.2 Usage

Figure 5.5 shows a screenshot of the software. A corpus is loaded and its utterances are listed in the Corpus explorer on the left. The currently opened utterance (`MADRID_AT_M_1`) is displayed in the main area of the program. A segmentation is loaded and the phone segment a is selected. The spectrogram is displayed at the bottom of the program's window.

Chapter 6

Vowel classification

The previous chapters explain how to build a speech segmentation system and how this system can handle pronunciation variation. Nevertheless, some mentioned problems when dealing with regional pronunciation variants remain. The standard phone inventory of the segmentation system may not be appropriate for the variant considered. The pronunciation variation module may not detect certain phone variants due the limited number of phone models. If only few data per phone is available for a regional variant, no new phone models can be trained and no general pronunciation rules can be extracted from the data. In this case, a phone classifier can help in supporting the detection of the right pronunciation variant. Preliminary experiments showed that the system described in chapter 5 is able to deal quite well with consonants of regional variants given the standard phone inventory and pronunciation rules. For vowels, however, there is room for improvement. Therefore, the focus of the phone classification system will be on vowels.

6.1 Articulatory features for classification

Several studies propose that the use of knowledge of the speech production process should be able to improve the performance of speech processing systems. In [KFL+07] a comprehensive overview of work on incorporating speech production knowledge into ASR systems is given. Articulatory features as described in section 2.1.1 are examples for speech production based features. They have a direct relation to the phonetic content of an utterance and promise to be very helpful for speech processing applications. The drawback is, however, that they usually can not be observed directly.

In [Kir99] an attempt is made to incorporate articulatory features in ASR, including a pilot study for small and large vocabulary recognition. A number of independent multilayer perceptrons (MLPs) is used to estimate five different articulatory features: *voicing*, *manner* of articulation, *place* of articulation, the relative *position of the tongue* on the front-back axis and *lip rounding*.

For vowels, the articulatory features are the *tongue position* and the *lip rounding*. The tongue position is described by the terms *tongue backness* and *tongue height*. These features are reflected in the axes of the IPA vowel quadrilateral In figure 6.1, the vowel quadrilateral as defined by the IPA is shown.

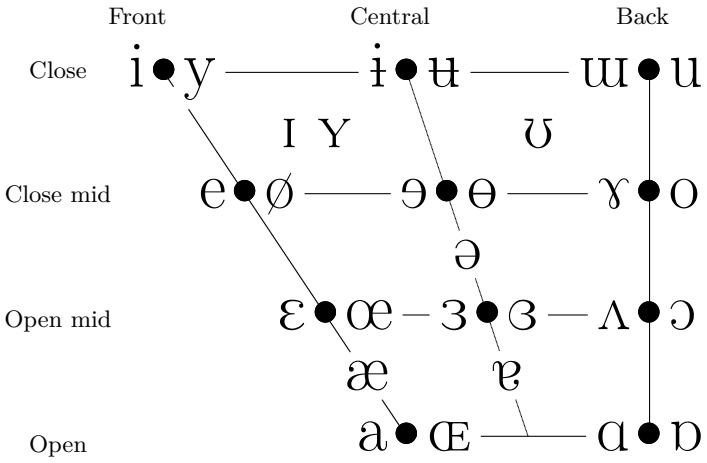


Figure 6.1: Vowel chart according to the International Phonetic Association. The vowels to the left of a point are unrounded, the ones to the right are rounded.

The descriptions along the axes of the IPA chart indicates that the location of the vowels in the chart is related to the tongue position. The x-axis is labeled with the three positions *Front*, *Central* and *Back*, the y-axis is labeled with the four positions *Open*, *Open-mid*, *Close-mid* and *Close*. The tongue height thus ranges from close to open, the backness from front to back. The lip rounding is not directly indicated by the position in the chart. If, however, a rounded and an unrounded vowel exist for one position, the right one is always the rounded version and the left one is unrounded. The chart represents a quite abstract description of the tongue's position as the absolute and relative positions are not covered exactly [AC99].

Classifying vowels by using the isolated acoustic signal segment for that vowel only (i.e. without having the context) is difficult. Even humans make confusions in this case [PB52].

The acoustic features *formants* have been described in section 2.4.2. There is an interesting relation between the articulatory features *tongue position* and the formants. [Joo48] noted that there is a notable match between the vowel quadrilateral and the first two formants. Example ranges for formant values of various speakers are shown in table 6.1. The table shows typical values of the first two formants for some vowels. The values are based on the work in [Wee06] and are reproduced here to give an idea of the formant range for different vowels. The differences between male and female speakers can clearly be observed. Additionally, there is a high inter-speaker variability, also between speakers of the same sex. The high variability together with the problems of their robust estimation makes formants not optimally suited to use as sole features for a vowel classification task. They can, however, be used as additional features.

If the first two formant frequencies F_1 and F_2 are plotted (F_2 on the x-axis and F_1 on the y-axis, running from right to left and top to down for increasing values), the similarity to the IPA vowel quadrilateral becomes obvious. Sometimes the difference of the two formants ($F_2 - F_1$) is used instead of F_2 on the

Vowel	F1 (male)	F1 (female)	F2 (male)	F2 (female)
u	275–361	305–448	560–748	578–842
a	696–921	692–1047	1244-1478	1333–1695
o	392–557	457–608	692–961	857–1109
ɑ	642–723	596–964	988–1218	1023–1386
ø	357–505	443–579	1385–1675	1529–1917
i	255–318	268–374	1984–2390	2060–2873
y	248–348	282–436	1504–1846	1391–2134
e	357–509	435–559	1748–2198	1950–2642
ʏ	388–499	397–549	1307–1639	1630–1900
ɛ	502–645	495–787	1679–2024	1897–2326
ɔ	395–531	462–785	662–837	690–1010
ɪ	353–429	369–511	1888–2269	2217–2636

Table 6.1: Typical ranges for the first two formants (extracted from 10 male and 10 female speakers from [Wee06]).

x-axis. An example for such a plot can be found in the evaluation section of this thesis in figure 7.4.

It has already been mentioned that a direct observation of articulatory features is usually not feasible. Nevertheless, databases exist, where such features have been recorded in addition to the acoustic signals by means of a laryngograph and/or an electromagnetic articulograph. An example is the MOCHA database [WH00]. For many applications, however, direct measurement is not practical. Therefore, prior to the usage of articulatory vowel features in a speech processing application, these features must be estimated from the acoustic signal. The relation between the acoustic speech signal and the articulatory features can be non-linear and non-unique [NAE08].

The estimation of the articulatory features *tongue position* and *lip rounding* is a feature-transformation with dimensionality reduction. Acoustic features, which usually have more than 10 dimensions (such as MFCCs) are transformed to articulatory features with three dimensions (tongue height, tongue backness and lip rounding). It is hypothesized that the new features allow for a better classification of the vowels as they a) represent production-oriented principal components of the phone and b) have a straightforward interpretation in the articulatory space.

6.2 The multilayer perceptron

For different experiments performed in this thesis a so-called multilayer perceptron (MLP) is used. A MLP is a special kind of an artificial neural network (ANN). ANNs are computational models inspired by the structure of the human brain and are used in machine learning. They are structured as directed graphs where the node elements are called neurons or perceptrons. A neuron transforms a linear combination of its input values to an output value by means of a so-called activation function. The coefficients of the linear combination are called *weights*.

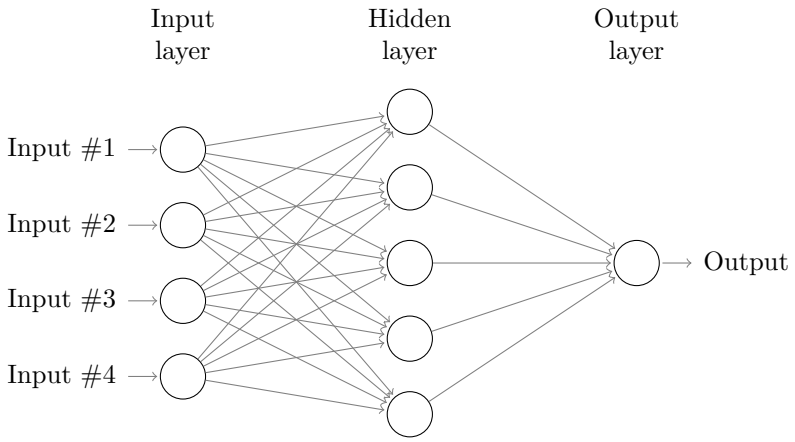


Figure 6.2: Multilayer perceptron with one hidden layer.

A MLP is a feedforward ANN consisting of several layers. The nodes in each layer share the same inputs. Two subsequent layers are fully connected, i.e. each output node of the first layer is connected to each input node of the next layer. Figure 6.2 shows a MLP with four input units, five hidden units and one output unit.

Design parameters of an MLP are the number of layers, the number of nodes in each layer and the activation function for the nodes. Parameters that are adapted during the learning phase of an MLP are the set of all neurons' weights and biases.

MLPs are often applied to machine learning problems that require supervised learning. This means that during training, input examples are presented to the inputs and corresponding target examples are presented to the outputs. Thus correctly labeled examples are needed for supervised learning. For unsupervised learning problems, on the other hand, no labeled examples are needed. Unsupervised learning finds patterns or clusters in the data.

MLPs can approximate arbitrary non-linear functions between inputs and outputs. The output nodes have a continuous value range. This makes MLPs well suited for regression problems, where the target value space is continuous. However, they may also be applied to classification problems, where the target value space is discrete. In this case, the network is designed in a way that each output node estimates the a-posteriori probability for a particular class (and all output values sum up to one).

Training an MLP is an iterative process. During each training step, the weights are adapted in a way that the error between its output values and the presented target values decreases. The process is repeated until a previously specified stopping-criteria is satisfied. The error is measured via an objective function. During training, the goal is to minimize the objective function.

Once trained, the MLP is usually applied to new input data, i.e. data, that has not been presented during training. A well-performing MLP should be able to *generalize* to new data. This means that the output values for previously unseen input values are a good approximation of the true target values. This implies that the MLP does not just memorize the training examples but learns

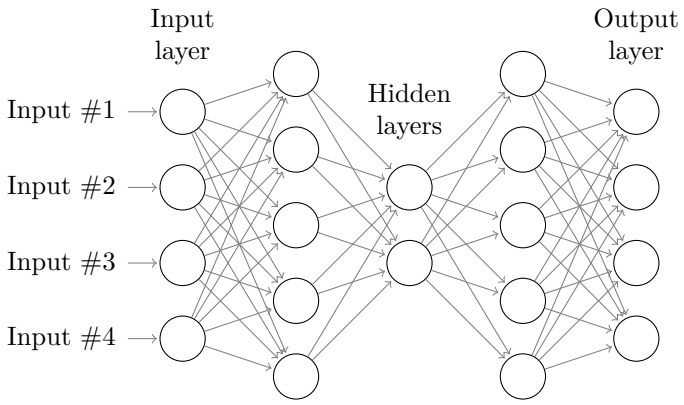


Figure 6.3: Topology of an autoassociative neural network.

an underlying function instead. The effect of a lack in generalization by only memorizing the training examples is commonly referred to as *overfitting*. To ensure proper generalization, a subset of the data, the evaluation set, is left out from the minimization of the objective function in the training process. Then, after each iteration step, the objective function is not only evaluated on the training set, but also on the evaluation set to estimate the performance of the MLP on unseen data. When, after several steps of training, the error on the evaluation set increases, the training is stopped to prevent overfitting. Nevertheless it is possible that overfitting on the *evaluation set* takes place, as it is not completely left out during training but used to evaluate the performance after each step, even if the influence of the evaluation set during optimization should be weak. Therefore another independent subset of the data, called the *test set*, is left out to assess the performance of different MLP that have finished training.

The selection of a proper minimization algorithm is another important aspect when training an MLP. An algorithm called *standard backpropagation* in combination with gradient descent for optimization is often used for this purpose. However, in many cases, other non-linear optimization techniques, like for example scaled conjugate gradient, might be more helpful [Sar97].

A special flavor of neural networks are so-called autoassociative neural networks [Kra92]. Figure 6.3 shows the basic topology of such a network.

These networks are able to learn in an unsupervised manner. No problem-specific labeled examples for the outputs are needed to train them. Instead, during training, the input values are presented to the inputs *and* the outputs of the network. Thus the network learns to map the input values to themselves. The network must have the same number of output nodes as it has input nodes. Of course, the fact that the network is able to predict the same values that are given as inputs does not make much sense alone. The interesting capability of autoassociative neural networks lie in their hidden layers. Such a network has at least three hidden layers. The middle one is the one with the smallest number of nodes in the network and thus enforces a compression of the input data before the mapping to the output nodes is done. It has been shown that these networks are able to perform a nonlinear principal component analysis [Kra91]. The

reduced data components are present at the outputs of the middle hidden layer nodes. The number of nodes in this layer determines the number of nonlinear principal components. Thus these networks are well suited for dimensionality reduction.

In this thesis, articulatory features that were mentioned in section 2.1.1 and 6.1 are estimated by using an MLP. It is analyzed whether they can help in analyzing the ADABA corpus and if they can be used to improve a speech segmentation system. In addition, an autoassociative neural network is used to estimate non-linear principal components of acoustic features. These principal components are then compared to the articulatory features. The next section explains the feature estimation in detail.

6.3 Feature transformation using a multilayer perceptron

Artificial neural networks have been used successfully in many machine learning and pattern recognition applications. For details on ANNs, refer to e.g. [Bis96]. In this work, it is proposed to use a MLP for the transformation of an acoustic feature stream (e.g. MFCCs) to articulatory features. For vowels, the articulatory features are tongue height, tongue backness and lip rounding, for consonants, they are manner and place of articulation.

The motivation of experimenting with a MLP for the estimation of articulatory features is as follows: The mapping between the acoustic and the articulatory space is considered to be non-linear [NAE08]. MLPs are able to deal well with regression problems for arbitrary non-linear target functions. Further, other studies suggest the application of MLPs for this task too. In the work [MNEW$^+$10], trajectory mixture density networks (TMDNs), feedforward artificial neural networks (FF-ANNs), support vector regression (SVR), autoregressive artificial neural networks (AR-ANNs) and distal supervised learning (DSL) are evaluated on the task of what the authors call *speech-inversion*, i.e. recovering articulatory information from the speech signal. In this study, a FF-ANN performs best for estimating vocal tract variables from acoustics. The authors mention that the inferred features contain estimation noise. Thus the recovered articulatory traces should be smoothed somehow in a postprocessing step. These traces are a time series of articulatory parameters and contain uncertainty. Additional information is available from physical constraints: Due to physical limitations, the actual articulatory traces are not able to change faster as around 15Hz. This information can be incorporated to reduce the uncertainty. In the cited study, a Kalman-filter is used for the smoothing. Kalman-filtering is a typical solution for the problem described, i.e. a physical model that predicts the values in addition with a series of uncertain measurements.

Using ANNs for articulatory feature estimation is the application of a supervised learning method. For a supervised learning approach, correctly labeled training data is required. For the current task this would require a training set of examples that have time-aligned articulatory features on the audio signal. In many cases, however, such data is not available because the articulatory features are not directly observable (for this reason a ANN is used to estimate them). Thus the only possibilities are either to use training data that comes

with recorded articulatory features as mentioned above or to derive the target values approximately from a previous phonetic segmentation. For practical reasons, the latter approach was chosen for this thesis. When using this approach, one has to bear in mind that the phonetic label stays the same for the entire phone segment, while the articulatory features are likely to change.

In related work where articulatory feature estimation is done by ANNs, often a separate MLP with only one output node is used for each feature (e.g. in [Kir99]). This is partly motivated by the reduced complexity of each individual network, partly by the argument, that independent features are modeled better by independent MLPs. As in this thesis a strong focus is on articulatory vowel features, it is checked if this assumptions still hold. For vowels, there are only the three articulatory features tongue height, tongue backness and lip rounding. The MLP is not expected to become too complex when using only three output nodes. Further, the independence of the features, particularly the two for the tongue position, can be questioned. Open vowels are more restricted in the range of their horizontal tongue position than closed vowels, which can easily be seen when looking at the IPA vowel quadrilateral in figure 6.1. Thus there is obviously a dependence between these two features. Preliminary experiments showed that a single MLP with three output nodes is well suited for this task and therefore, this approach was chosen.

6.4 Classification

Estimating the articulatory features results in probabilities for the different articulatory feature classes if discrete features are used, or in points in a continuous feature space if continuous articulatory feature values are used. In either case, in order to perform phone classification, the estimated articulatory features must be combined to result in a single phonetic label. This can be done in various ways. If a continuous feature space is used, a possibility is to measure the distance to each possible phone of the considered phone inventory in the articulatory feature space. As this distance can be measured for each feature frame, averaging over the phone's length can be done. An alternative to simple averaging is to model the time trajectories in the articulatory feature space. This could be done via a DTW-comparison with a reference phone in the articulatory feature space or with HMMs trained on the articulatory features. Another option to combine the estimated articulatory features to phones is to use MLPs again. This approach is chosen in [Kir99]. The inputs are the probabilities of discrete articulatory features and the targets are the probabilities for the different phonetic classes.

In this thesis, however, the used feature space for vowels is continuous. The outputs of the articulatory-feature MLP define a geometric point in this feature space. The reference phonetic classes (the vowels) can be defined as points in this space too. Trajectories in this space are interpreted as estimations of the movements of the articulators tongue position and lip rounding. They may carry important information about the phonetic quality, especially for diphthongs. For single vowels, however, we have the hypothesis that the average position in this feature space is a sufficiently good estimation for the perceived vowel class. If this assumption holds, this approach has the advantage that it is not necessary to have a trajectory model for each possible vowel in this feature

space. Further, if it is assumed that the feature space is euclidean and that geometric distances in this articulatory space are correlated to different perceptions of vowel quality, then no modeling of the articulatory feature distribution needs to be done and vowels can be identified by comparing the euclidean distances to the nominal vowel positions and choosing the closest one to label the vowel. The reference models reduce to points represented by the nominal values (or the means/medians) of a vowel class. Variation to this points represents perceptional difference and the closest reference point can be assumed to define the vowel class. The euclidean distance is calculated as follows:

$$d_e = \sqrt{\sum_{i=1}^{n}(x_i - \mu_i)^2}$$

x_i represents the ith articulatory feature component of the considered feature value and μ_i is the mean or nominal value of this component for the considered vowel class. However, if the vowel clusters are assumed to have different variability, another distance measure like the Mahalanobis distance might be used as well:

$$d_m = \sqrt{(\boldsymbol{x} - \boldsymbol{\mu})^T \cdot \boldsymbol{\Sigma}^{-1} \cdot (\boldsymbol{x} - \boldsymbol{\mu})}$$

In this equation, $\boldsymbol{\mu}$ represents the mean value vector of the respective vowel class and $\boldsymbol{\Sigma}$ equals the covariance matrix of this vowel's articulatory feature values. The Mahalnanobis distance accounts for different covariance matrices of the individual classes when calculating the distance. For example, consider the following situation: An observed point has the same euclidean distance to two cluster centers. One of this clusters has a large variance whereas the other one has a relatively small variance with most of the cluster samples close to the mean. Then the Mahalanobis distance to the cluster with the large variance is shorter than for the cluster with the small variance. In this thesis, it is experimented with both distance measures.

Chapter 7

Evaluation

In the previous chapters, the speech segmentation system and the approach for phone classification were described. They are used to perform various experiments on the ADABA speech corpus to evaluate their usefulness. The preparation of the evaluation, the experiments and their results are described in this chapter.

7.1 Setup

The experiments make use of the ADABA corpus described in chapter 4. This database is useful for the scope of this thesis due to its large isolated word corpus of three regional varieties of German and its included phonetic transcription. A drawback of the ADABA corpus is that only a transcription, but no manual segmentation is available, i.e. the exact alignment of the phone sequence with the signal is unknown. For evaluation purposes, however, a manual segmentation of at least some entries is needed. Further it is necessary to quantify the deviation from the reference segmentation. The manual reference segmentation and remarks on evaluation measures is described in this section.

7.1.1 Reference data preparation

Thus, to create an evaluation set for experiments, 25 words of each speaker were selected. The total of 150 words were re-transcribed and segmented manually by the author. Manual segmentation is a laborious process as many boundaries need to be moved by hand and the corresponding segments need to be listened to repeatedly. Further, the main criteria to set the phone boundaries for the individual segments is the subjective impression of the listener. Therefore and if no systematic process is followed, a manual segmentation tends to be more inconsistent than an automatic one. To ensure a higher level of consistency, guidelines for the manual segmentation process are established. These are defined as follows:

- General rules
  - For periodic segments, the period boundaries are defined to be located at the local negative maximum of the repetitive signal part.

The boundaries of voiced segments should be aligned with the period boundaries.

- Unvoiced plosives
  - They start after exceeding a certain silence floor.
  - They end at the period boundary before the first period of the next phone, if applicable.
  - Preplosive pauses are removed if they occur after silences.
- Voiced plosives
  - They start at the first period boundary of the plosive.
  - They end at the period boundary before the first period of the next phone, if applicable.
  - Preplosive pauses: They cover all breathy periods before the first period of the plosive.
- Voiced/Voiced boundaries
  - They are always set at a period boundary.
  - They are always set close to the center of the spectral transition region between the two voiced phones.
- Voiced/Unvoiced boundaries
  - They are always set at a period boundary.
  - They are set close to the center of the spectral transition region between the two phones.
  - If some periods at the boarder of the periodic part of the signal are deformed, this periods are part of the unvoiced phone.
  - If there is a mixture region, the boundary is set in the middle of it.
- Unvoiced/Unvoiced boundaries
  - The boundaries are set in the middle of the spectral transition region or at the spectral transition point.
- Voiced/Silence boundaries
  - They are set in the middle of the breathy region.
  - They are set at the last period boundary.
- Unvoiced/Silence boundaries
  - They are set at the transition point to the silence floor.
  - They are set at the point where perception is not possible anymore.
- Silence/Silence boundaries
  - Such boundaries are not allowed. Consecutive silence segments are merged to a single one: Pre-plosive pauses are merged into neighboring silence segments.

In addition to the segment boundaries, the phone labels are changed too during the manual segmentation. A close manual phonetic transcription is performed, i.e. the actual realization of each phone (its surface form) is transcribed. Especially for vowels and when using a large phone set, consistency is not easy to achieve among multiple iterations of manual labelings. Repeated cross-comparisons of segments need to be done. To facilitate this task, a set of reference phones for several vowels for each speaker is defined. They are used for the perceptive comparisons during the manual transcription. The following procedure is used to find these reference phones:

1. All phones in the given phonetic transcription that have no corresponding speaker-independent model in the segmentation framework are mapped to similar phones.

2. An initial segmentation for the corpus using the speaker-independent models and the available transcription is performed. For the purpose of obtaining reference vowels, the ADABA transcription is used.

3. For each reference vowel, the five phones with the highest loglikelihood after the forced alignment are identified. This is independently done for each speaker.

4. Different people listen to all the identified examples and each of them judges them with a subjective score (0, 1 or 2, where 2 is the best). For this thesis, the adviser and the author perform the scoring.

5. The phones with the highest overall score are defined as reference phones for the respective vowel.

The reference vowels are identified by this procedure. It is applied for the cardinal vowels ([i], [e], [ɛ], [a], [ɑ],[ɔ], [o], [u]), the rounded front vowels and the schwa. It is expected that other vowels can be interpolated from these vowels.

With the resulting reference phones, the manual relabeling is performed by the author by listening to the segments and comparing it with the reference phones. For this process, the segmentation tool described in section 5.4 is used.

7.1.2 Evaluation measurement

The automatic segmentation system is evaluated by comparing its results to the manual labelings. In automatic segmentation, manual segmentations often serve as a gold standard. Nevertheless, different human labelers produce different segmentation boundaries and different labels. As a consequence, usually the intra-human difference is considered as an upper boundary that an automatic system can achieve. For example, in [Hos09], the intra-human boundary agreement was reported to lie between 93.5% and 96% within 20ms, depending on the conventions used for the labeling.

The criteria used for the evaluation of the automatically determined segment boundaries is the percentage of agreement with the manual segment boundaries on the test set. The agreement is reported with respect to a certain interval. Frequently, a 20ms interval is used, as done in various studies mentioned in chapter 3. In this thesis, several intervals, among them 20ms, will be used to report agreement with the manual alignment.

7.1.3 Phonetic distance measurement

Just like the segment boundaries, the phonetic labels need to be evaluated with respect to the manually labeled test set. Therefore a distance measure between phonetic transcriptions is needed. The Levenshtein distance is a string-edit distance which is based on a dynamic programming algorithm. It allows for the efficient computation of the best overall alignment between two strings by using insertion-, deletion- and substitution-costs. It does not only provide the alignment, but can also be used to measure the distance between two strings. Thus it is also applicable to compare two phonetic transcriptions.

In phone and vowel classification experiments, a relabeling of the phones in the transcription is done, without changing the boundaries. Therefore only substitutions of symbols are expected for such an experiment. In this case the number of substitutions is equivalent to the Levenshtein distance.

The number of substitutions between the automatic and the manual phonetic transcription, however, may not be the best choice for a distance measure in this case. Consider the example where two vowel classifiers transcribe a vowel that has the correct transcription [e]. The first classifier transcribes it as [ɛ] and the second one transcribes it as [u]. The number of substitutions increases by 1 in either case even though one would consider the second mismatch as more severe. Therefore a better distance measure that considers the phonetic similarity of the substitution is desired. Phonetic similarity can easily be observed in the articulatory feature space. A *phonetic distance* therein provides a more appropriate distance measure than the number of substitutions.

The phonetic distance measure used is simply the euclidean distance in the articulatory vowel feature space with the dimensions backness, openness and roundness. The three feature values are defined over a range from -1 to 1.

7.2 Experiments

Several experiments are performed and evaluated for this thesis:

- Automatic segmentation of the ADABA isolated word speech corpus for all six speakers
- Training and evaluating MLPs as articulatory feature estimators, especially for vowels
- Analysis of the vowel distributions of the ADABA using MLPs and formants
- Comparing the performance of MLPs and HMMs on a vowel classification task
- Analyzing the diphthong realizations of the six speakers
- Analysing the voicing of serveral consonants
- Integrating MLPs for articulatory feature estimation in the segmentation algorithm and evaluating the performance

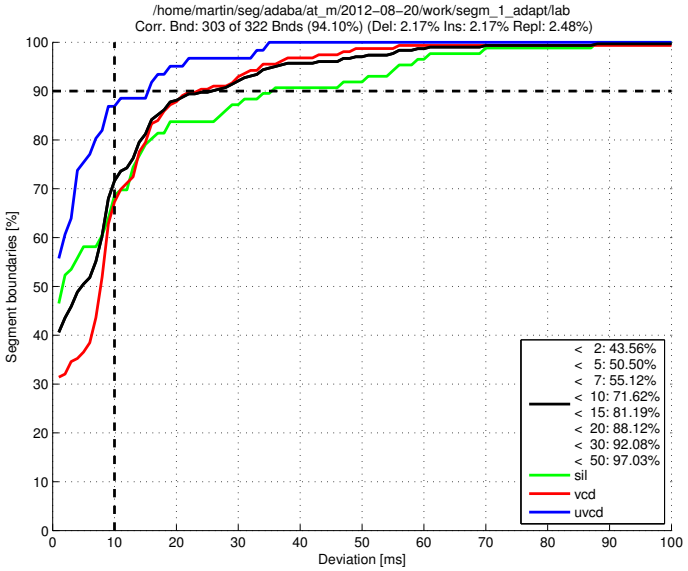


Figure 7.1: Segmentation results after the first iteration, compared with the manual segmentation.

7.2.1 Segmentation

One goal of this thesis is the automatic phonetic segmentation of the ADABA isolated word speech corpus. Furthermore, this segmentation serves as a basis for the phone classification experiments. The automatic segmentation is done with the tool and the framework described in chapter 5. Pronunciation rules and phone mappings are adapted to the ADABA. The segmentation is performed using the transcription provided with the ADABA as well as with the standard pronunciation. The standard pronunciation is obtained by using the *text analysis* module of the segmentation framework and is based on the transcription in the Aussprache-Duden [MI00]. Around 95% of the entries in the isolated word corpus of the ADABA can be transcribed with the text analysis module. The remaining words are ignored in all experiments. While the ADABA already provides a narrow phonetic transcription of the recordings, the Aussprache-Duden only offers the canonical phonetic transcription. As the ADABA is a corpus containing regional language variants, pronunciation variation with respect to the canonical transcription is expected. To account for the expected variation, more pronunciation rules were used when using the Duden transcription as a basis.

Pronunciation variation for regional language variants of German has already been investigated in [Bau03] with the goal of improving ASR for regional varieties of German. In this work, the Austrian and the German variety of German is analyzed on a large telephone corpus. The rules identified there are used in this thesis to create alternative pronunciation variants when performing segmentation of the ADABA corpus based on the Duden transcription.

Several iterations of the segmentation algorithm are applied to the isolated word corpus of the ADABA:

- the initial segmentation where the speaker-independent HMMs are used,
- a segmentation where the speaker-independent models are adapted to the current speaker and
- several additional segmentation iterations, where the models for the segmentation are trained from scratch on the data of the previous iteration.

The segmentation is done for all six speakers separately. Evaluation is done with respect to the manually segmented test set. The segmentation based on the Duden transcription with pronunciation variants using the speaker-independent models adapted to the current speaker turns out to perform best. Figure 7.1 shows the results of the segmentation based on the Duden transcription after the iteration where the models are adapted to the current speaker. In this figure, the results for the speaker `AT_M` are shown. Agreement with the manual segment boundaries is reported on intervals of 2, 5, 7, 10, 15, 20, 30 and 50ms. A common reported result is the number of boundaries that are within 20ms of the manually derived segment boundaries. In this case, 88.12% of the utterances fulfill this condition. Boundaries where the manual phonetic transcription differs from the automatic one are ignored in the comparison. For speaker `AT_M`, 94.1% of all boundaries could be used for the comparison.

7.2.2 Training MLPs for articulatory feature extraction

The resulting segmentation is used as a basis for the training of different MLPs. Feature vectors that correspond to vowel segments serve as inputs for the MLP training. The target values that are presented to the MLP outputs during training are derived from the phonetic labels. That means that for all feature vectors that correspond to a phone segment, the target values are the nominal values that this phone has in the articulatory feature space.

A continuous, normalized features space is defined over the articulatory features *tongue backness*, *openness*, and *roundness*. Figure 7.2 shows the reference values of the first two features for the cardinal vowels [i], [e], [ɛ], [a], [ɑ],[ɔ], [o] and [u] in the IPA vowel quadrilateral. The values for each of the three features range from -1 to $+1$, where -1 specifies the minimum and $+1$ the maximum value of the feature. Thus every feature frame of an /a/ has the nominal values $[0, 1, -1]$ (50% backness, 100% openness and 0% roundness) in this feature space. The continuous feature space approach is chosen due to the continuous nature of the anatomically possible tongue positions. If a rounded and an unrounded version of a vowel position in the IPA chart exists, the rounded one is defined as 1 (100% rounded) and the unrounded one as -1 (0% rounded). If only one version of a vowel exists, the roundness target value is set to 0 (50% rounded). The target values for all vowel frames used during training are determined by their position in the IPA vowel chart and by their roundness property.

The MLP training requires an existing phonetic segmentation to infer the target values of the articulatory features. To account for minor errors of this segmentation, only feature vectors at the core 30% of the vowels are used for the training. In addition to reducing the influence of segmentation errors, this has the advantage that transition regions to the previous and subsequent phone are not used. In this transition regions, the articulatory feature values are expected to be unstationary and quite far away from the nominal target values.

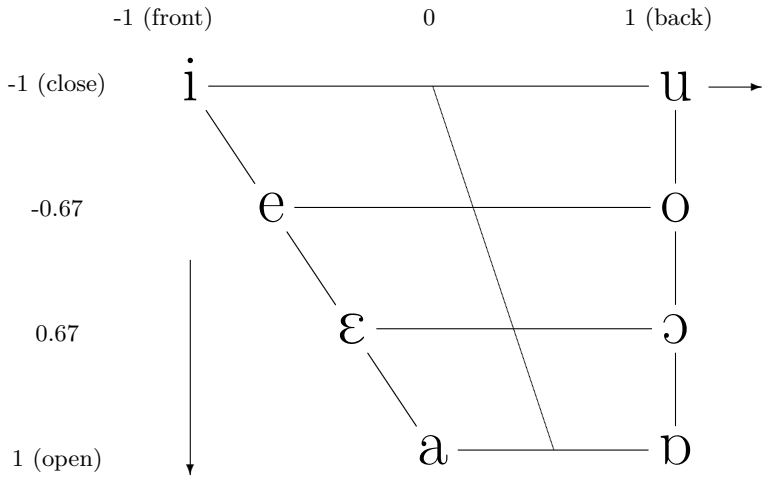


Figure 7.2: Coordinate system of the tongue-position feature space. The x-axis represents the backness, the y-axis the openness. Both features (and the roundness feature that is not shown here) range from -1 to $+1$. For example, the vowel [a] is represented as $(0, 1, -1)$ in this feature space.

When using an MLP for articulatory feature estimation, many degrees of freedom exist. Preliminary experiments are performed for

- a different number of hidden units
- a different number of hidden layers
- different types of input features
- a different number of context frames
- normalized and unnormalized input features

Evaluation of the informal experiments was done using the prediction error of the different models on an evaluation set. No significant differences were observed whether MFCCs or PLPs were used as input features. Experiments were performed with frame shifts shorter than 10ms but this just generated more similar training data and did not improve the results. Analysis window lengths of 25 and 33 ms turned out to perform best. A combination of features with short and long analysis windows did not further improve the results. We also incorporated context frames (the frames immediately before and after the analyzed frame). The optimal number of context frames was determined to be 4 for the left and the right context respectively. We did not observe differences dependent on whether we used normalized or unnormalized input features.

Another architectural choice is number of MLPs to use for articulatory feature estimation. One possibility is to use a seperate MLP for each articulatory feature, another one is to have only one MLP with as many output nodes as articulatory features exist (3 for vowels). Using a seperate MLP per articulatory feature assumes independence between these features. Seperate MLPs also have lower complexity than one MLP that estimates different features at the same time.

For this thesis, the approach with a single MLP is chosen for the articulatory features of vowels as the features for tongue position and mouth shape used are not independent of each other. Furthermore, there are only three articulatory features that are estimated and thus the complexity of the MLP is expected to be manageable. For consonants, seperate MLPs for the place and manner features are used.

The features finally used are 13 MFCCs with delta and acceleration coefficients. The 0th cepstral coefficient is included and cepstral mean normalization is performed. All features are extracted at a frame shift of 10ms and with a 33ms analysis window. Two hidden layers are used with 67 and 100 nodes, respectively. The input features of the training set are normalized to zero mean and unity variance. The input features of the evaluation set, the test set and all future inputs are also normalized to the mean and variance of the training set. Four left and four right context frames are used for each feature vector, which leads to a total of 351 inputs to the MLP.

With the vowel set from the Duden transcription and the utterances limited to the ones where a Duden transcription was available, a total of around 2.7 million feature vector frames are available after a segmentation. 600000 feature vectors are used for training the multi-speaker MLP and 100000 for the speaker-dependent MLPs. 50% of these feature vectors are used as training set, 25% as evaluation set and 25% as test set. Using more than 600000 is hardly possible

due to memory limitations when using a 39 dimensional feature vector with 4 left- and 4 right-context frames. Thus only a subset of the available data is used. The selection of this subset is done by randomly shuffling a list of all available utterances. Then this list is traversed in sequence and the feature vectors of all occurring vowels are selected until the desired number of feature vectors is available. The resulting set is then partitioned into training set, evaluation set and test set. Due to the fact that some vowels are represented in the corpus more often than others, the amount of how often certain vowels appear in the training set is limited to achieve a more balanced training data set.

Six MLPs are trained, one for each speaker. An additional MLP is trained with data from all six speakers. The MLP processing is done with a neural network processing toolkit from the SYNVO company. For the hidden units, a hyperbolic tangent activation function is used. The output units have a linear activation function. Scaled conjugate gradient is used as optimization algorithm for the MLP parameters.

7.2.3 Formant extraction

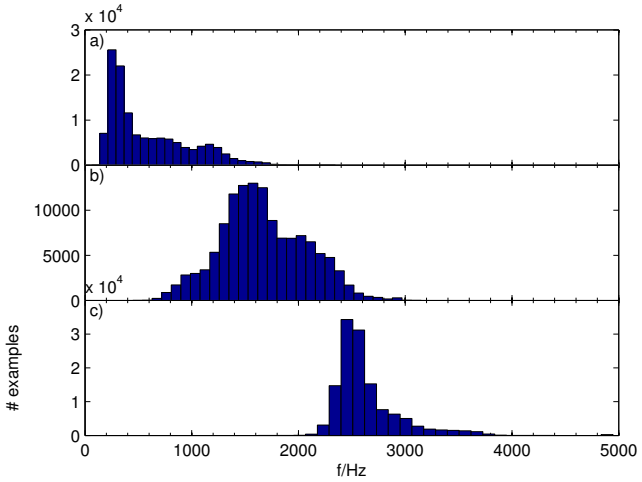
Formants for all the vowels in the corpus are extracted using the program Praat [Boe01]. The Burg algorithm (see e.g. [PTVF07]) is used to compute the LPC coefficients. The recommended settings from the manual were chosen: a maximum of 5 formants and a maximum formant frequency of 5000 Hz for the male and 5500 Hz for the female speaker. Formants are extracted every 2 ms using a window length of 25 ms. Pre-emphasis is applied above 50 Hz with 6 dB/octave. The analysis is automated via a Praat script and performed for all six speakers.

The figures 7.3(a) and 7.3(b) show the distribution of the first three formants for the Austrian male and female speaker respectively. Considering the speaker `AT_M`, one can see that the first formant has one narrow peek around 300 Hz. This is because all the vowels with the exception of /a/ have their first formant around this value. The variation for the second formant is larger. It distinguishes between front-, mid- and back-vowels.

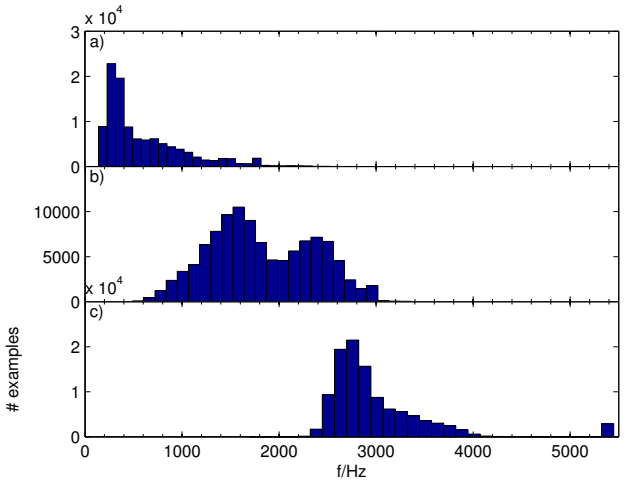
7.2.4 Vowel cluster analysis of the ADABA corpus

To get an overview of the distribution of the vowels in the formant and in the articulatory feature space, clusters of the different vowels are plotted. In these plots, each cluster is visualized by an ellipse. The center of the ellipse represents the mean value μ of the respective vowel and the radius of the half axes is a multiple of the standard deviation s in the direction of the first two principal components of all the vowel's data points in the feature or formant space.

Figure 7.4 is showing such a vowel cluster plot for the formants of speaker `AT_M`. In this plot, the radius of the half axes represents the (single) standard deviation (s) in the direction of the first two principal components. The similarity of the position of the mean vowel formants in the acoustic space to the IPA vowel quadrilateral in figure 6.1 can clearly be observed. Nevertheless the relative distances in the plot can not be compared with those in the vowel quadrilateral. For example, the relative gap between [aː] and the other vowels is significantly greater than the other distances, which is not the case in the IPA vowel quadrilateral. Further, the variance of the individual vowel clusters



(a) Austrian male speaker



(b) Austrian female speaker

Figure 7.3: Distribution of the first three formants (from top to bottom) for the Austrian male and female speaker.

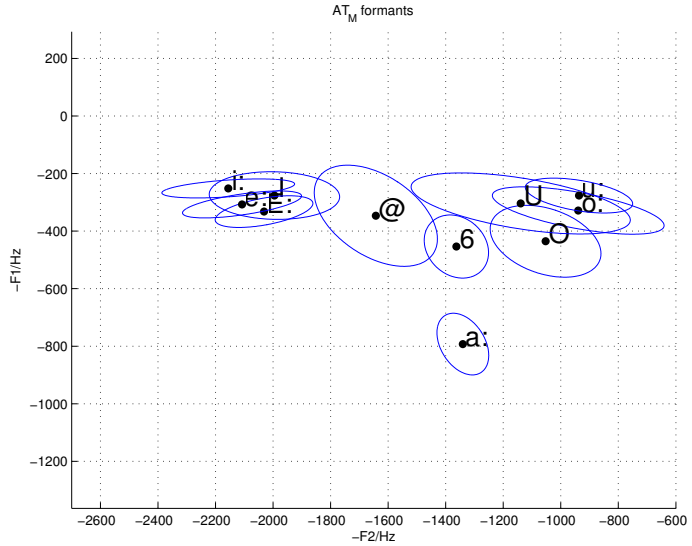


Figure 7.4: Formant clusters for the speaker `AT_M`. The ellipses show the area that is in $\mu + 1s$. Vowel labels are in SYNVOPA notation.

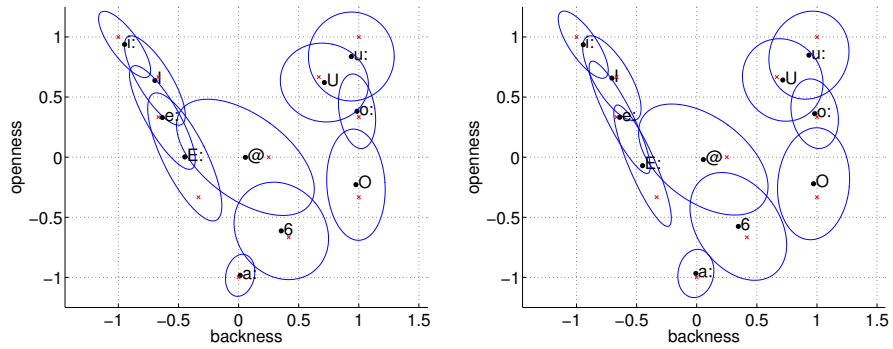
is quite high which leads to substantial overlaps between the individual clusters (assuming the normal distribution, only 68% of the feature points lie within one standard deviation from the mean).

When vowel clusters are plotted for the articulatory features estimated by an MLP, however, the result is different. Figure 7.5 shows the vowel cluster plot with articulatory features estimated by a MLP for all six speakers of the ADABA corpus. The result clearly differs from the formant cluster plot in figure 7.4. Now the radius of the half axes represents **two times** the standard deviation (s) in the direction of the first two principal components. The vowels /aː/, /ɪ/, /iː/, /oː/, /uː/, /ə/, /ɛː/, /eː/, /ɔ/, /ʊ/, /ɐ/ are shown, i.e. the long vowels and vowels that are not distinguished by the length marker ː (/ɔ/, /ʊ/, /ɐ/ and /ə/). The filled dots next to the vowel labels represent the predicted vowel cluster means. The crosses represent the nominal vowel target positions derived from the IPA vowel quadrilateral which were presented as target values during training. The cluster plots only show the first two dimensions, the third one is omitted to be compatible with the IPA vowel quadrilateral.

Considering that the half axes of the ellipses equal $2 * s$ for the articulatory feature clusters whereas in the formant case they equal $1 * s$, it is obvious that the separability of the vowels is much better when using the articulatory features. They are thus better suited as features for classification than the formants.

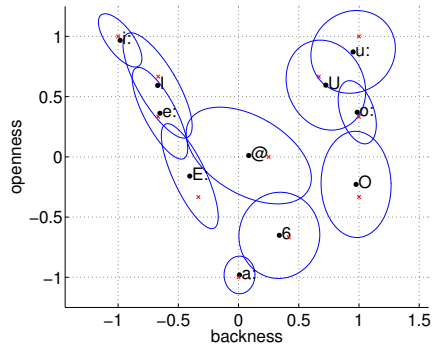
As an example, consider figure 7.5(a), where the vowel clusters for the speaker `AT_M` are displayed. In many regions, the vowels can be distinguished easily. For some clusters, however, significant overlaps exist. For example, there is a large overlap between the [eː] and the [ɛː] cluster for this speaker. The cluster for the [ə] is the largest one and thus indicates that the schwa-phone shows most variation, which is no surprise, as it is a reduced vowel and only occurs in unstressed positions.

In figure 7.5(c) the vowel clusters for the speaker `DE_M` are shown. One

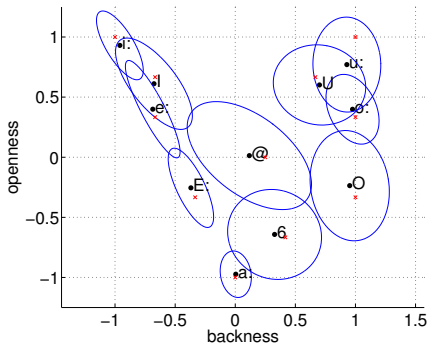


(a) Speaker AT_M

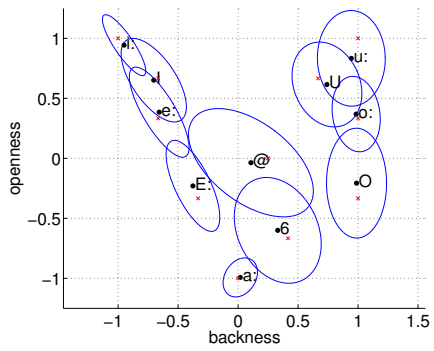
(b) Speaker AT_W



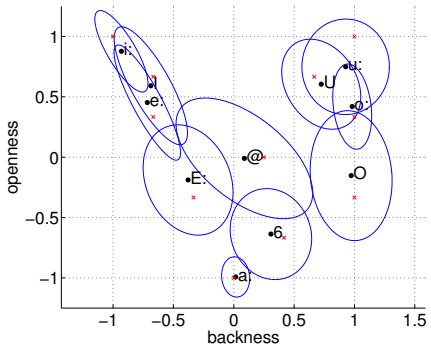
(c) Speaker DE_M



(d) Speaker DE_W



(e) Speaker CH_M



(f) Speaker CH_W

Figure 7.5: Vowel clusters for the six speakers. The ellipses show the area that is in $\mu + 2s$. Vowel labels are in SYNVOPA notation.

can observe some differences to the speaker AT_M, like a slightly larger overlap between /eː/ and /ɪ/. The most significant difference, however, is that the overlap between /eː/ and /ɛː/ is smaller than the respective one for the speaker AT_M.

When comparing the vowel clusters for the two female speakers, one can observe the same phenomenon (figure 7.5(b) for speaker AT_W and figure 7.5(d) for speaker DE_W). For speaker DE_W the overlap between [eː] and [ɛː] is smaller than for speaker AT_W. The two speakers from Switzerland neither show this overlap (figure 7.5(e) and 7.5(f)).

This leads to the idea that the two speakers of the Austrian standard variety in general have a higher confusion between the two phones [eː] and [ɛː] than the two speakers of the German and the Swiss standard variety. The Aussprache-Duden [MI00] also states that the phone [ɛː] can sometimes be pronounced as [eː]. In the current analysis, this can apparently be observed more often for the two speakers of Austrian German.

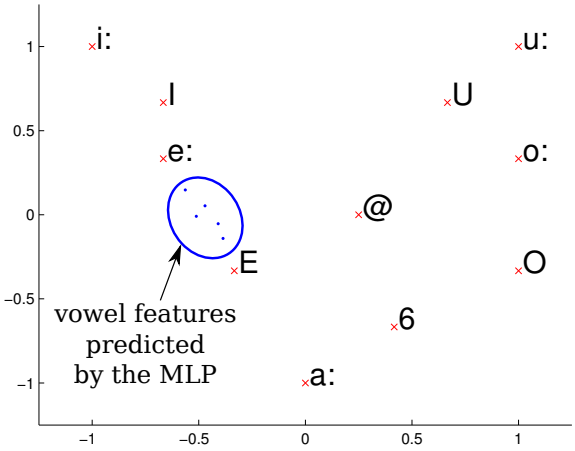
Of course it is not possible to generalize from these four speakers to a difference that is characteristic for the respective standard variety. Nevertheless this observation leads to a hypothesis that can be evaluated in a broader study using the same methodology. It must be noted that the labels [eː] and [ɛː] correspond to the Duden standard transcription that is the same for all six speakers. This transcription is chosen as it is desired to compare the individual speakers and the individual regional variants with respect to the same basis, which is the canonical phonetic transcription. Thus it will also be possible to verify certain claims of the ÖAWB and the ADABA corpus.

The knowledge of the high degree of overlapping between the two phones [eː] and [ɛː] can also be used to improve results for these particular speakers in a speech segmentation and phone classification task. This phenomenon was observed only for the speakers AT_M and AT_W.

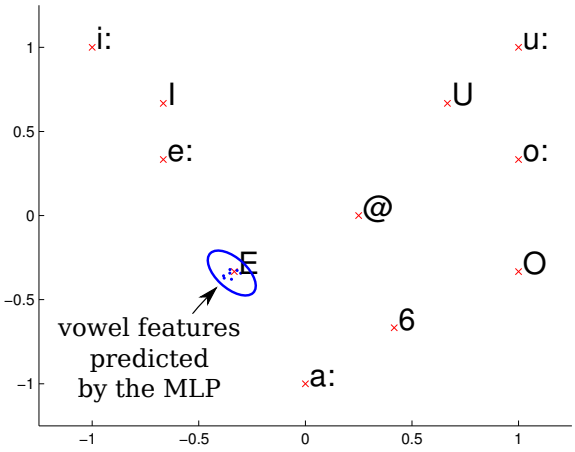
An concrete example for a vowel that is transcribed /ɛː/ according to the Aussprache-Duden is given in figures 7.6(a) and 7.6(b). It can clearly be seen that, when pronouncing the word *Dänen* (the Danish), the speaker AT_M realizes the vowel somewhere between the nominal /ɛː/ and the nominal /eː/ whereas the speaker DE_M realizes it close to the nominal position.

The observed overlap supports the idea of adding an additional vowel between /ɛː/ and /eː/ as it was done in the ADABA transcription. In the corresponding dictionary ÖAWB it is argued that the phones /ɛ/, /œ/ and /ɔ/ are generally realized more open in the Austrian standard variety. The analysis with the MLP does not support this claim for /œ/ and /ɔ/ but does support the ÖAWB for the mid-open front vowel /ɛ/. In the ADABA corpus, however, all vowels that have the transcription /ɛː/ or /ɛ/ in the Aussprache-Duden, are transcribed as this mid-open vowel [e̞] for the Austrian speakers. Informal experiments with the MLP showed that segments, where a vowel [e̞] would make sense according to the MLP were especially the vowels labeled as long mid-open front vowel (/ɛː/) in the Aussprache-Duden.

Another interesting observation can be made for the speaker CH_W. The MLP analysis for the swiss female speaker shows an overlap between the vowels /eː/ and /ɪ/ that is significantly larger than for the other speakers. Informal listening experiments confirmed that many /eː/ vowels spoken by speaker CH_W are indeed perceived more closed or even identified as /ɪ/. For speaker CH_W, one may further notice that there is greater confusability between the vowels /ə/ and



(a) /ɛː/ for speaker AT_M



(b) /ɛː/ for speaker DE_M

Figure 7.6: Realization of the vowel /ɛː/ in the word *Dänen* (the Danish). Vowel labels are in SYNVOPA notation.

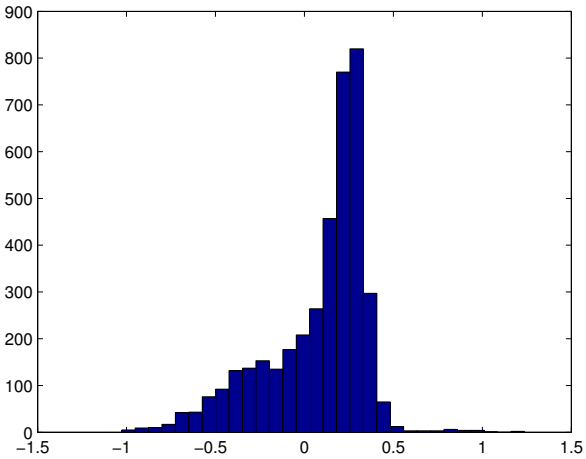


Figure 7.7: Histogram of the feature *backness* for the vowel /ə/ produced by the speaker `CH_W`.

/ɛː/ due to a larger variance of both vowels. Figure 7.7 shows the distribution of the feature *backness* for the vowel /ə/ and the speaker `CH_W` visualized by a histogram with 30 classes. The distribution is asymmetric and has a negative skewness. There is a group of schwa-Vowels that tend to be located more in the front section than the peak of the distribution. Some schwa-Vowels are realized closer to the front vowels /eː/ and /ɛː/.

The vowel clusters plotted in figure 7.5(f) show only the mean and the variance of the data but not their actual distribution. If the distribution is not Gaussian, the diagram may be misleading. Many of the mean cluster values in the plots have an offset to their nominal position (indicated by the crosses) that was used as target during the MLP training. When looking at figure 7.7, however, one notices that the peak of the distribution is clearly around the nominal tongue backness value of the schwa-vowel (0.25) while the mean is located closer to the front vowels. An interpretation is that many schwa-vowels are articulated very clearly and close to the nominal position but others are actually realized as front vowels and just mis-labeled as schwa (because this is their canonical transcription). The mean of the clearly articulated schwa-vowels is right at the nominal schwa-position. Due to this non-Gaussian distributions a classifier that uses the means and variances of the whole vowel cluster will not perform best. Instead it is a better idea to use the nominal target value of the cluster means for classification.

MLP trained on all six speakers

In the previous experiment, six different MLPs were trained, each one on the data of one speaker only. With this method, the vowels space of each speaker could be analyzed separately, and some interesting phenomena could be identified, like the overlap of /ɛː/ and /eː/ for two speakers. The question arises whether this method allows for the comparison of one speaker with another as

different MLPs are used for their analysis. The hope was that such a comparison is possible as the target value space is the same during training and as phonetically similar data is used (the isolated word list from the ADABA corpus). Nevertheless, to obtain more evidence, a new MLP is trained on data of all six speakers with the same target value space. This MLP is then applied to the speakers individually. The MLP has the same topology (two hidden layers, a hyperbolic tangent hidden activation function and a linear output function), but more data is used: For each speaker 100000 feature frames as for the individual MLPs, thus resulting in a total of 600000 feature frames. Like in the previous experiment, half of the feature frames is used as training set, 25% is used as evaluation set and 25% are reserved as test set. Figure 7.8 shows the resulting vowel cluster plot for all six speakers.

It can be seen that the distributions of the vowels are very similar and that the described phenomena are basically the same. The overlap between /ɛː/ and /eː/ is significantly larger for speaker `AT_M` and `AT_W` than for the other speakers. The partial congruence between /ɪ/ and /eː/ of speaker `CH_W` can still be observed.

Combining MFCCs and formants

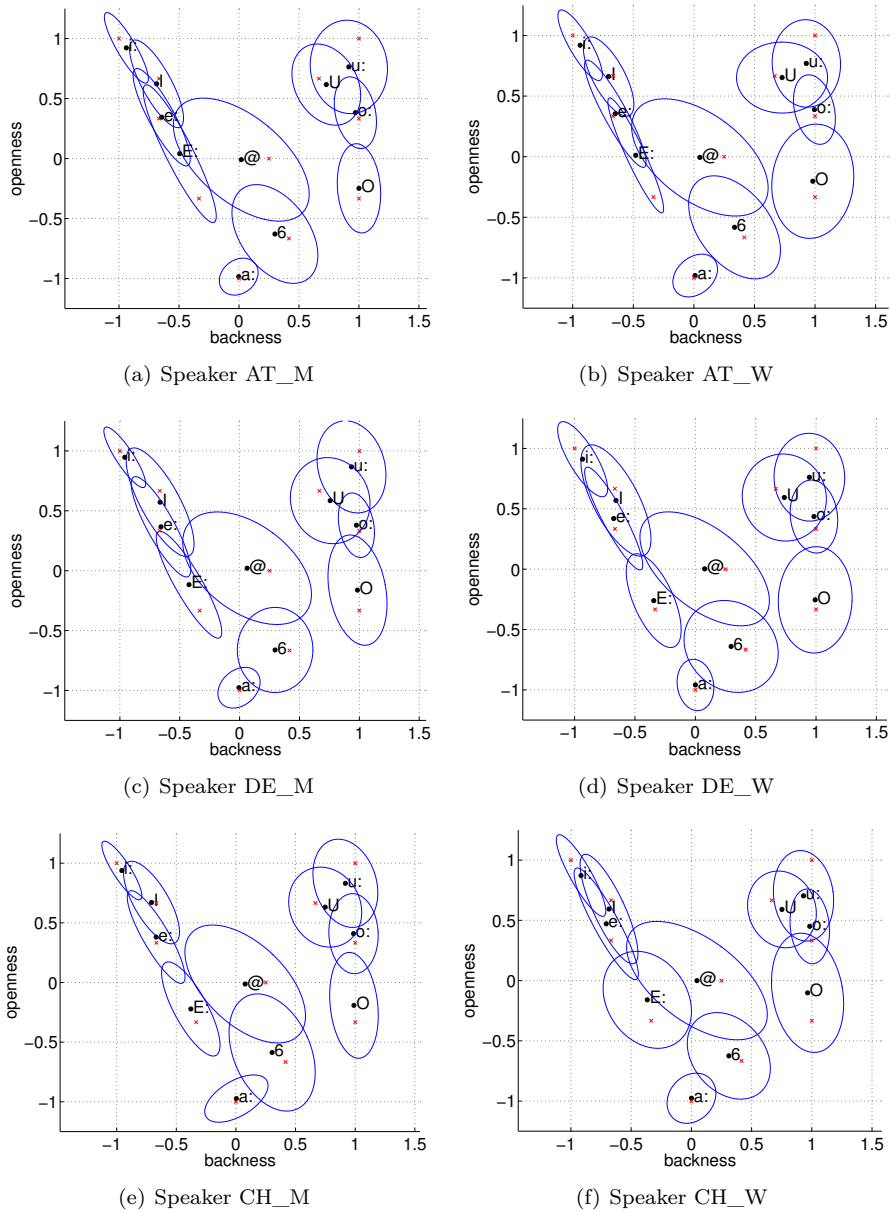
While the MLP-approach with MFCC features showed better separability of the vowel clusters than an approach based on the three formants only, the clear relationship of average formant positions and the vowel quadrilateral is obvious. Although they show higher variability as well as they suffer from some misdetections, they are, in general, a good hint for a vowel's position based on a few dimensions only.

This leads to the idea of adding the automatically extracted formants as inputs for the MLP. Thus a new MLP is trained that has as inputs the MFCC features and the first five formants. The formants are automatically extracted with a Praat script as mentioned in section 7.2.3. No manual corrections on the extracted formants are performed. Again, 4 left and 4 right context features are applied and all features of the training set are normalized to zero mean and unity variance. The evaluation set and the test set is also normalized based on the mean and variance of the training set.

The result is shown in figure 7.9 for the speaker `AT_M`. The figure shows the predictions on the test set. It can be seen that the incorporation of the formants leads to a (small) improvement of the separability when compared to figure 7.5(a). The clusters overlap a little less, except for the described phenomenon for /ɛː/ and /eː/, which can still be observed. Therefore, a combination of MFCCs and formants is recommended for usage in vowel classifier systems where this is applicable.

Vowel clusters with Autoassociative ANNs

Vowel cluster analysis is also performed based on autoassociative neural networks. The number of hidden layers is three. The outputs of the units of the middle layer of the AANN are used as reduced feature space. Several different AANN architectures are trained. We varied the number of hidden units in the middle layer (3 and 5), the number of hidden units in the first and third hidden layer (50, 150 and 350) and experimented with two different types of acoustic



(a) Speaker AT_M

(b) Speaker AT_W

(c) Speaker DE_M

(d) Speaker DE_W

(e) Speaker CH_M

(f) Speaker CH_W

Figure 7.8: Vowel clusters for the six speakers evaluated with the MLP trained on data of all speakers. The ellipses show the area that is in $\mu + 2s$. Vowel labels are in SYNVOPA notation.

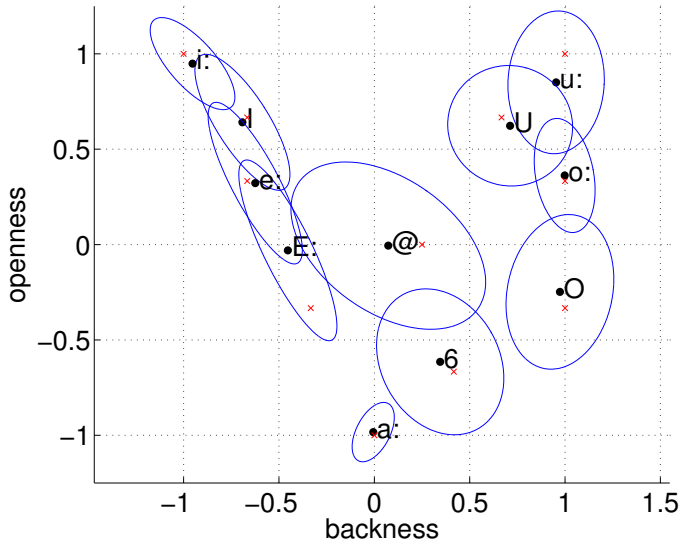


Figure 7.9: Vowel clusters for the speaker `AT_M` when combining MFCCs and formants. The ellipses show the area that is in $\mu + 2s$. Vowel labels are in SYNVOPA notation.

features (MFCCs and PLPs). The outer two hidden layers are chosen to have a hyperbolic tangent hidden activation function and the middle layer is given a linear activation function. The middle hidden layer activation function is chosen by keeping in mind that the outputs of this layer, even if they are not the final outputs of the network, represent the outputs of the non-linear principal component analysis. As a comparison with the conventional ANN is intended, using the same output function is a good choice. The chosen ANN had MFCCs as input features, one left and one right context frame, 3 hidden units in the middle hidden layer and 150 hidden units in each of the two outer hidden layers respectively.

In general, the AANNs need more time for training as their target values have much more dimensions. Without normalizing, training did not even converge. To cope with the larger increased complexity, it is further necessary to restrict the number of left- and right-context feature vectors to two instead of four. Further we tried to use no context frames at all for this kind of network, as the analyzed non-linear principal components in this structure try to reproduce the input features. Trying to predict the context frames from the reduced feature set in addition to the actually analyzed frame can be thought of being quite difficult.

After training, the input features are passed to the respective AANN and the outputs of the middle hidden layer are treated as outputs and analyzed. A correlation analysis with the outputs of the normal ANN is done. The Pearson product-moment correlation coefficient is calculated for the different target vectors of the two networks. We found correlations with an r-value greater than 0.75 between the *backness* and the *height* feature of the conventional MLP and two different principal component features extracted from the AANN (0.81 and 0.77) when using the inputs with one context frame left and right. Due to the

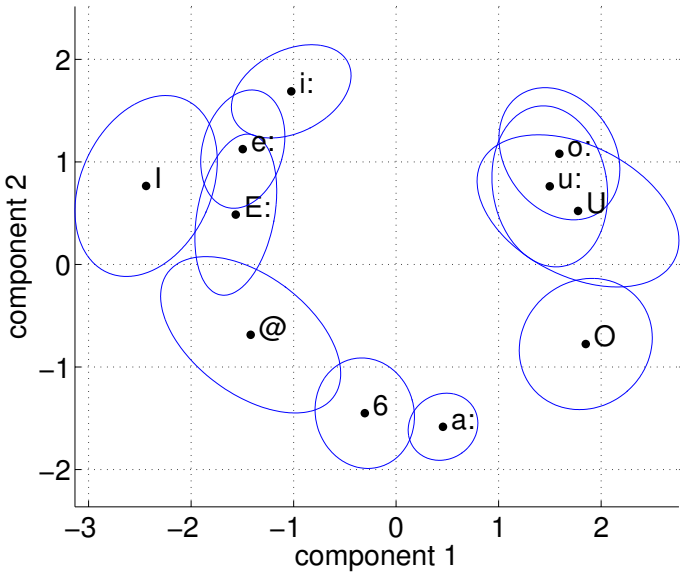


Figure 7.10: Two non-linear principal components as represented by an Autoassociative Neural Network for the speaker `AT_M`. The ellipses show the vowel cluster areas in $\mu + 1s$.

large sample size, these correlations are highly significant.

The two non-linear principal components with the highest correlation are plotted in figure 7.10. Although there are slight differences, the similarity of the vowel cluster centers to the formant space and to the articulatory feature space is obvious. Thus the AANN estimated the most representative components of the vowels to be similar to the articulatory features backness and openness. Furthermore, the AANN clusters show a higher overlap between the vowels [oː], [uː] and [ʊ] than between other clusters. During the manual transcriptions for the experiments, a higher confusion between these vowels was observed.

7.2.5 Vowel classification experiment

The performance of the vowel MLPs is evaluated in a phone classification task. Phone classification is performed on all vowels of the manually segmented and labeled testset for each speaker. The labels of the vowel segments are replaced with the classification result. Diphthongs are excluded due to their special characteristics. They can not be represented by a single nominal point in the articulatory feature space, but instead build a trajectory therein. No further restrictions on pronunciation variants are applied for this experiment. The classification of a vowel is done by averaging its articulatory feature outputs over time and determining the euclidean distance to all reference vowel coordinates in the target feature space. The vowel with the minimum euclidean distance to the average articulatory feature output is the classification result.

To evaluate the performance, a re-classification of the vowels is also done with HMMs. The HMMs used are speaker-independent and speaker-dependent ones. The speaker-independent models are the HMMs used in the first iteration

Speaker	PD_{HMMSI}	PD_{HMMSD}	PD_{MLP6}	PD_{MLP1}
AT_M	0.859	0.753	0.609	0.633
AT_W	0.779	0.748	0.709	0.737
DE_M	0.669	0.482	0.834	0.817
DE_W	0.912	0.801	0.746	0.757
CH_M	0.801	0.682	0.679	0.732
CH_W	0.707	0.699	0.747	0.739

Table 7.1: Vowel classification results: mean phonetic distances (PD) to the reference transcriptions (substituted vowels only) for classification with speaker-independent HMMs (HMMSI), speaker-dependent HMMs (HMMSD), the MLP trained on all six speakers (MLP6) and the MLP trained on the analyzed speaker only (MLP1).

Speaker	PD_{HMMSI}	PD_{HMMSD}	PD_{MLP6}	PD_{MLP1}
AT_M	0.511	0.386	0.297	0.339
AT_W	0.456	0.401	0.225	0.359
DE_M	0.456	0.259	0.229	0.323
DE_W	0.491	0.390	0.268	0.310
CH_M	0.549	0.383	0.236	0.296
CH_W	0.455	0.412	0.246	0.293

Table 7.2: Vowel classification results: mean phonetic distances (PD) to the reference transcriptions (all vowels) for classification with speaker-independent HMMs (HMMSI), speaker-dependent HMMs (HMMSD), the MLP trained on all six speakers (MLP6) and the MLP trained on the analyzed speaker only (MLP1).

in segmentation system from chapter 3. Thus they were trained on a large corpus containing many different speakers, however, all of them were Austrians. The speaker-dependent HMMs used are from the next iteration in the system from chapter 3. They thus result from the adaptation of the speaker-independent models to the data of the respective speaker. They are expected to show better performance, just like speaker-dependent models do in speech recognition. The results of the classifications are shown in tables 7.1 and 7.2.

Two numbers are reported for each speaker and each classifier. The first one is the mean phonetic distance of all substituted vowels to their respective nominal position. This measure does not consider the vowels that have the same label as in the manual reference transcription and thus reports the distance for vowels that are labeled incorrectly by the respective classifier. The second measure also considers the correctly classified vowels and reports the mean phonetic distance to the nominal position over all vowels. It may be a little surprising that the MLP trained on all speakers performs better on each speaker than the MLPs trained on the respective speaker. A reason for this may be the larger amount of training data that is used for `MLP6`.

It can be seen that the MLPs show considerable performance when compared tho the HMMs. While the speaker-dependent HMM in some cases shows better

performance when considering the substituted vowels only, the first MLP is superior when considering all classified vowels. For three speakers (`AT_M`, `DE_M` and `DE_W`), however, the difference between the models `HMMSI` and `MLP6` is too small to be statistically significant at a 95% confidence level (using Student's t-test). Nevertheless this experiment shows that the MLPs can not only be used for a vowel cluster analysis, but also in a phone classification task.

7.2.6 Integrating MLPs and forced alignment

Traditional forced alignment for speech recognition or speech segmentation is usually done using some framework (e.g. HTK, like in the framework used in this thesis) which performs a Viterbi-search over a lattice of states in phone models. Modeling is done by a HMM/GMM approach, where transition probabilities between states are constant and observation probabilities for a particular feature vector input is modeled by a GMM.

If one wants to improve accuracy by incorporating additional information from a different modeling approach, the options are limited when using an existing framework. It is generally not possible or easy to inject additional information to the forced alignment algorithm of HTK. The only options are to incorporate additional features in the models and thus also use them during the decoding, or to do make use of the additional information after the forced alignment. In this post-decoding step, the alignment is adjusted according to the additional information.

To achieve greater flexibility, the decoding algorithm of HTK is replaced with a custom version programmed in Java. The decoding algorithm is a token passing algorithm [YRT89] that is applied on the lattice of possible pronunciation variants. When calculating the observation probabilities, the GMM probability of the original algorithm is replaced with a combination of the probabilities estimated by the GMM and the MLPs. The combination is done using the law of total probability with the GMM observation probability and the outputs of four different MLPs. One MLP is used to estimate whether the current observation belongs to a vowel or a consonant, one is used to estimate the vowel articulatory features and two are used to estimate the consonant articulatory features place and manner. When the articulatory features are estimated via MLPs, a distance to the nominal articulatory feature value of the considered phone can be calculated. For the vowel MLP, however, this only makes sense when the considered phone is a vowel. The same applies to the consonant MLP.

$$p_{obs} = p_{GMM} \cdot \sum_{i=1}^{n} (P(ph \in C_i) \cdot p_{MLP_{Ci}}$$

C_i is the broad phonetic class, ph is the phone symbol, p_{GMM} is the GMM observation probability desnsity and $p_{MLP_{Ci}}$ is the observation probability density from the MLP of the respective broad phonetic class C_i. C_i can be *vowel*, *consonant* or *silence* in this experiment.

The MLPs for articulatory features have a linear output function estimating the position in an articulatory feature space. This position must be mapped to a probability density. The estimation is done using a simple exponential probability distribution:

$$P_{vowel} = e^{-d}$$

where d is the distance to the respective vowel class center. For the speaker `AT_M`, the integration described here leaded to an improvement of the segmentation accuracy to 88.7% within 20ms.

7.2.7 Diphthongs

For the vowel cluster analysis diphthongs are excluded due to their articulatory feature dynamics. They describe trajectories in the articulatory feature space that range from the starting to ending vowel of the diphthong. They can not be modeled by a single cluster that assumes a Gaussian distribution, as it is done for individual vowels. Therefore the diphthong articulatory features are modeled as mixture models in the articulatory feature space for the analysis. A GMM with 16 mixture components is used for each diphthong. The parameters of the model are estimated using the Expectation-Maximization (EM) algorithm [DLR77]. A maximum number of 100 iterations is performed.

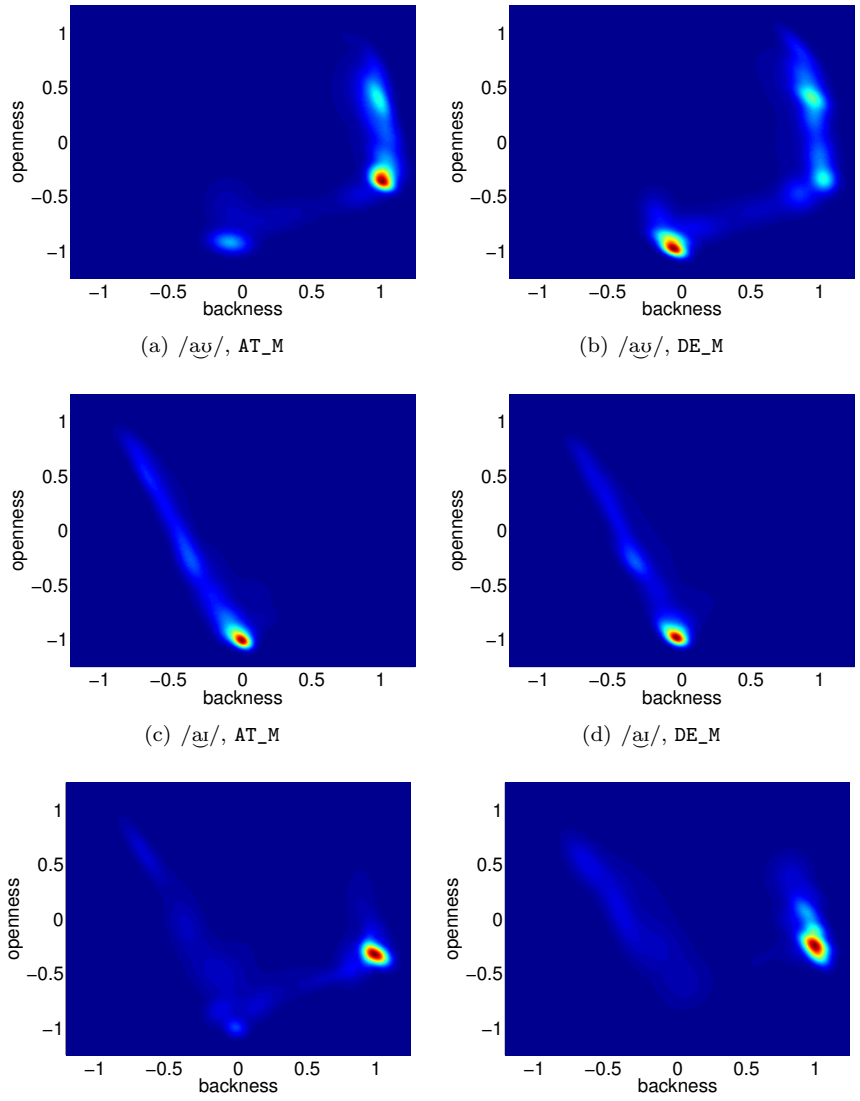
Figure 7.11 shows the resulting distributions of the three diphthongs /a͡ʊ/, /a͡ɪ/ and /ɔ͡y/ for the speakers `AT_M` and `DE_M`. An interpretation of the distributions is that their centers reveal the main articulation locations that are relevant for the diphthongs. It can be seen that these main articulation locations are usually constituted by two or three dominant clusters. Some of this regions have significantly higher density than others, meaning that they represent more time frames of the diphthongs. The transition regions between these clusters show a lower density. Figure 7.12 shows diphthong realizations by the female speakers from Austria and Germany. In figure 7.13, the articulatory feature distributions of the diphthongs realized by the Swiss speakers is plotted.

The figures show that no systematic differences between the starting and ending points of the diphthongs exist between the varieties. Some speakers, however, have the focus on different regions than others when articulating a diphthong (e.g. the speakers `DE_M` and `AT_W` focus more on the beginning when pronouncing the diphthong /a͡ʊ/).

7.2.8 Voiced consonants

During informal listening tests, some consonant realizations by the Austrian speakers were perceived as less voiced then the pronunciations of the other speakers. Therefore, an experiment is performed where the average voicing of several consonants is quantified using the period segmentation of the audio signals. The experiment is based on the automatic phonetic segmentation from section 7.2.1. The voiced consonants /z/, /ʒ/, /b/, /d/ and /g/ are analyzed. For each of the considered consonants, the corresponding voicing information is obtained from the period segmentation. Where a consonant has multiple types of speech in the period segmentation (*voiced*, *unvoiced*, *silence* or *irregular*), each speech type is counted with the respective fraction of occurrence.

Table 7.3 lists the results of the analysis. Only the *voiced* and *unvoiced* fractions are shown, the fractions labeled as *silence* or *irregular* are omitted. For the reported consonants, the results clearly show a trend to reduced voicing by the Austrian speakers in relation to the other speakers. The reduction is most significant for the consonants /z/ and /ʒ/.

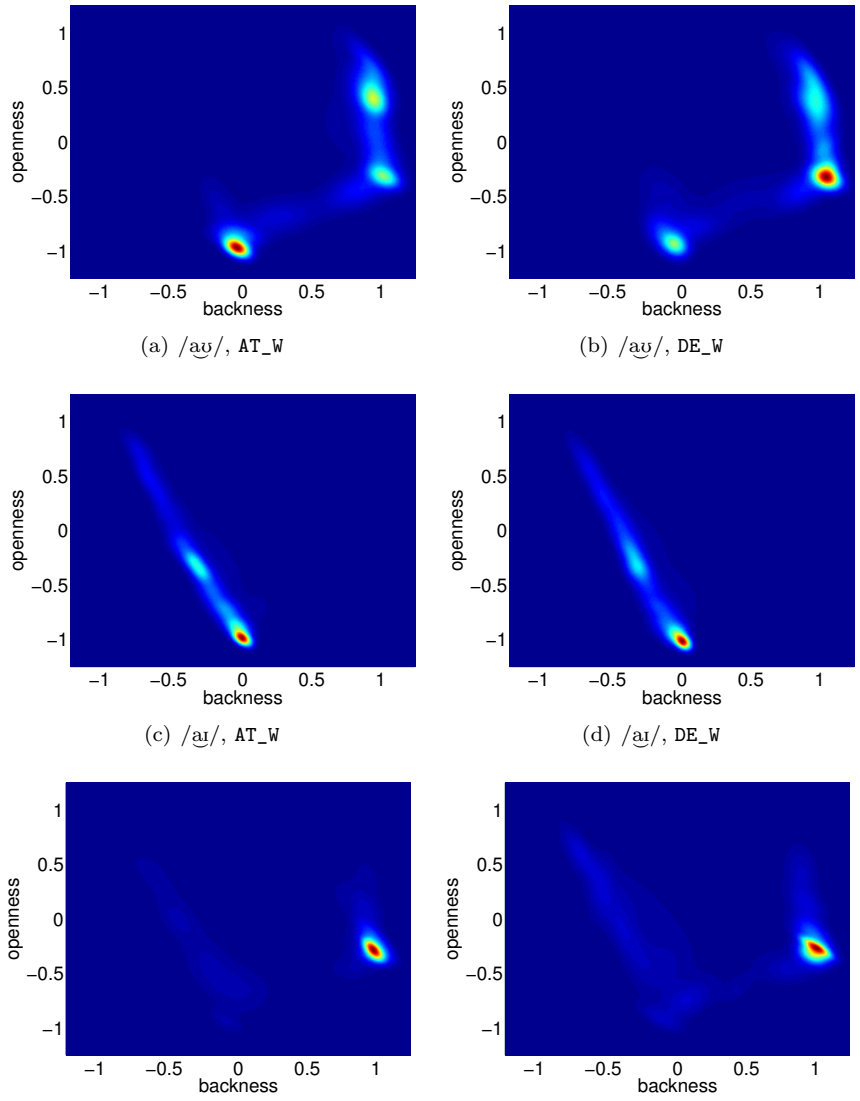


(a) /a͡ʊ/, AT_M (b) /a͡ʊ/, DE_M

(c) /a͡ɪ/, AT_M (d) /a͡ɪ/, DE_M

(e) /ɔ͡y/, AT_M (f) /ɔ͡y/, DE_M

Figure 7.11: Diphthongs of the Austrian (left) and the German (right) male speakers.

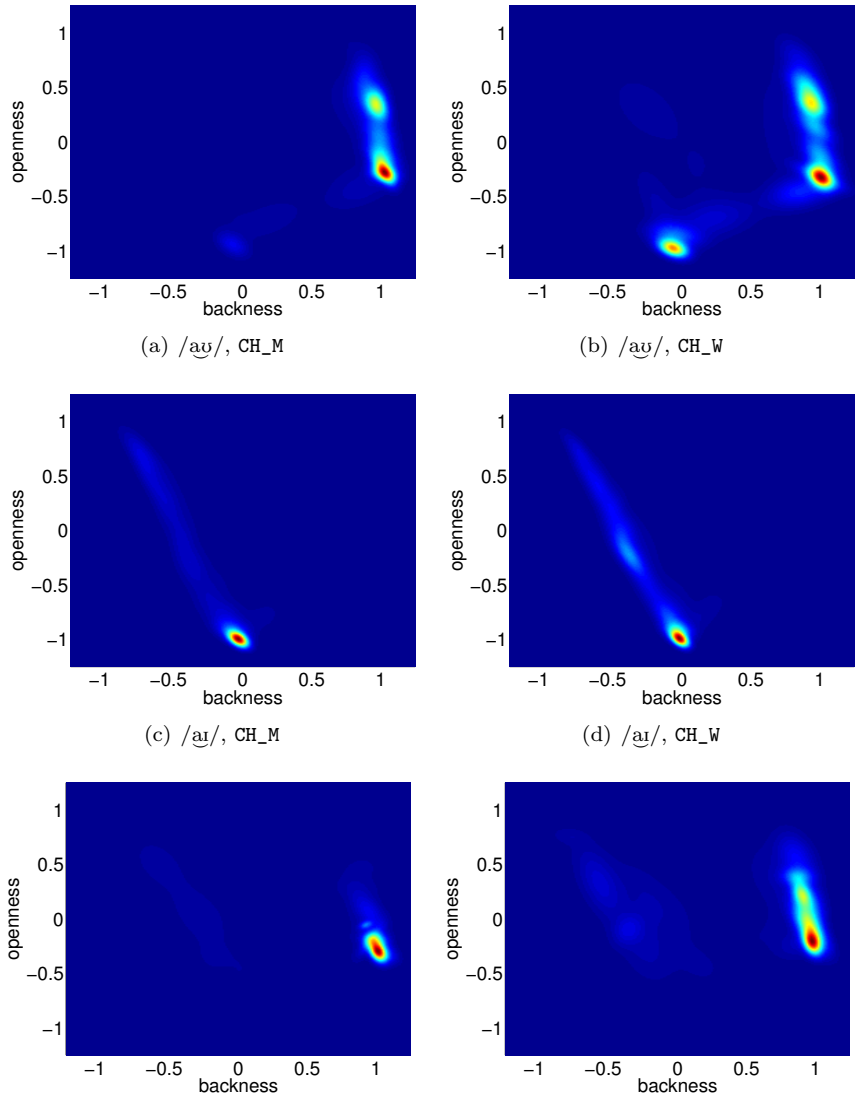


(a) /a͡ʊ/, AT_W (b) /a͡ʊ/, DE_W

(c) /a͡ɪ/, AT_W (d) /a͡ɪ/, DE_W

(e) /ɔ͡y/, AT_W (f) /ɔ͡y/, DE_W

Figure 7.12: Diphthongs of the Austrian (left) and the German (right) female speakers.



(a) /a͡ʊ/, CH_M (b) /a͡ʊ/, CH_W

(c) /a͡ɪ/, CH_M (d) /a͡ɪ/, CH_W

(e) /ɔ͡y/, CH_M (f) /ɔ͡y/, CH_W

Figure 7.13: Diphthongs of the Swiss male (left) and the Swiss female (right) speakers.

	/z/		/ʒ/		/b/		/d/		/g/	
	v %	u %	v %	u %	v %	u %	v %	u %	v %	u %
AT_M	4.9	93.9	19.1	77.9	5.3	49.7	10.5	57.7	6.1	69.9
AT_W	10.5	81.7	19.4	71.0	29.5	15.0	30.2	25.6	28.8	30.6
DE_M	45.9	44.8	57.5	31.9	52.7	9.6	51.9	11.8	47.2	22.5
DE_W	39.8	50.4	37.0	58.3	34.7	15.7	43.1	19.9	31.1	33.9
CH_M	29.3	65.1	34.9	54.9	54.6	10.0	58.6	14.2	56.7	18.1
CH_W	31.6	65.6	32.1	62.9	44.7	12.7	45.8	20.6	38.6	26.6

Table 7.3: Consonant voicing. The u-columns denote unvoiced, the v-columns voiced realizations in percent.

7.3 Discussion

According to the ÖAWB there are several pronunciation characteristics of the Austrian variety of German (see section 4.1). Some of these claims can be tested with the results of the experiments in this thesis.

Pronunciation of /e/ and /ɛ/: There is evidence from the experiments that the /e/–/ɛ/ pronunciation of the Austrian speakers differ from the other two varieties. The ÖAWB claims that the phonetic distance between these two vowels is smaller in the Austrian variety and thus introduces the vowel [e̞]. According to the vowel cluster analysis this is supported by the data.

Diphthongs: The ÖAWB and the Duden use different phonetic symbols for the German diphthongs. It remains unclear whether the different symbols are used to emphasize characteristics of the Austrian variety. In chapter D of the ÖAWB it is said so, in the phonetic transcriptions, however, the same diphthong symbols are used for all varieties. The analysis of the articulatory features for diphthongs [o͡e], [a͡o] and [a͡e] support the change of the diphthong-endpoint symbols when compared to the Duden transcription. The experiment indicates that this is not specific to the Austrian speakers.

Pronunciation of voiced consonants: According to the analysis, the pronunciation by the two Austrian speakers is much less voiced on average. In the ÖAWB it is claimed that the phone /z/ in the Austrian variety is less voiced. The experiments confirmed this and showed the same phenomenon for other consonants as well.

Different schwa-symbols: There is no evidence from the analysis that the Austrian speakers realize three different versions of the schwa vowel as claimed in the ÖAWB. The experiments show similar distributions for the schwa articulatory features for all speakers. Nevertheless, the variance of the schwa is the highest of all vowels. Therefore a finer distinction into several symbols may be possible, however, the experiments suggest that a narrower schwa-transcription can not be used to emphasize characteristics of a certain variety.

Pronunciation of /o/ and /ɔ/: The ÖAWB claims that the vowel /ɔ/ is realized more closed. This is not supported by the vowel cluster analysis.

Vowels /ɪ/ and /ʊ/: The experiments do not show more overlap between /ɪ/ and /i/ or /ʊ/ and /u/ for the Austrian speakers than for the German and the Swiss ones.

Other differences: Several other phonetic characteristics of the Austrian variety are claimed by the ÖAWB. They are not analyzed in this thesis.

The analysis shows that some phonetic claims in the ÖAWB are supported by the experiments in this thesis. These experiments do not use any phonetic transcriptions from the ÖAWB but instead only use the audio recordings along with its Duden-transcription.

Chapter 8

Conclusion and outlook

In this thesis, automatic phonetic segmentation and labeling of the isolated word corpus of the ADABA corpus was done for all six speakers of three regional language variants (the German in Austria, Germany and Switzerland). The segmentation used no hand-labeled bootstrap data and achieved a high accuracy (>88% within 20ms) for segmentation with pronunciation variants. Based on this segmentation, an analysis of the vowels was performed using an MLP-based approach. The method proved useful for estimating the articulatory features of the vowels, allowing better separability than the formants. A supplementary experiment showed that the articulatory vowel features *backness* and *openness* are closely related to the first two non-linear principal components that an autoassociative ANN estimates from the acoustic features. In a phone classification experiment the MLPs performed slightly better than a HMM/GMM classifier. The articulatory feature approach allowed a comparison of the vowel pronunciation of the six speakers. Analysis with the vowel MLPs revealed some interesting properties of the regional variants, especially of the Austrian variety. All experiments did not rely on phonetic claims of the ÖAWB or the ADABA transcriptions. Instead, only the canonical phonetic transcription (based on the Aussprache-Duden) was used along with the audio recordings. The results of the experiments support several claims of the pronunciation dictionary ÖAWB. In addition, a method to integrate a phone classifier output based on MLPs into an existing forced-alignment based segmentation system was presented.

Bibliography

[AC99] International P. Association and C. A. I. Corporate. *Handbook of the International Phonetic Association : A Guide to the Use of the International Phonetic Alphabet*. Cambridge University Press, June 1999.

[Bau03] Michael Baum. *Improving speech recognition for pluricentric languages exemplified on varieties of German*. PhD thesis, Graz University of Technology, 2003.

[BEK00] Micha Baum, Gregor Erbach, and Gernot Kubin. Speechdat-at: A telephone speech database for Austrian German. In *Proceedings of the LREC Workshop Very Large Telephone Databases (XLDB)*, pages 51–56, Athens, Greece, 2000.

[Bis96] Christopher M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, Oxford, UK, 1996.

[Boe01] Paul Boersma. Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10):341–345, 2001.

[BOW07] Rebecca A. Bates, Mari Ostendorf, and Richard A. Wright. Symbolic phonetic features for modeling of pronunciation variation. *Speech Communication*, 49(2):83–97, February 2007.

[BSH07] Jacob Benesty, M. Mohan Sondhi, and Yiteng (Arden) Huang. *Springer Handbook of Speech Processing*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[Cam96] N. Campbell. Autolabelling japanese tobi. In *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, volume 4, pages 2399 – 2402, October 1996.

[CH68] Noam Chomsky and Morris Halle. *The Sound Pattern of English*, volume 26. Harper & Row, 1968.

[Cly92] M.G. Clyne. *Pluricentric Languages: Differing Norms in Different Nations*. Contributions to the Sociology of Language. Mouton De Gruyter, 1992.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[DM80] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(4):357 – 366, August 1980.

[EA05] Anna Esposito and Guido Aversano. Text independent methods for speech segmentation. In Gérard Chollet, Anna Esposito, Marcos Faundez-Zanuy, and Maria Marinaro, editors, *Nonlinear Speech Modeling and Applications*, volume 3445 of *Lecture Notes in Computer Science*, pages 261–290. Springer Berlin / Heidelberg, 2005.

[Ehr09] Karoline Ehrlich. *Die Aussprache des österreichischen Standarddeutsch – umfassende Sprech- und Sprachstandserhebung der österreichischen Orthoepie*. PhD thesis, Universität Wien, 2009.

[Fan60] Gunnar Fant. *Acoustic Theory of Speech Production*. Mouton, The Hague, 1960.

[For73] Jr. Forney, G.D. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268 – 278, March 1973.

[GH93] Yifan Gong and Jean-Paul Haton. Iterative transformation and alignment for speech labeling. In *Proceedings of EUROSPEECH'93*, volume 3, pages 1759 – 1763, Berlin, Germany, 1993.

[GSCB10] Jon Gómez, Emilio Sanchis, and María Castro-Bleda. Automatic speech segmentation based on acoustical clustering. In Edwin Hancock, Richard Wilson, Terry Windeatt, Ilkay Ulusoy, and Francisco Escolano, editors, *Structural, Syntactic, and Statistical Pattern Recognition*, volume 6218 of *Lecture Notes in Computer Science*, pages 540–548. Springer Berlin / Heidelberg, 2010.

[Her90] Hynek Hermansky. Perceptual linear predictive (plp) analysis of speech. *The Journal of the Acoustical Society of America*, 87:1738, 1990.

[Hos00] John-Paul Hosom. *Automatic Time Alignment of Phonemes Using Acoustic-Phonetic Information*. PhD thesis, Oregon Graduate Institute of Science and Technology, 2000.

[Hos09] John-Paul Hosom. Speaker-independent phoneme alignment using transition-dependent states. *Speech Communication*, 51(4):352–368, April 2009.

[JMK00] D. Jurafsky, J. H. Martin, and A. Kehler. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. MIT Press, 2000.

[Joo48] Martin Joos. Acoustic Phonetics. *Language*, 24(2):5–136, 1948.

[KC02] Yeon-Jun Kim and Alistair Conkie. Automatic segmentation combining an hmm-based approach and spectral boundary correction. In *Proceedings of Interspeech 2002*, pages 145–148, Denver, Colorado, USA, 2002.

[KFL+07] S. King, J. Frankel, K. Livescu, E. McDermott, K. Richmond, and M. Wester. Speech production knowledge in automatic speech recognition. *Journal of the Acoustical Society of America*, 121:723–+, 2007.

[Kir99] Katrin Kirchhoff. *Robust Speech Recognition Using Articulatory Information*. PhD thesis, University of Bielefeld, 1999.

[Kos83] Kimmo Koskenniemi. Two-level morphology: A general computational model for word-form recognition and production. *COLING 84*, 7(Publication No. 11):178–181, 1983.

[Kra91] Mark A. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal*, 37(2):233–243, 1991.

[Kra92] M.A. Kramer. Autoassociative neural networks. *Computers & Chemical Engineering*, 16(4):313 – 328, 1992. Neutral network applications in chemical engineering.

[Lad75] Peter Ladefoged. *A Course in Phonetics*. Harcourt Brace Jovanovich, Inc., 1975.

[Lad05] Peter Ladefoged. *Vowels and Consonants*. Blackwell, Malden, MA, 2005.

[Lew09] M. Paul Lewis, editor. *Ethnologue: Languages of the World*. SIL International, Dallas, Texas, 16 edition, 2009.

[LHGR78] P Ladefoged, R Harshman, L Goldstein, and L Rice. Generating vocal tract shapes from formant frequencies. *Journal of the Acoustical Society of America*, 64(4):1027–35, 1978.

[MD97] Fabrice Malfrère and Thierry Dutoit. High-quality speech synthesis for phonetic speech segmentation. In *Proceedings of EUROSPEECH'97*, pages 2631–2634, Rhodes, Greece, 1997.

[MDDR03] F. Malfrère, O. Deroo, T. Dutoit, and C. Ris. Phonetic alignment: speech synthesis-based vs. viterbi-based. *Speech Communication*, 40(4):503 – 515, 2003.

[MGF08] I. Mporas, T. Ganchev, and N. Fakotakis. A hybrid architecture for automatic segmentation of speech waveforms. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 4457 –4460, April 2008.

[MI00] M. Mangold and Dudenredaktion (Bibliographisches Institut). *Duden: das Aussprachewörterbuch*. Der Duden in 12 Bänden: Das Standarwerk zur deutschen Sprache. Dudenverlag, 2000.

[ML05] Jeff McAffer and Jean-Michel Lemieux. *Eclipse Rich Client Platform: Designing, Coding, and Packaging Java(TM) Applications*. Addison-Wesley Professional, 2005.

[MNEW+10] V. Mitra, H. Nam, C.Y. Espy-Wilson, E. Saltzman, and L. Goldstein. Retrieving tract variables from acoustics: A comparison of different machine learning strategies. *Selected Topics in Signal Processing, IEEE Journal of*, 4(6):1027 –1045, December 2010.

[Muh07] R. Muhr. *Österreichisches Aussprachewörterbuch, österreichische Aussprachedatenbank*. P. Lang, 2007.

[Muh08] Rudolf Muhr. The pronouncing dictionary of Austrian German (AGPD) and the Austrian Phonetic Database (ADABA): Report on a large phonetic resources database of the three major varieties of german. In *Proceedings of the Sixth International Language Resources and Evaluation Conference (LREC'08)*, Marrakech, Morocco, may 2008. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

[NAE08] Daniel Neiberg, Gopal Ananthakrishnan, and Olov Engwall. The acoustic to articulation mapping : Non-linear or non-unique? In *Interspeech 2008 : 9th Annual Conference of the International Speech Communication Association 2008*, pages 1485–1488, 2008.

[OCB10] K.U. Ogbureke and J. Carson-Berndsen. Framework for cross-language automatic phonetic segmentation. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 5266 –5269, March 2010.

[Ost99] M. Ostendorf. Moving beyond the 'beads-on-a-string' model of speech. In *Proceedings of the IEEE ASRU Workshop*, pages 79–84, 1999.

[PB52] Gordon E. Peterson and Harold L. Barney. Control Methods Used in a Study of the Vowels. *J. Acoustical Soc. Am.*, 24(2):175–184, 1952.

[PTVF07] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 3 edition, 2007.

[Rab89] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257 –286, February 1989.

[Rom12] Harald Romsdorfer. Pitch period segmentation of speech signals. Patent, November 2012. EP2519944.

[RP05] Harald Romsdorfer and Beat Pfister. Phonetic labeling and segmentation of mixed-lingual prosody databases. In *Proceedings of Interspeech 2005*, pages 3281–3284, Lisbon, Portugal, 2005.

[Sar97] Warren S. Sarle. Neural network FAQ, periodic posting to the usenet newsgroup comp.ai.neural-nets, 1997. Available from: `ftp://ftp.sas.com/pub/neural/FAQ.html`.

[SC99] Helmer Strik and Catia Cucchiarini. Modeling pronunciation variation for ASR: A survey of the literature. *Speech Communication*, 29(2-4):225 – 246, 1999.

[Sch99] Florian Schiel. Automatic phonetic transcription of non-prompted speech. In *Proceedings of the XIVth International Congress of Phonetic Sciences*, pages 607–610, 1999.

[Sch04] Florian Schiel. Maus goes iterative. In *Proceedings of the LREC 2004*, pages 1015–1018, 2004.

[SK83] David Sankoff and Joseph B. Kruskal, editors. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison.* Addison-Wesley Publishing Co, Reading, Massachusetts, 1983.

[SS87] T. Svendsen and F. Soong. On the automatic segmentation of speech signals. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '87.*, volume 12, pages 77 – 80, April 1987.

[TGG03] D.T. Toledano, L.A.H. Gomez, and L.V. Grande. Automatic phonetic segmentation. *Speech and Audio Processing, IEEE Transactions on*, 11(6):617 – 625, November 2003.

[TW94] David Talkin and Coling W. Wightman. The aligner: Text to speech alignment using markov models and a pronunciation dictionary. In Jan P. H. van Santen, Richard W. Sproat, Joseph P. Olive, and Julia Hirschberg, editors, *Proceedings of Secord ESCA/IEEE Workshop on Speech Synthesis*, pages 89 – 92, 1994.

[VSK12] Dimitrios Ververidis, Daniel Schneider, and Joachim Köhler. The vocal tract latex package. *The PracTEX Journal*, (1), 2012. Available from: `http://tug.org/pracjourn/2012-1/ververidis`.

[vSS99] Jan P. H. van Santen and Richard W. Sproat. High-accuracy automatic segmentation. In *Proceedings of EUROSPEECH'99*, pages 2809 – 2812, Budapest, Hungary, 1999.

[Wag81] M. Wagner. Automatic labelling of continuous speech with a given phonetic transcription using dynamic programming algorithms. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '81.*, volume 6, pages 1156 – 1159, April 1981.

[Wee06] D. J. M. Weenink. *Speaker-adaptive vowel identification.* PhD thesis, University of Amsterdam, 2006.

[Wes03] Mirjam Wester. Pronunciation modeling for ASR – knowledge-based and data-derived methods. *Computer Speech and Language*, 17:69–85, 2003.

[WH00] Alan A. Wrench and William J. Hardcastle. A multichannel articulatory speech database and its application for automatic speech recognition. In *Proceedings of the 5th Seminar on Speech Production: Models and Data & CREST Workshop on Models of Speech Production: Motor Planning and Articulatory Modelling*, pages 305 – 308, Kloster Seeon, Bavaria, Germany, May 2000.

[WHKL08] Gerd Wütherich, Nils Hartmann, Bernd Kolb, and Matthias Lübken. *Die OSGi Service Platform: Eine Einführung mit Eclipse Equinox*. dpunkt, Heidelberg, 2008.

[YEG+06] Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying (Andrew) Liu, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, Valtcho Valtchev, and Phil Woodland. *The HTK Book*. Cambridge University Engineering Department, 2006.

[YRT89] S.J. Young, N.H. Russell, and J.H.S Thornton. Token passing: a simple conceptual model for connected speech recognition systems. Technical report, 1989.

Appendix A

Phonetic alphabets

In this appendix, the phonetic alphabets used during this thesis are listed. The first listed symbol set is the IPA chart containing the international phonetic alphabet. It is followed by the IPA number chart, which assigns numbers to all IPA symbols. Next, the SAMPA Austria is shown, which was generated during the ADABA project. Finally the SYNVOPA, the phonetic alphabet used in the company SYNVO is listed.

THE INTERNATIONAL PHONETIC ALPHABET (revised to 2005)

CONSONANTS (PULMONIC)

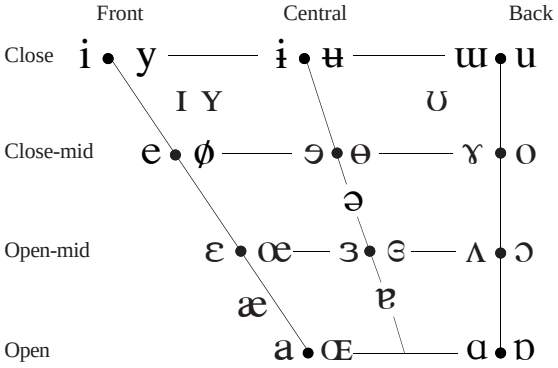
	Bilabial	Labiodental	Dental	Alveolar	Postalveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p b			t d		ʈ ɖ	c ɟ	k g	q ɢ		ʔ
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ		
Trill	ʙ			r					ʀ		
Tap or Flap		ⱱ		ɾ		ɽ					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lateral fricative				ɬ ɮ							
Approximant		ʋ		ɹ		ɻ	j	ɰ			
Lateral approximant				l		ɭ	ʎ	ʟ			

Where symbols appear in pairs, the one to the right represents a voiced consonant. Shaded areas denote articulations judged impossible.

CONSONANTS (NON-PULMONIC)

Clicks		Voiced implosives		Ejectives	
ʘ	Bilabial	ɓ	Bilabial	’	Examples:
|	Dental	ɗ	Dental/alveolar	pʼ	Bilabial
!	(Post)alveolar	ʄ	Palatal	tʼ	Dental/alveolar
ǂ	Palatoalveolar	ɠ	Velar	kʼ	Velar
‖	Alveolar lateral	ʛ	Uvular	sʼ	Alveolar fricative

VOWELS



Where symbols appear in pairs, the one to the right represents a rounded vowel.

OTHER SYMBOLS

ʍ	Voiceless labial-velar fricative	ɕ ʑ	Alveolo-palatal fricatives
w	Voiced labial-velar approximant	ɺ	Voiced alveolar lateral flap
ɥ	Voiced labial-palatal approximant	ɧ	Simultaneous ʃ and x
ʜ	Voiceless epiglottal fricative		
ʢ	Voiced epiglottal fricative		
ʡ	Epiglottal plosive		

Affricates and double articulations can be represented by two symbols joined by a tie bar if necessary. k͡p t͡s

SUPRASEGMENTALS

ˈ	Primary stress	ˈfoʊnəˈtɪʃən	
ˌ	Secondary stress		
ː	Long	eː	
ˑ	Half-long	eˑ	
˘	Extra-short	ĕ	
|	Minor (foot) group		
‖	Major (intonation) group		
.	Syllable break	ɹi.ækt	
‿	Linking (absence of a break)		

DIACRITICS Diacritics may be placed above a symbol with a descender, e.g. n̥ ŋ̊

◌̥	Voiceless	n̥ d̥	◌̤	Breathy voiced	b̤ a̤	◌̪	Dental	t̪ d̪
◌̬	Voiced	s̬ t̬	◌̰	Creaky voiced	b̰ a̰	◌̺	Apical	t̺ d̺
◌ʰ	Aspirated	tʰ dʰ	◌̼	Linguolabial	t̼ d̼	◌̻	Laminal	t̻ d̻
◌̹	More rounded	ɔ̹	◌ʷ	Labialized	tʷ dʷ	◌̃	Nasalized	ẽ
◌̜	Less rounded	ɔ̜	◌ʲ	Palatalized	tʲ dʲ	◌ⁿ	Nasal release	dⁿ
◌̟	Advanced	u̟	◌ˠ	Velarized	tˠ dˠ	◌ˡ	Lateral release	dˡ
◌̠	Retracted	e̠	◌ˤ	Pharyngealized	tˤ dˤ	◌̚	No audible release	d̚
◌̈	Centralized	ë	◌̴	Velarized or pharyngealized	ɫ			
◌̽	Mid-centralized	e̽	◌̝	Raised	e̝	(ɹ̝ = voiced alveolar fricative)		
◌̩	Syllabic	n̩	◌̞	Lowered	e̞	(β̞ = voiced bilabial approximant)		
◌̯	Non-syllabic	e̯	◌̘	Advanced Tongue Root	e̘			
◌˞	Rhoticity	ə˞ a˞	◌̙	Retracted Tongue Root	e̙			

TONES AND WORD ACCENTS

LEVEL			CONTOUR		
e̋ or ˥	Extra high		ě or ˩˥	Rising	
é ˦	High		ê ˥˩	Falling	
ē ˧	Mid		e᷄ ˦˥	High rising	
è ˨	Low		e᷅ ˩˧	Low rising	
ȅ ˩	Extra low		e᷈ ˩˦˥	Rising-falling	
↓	Downstep		↗	Global rise	
↑	Upstep		↘	Global fall	

THE INTERNATIONAL PHONETIC ALPHABET (revised to 2005)

NUMBER CHART

CONSONANTS (PULMONIC)

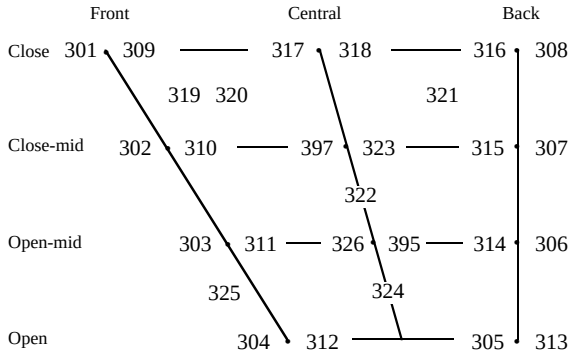
	Bilabial	Labiodental	Dental	Alveolar	Postalveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	101 102		103 104			105 106	107 108	109 110	111 112		113
Nasal	114	115	116			117	118	119	120		
Trill	121		122						123		
Tap or Flap		184	124			125					
Fricative	126 127	128 129	130 131	132 133	134 135	136 137	138 139	140 141	142 143	144 145	146 147
Lateral fricative			148 149								
Approximant		150	151			152	153	154			
Lateral approximant			155			156	157	158			

Where symbols appear in pairs, the one to the right represents a voiced consonant. Shaded areas denote articulations judged impossible.

CONSONANTS (NON-PULMONIC)

Clicks		Voiced implosives		Ejectives	
176	Bilabial	160	Bilabial	401	Examples:
177	Dental	162	Dental/alveolar	101 + 401	Bilabial
178	(Post)alveolar	164	Palatal	103 + 401	Dental/alveolar
179	Palatoalveolar	166	Velar	109 + 401	Velar
180	Alveolar lateral	168	Uvular	132 + 401	Alveolar fricative

VOWELS



Where symbols appear in pairs, the one to the right represents a rounded vowel.

OTHER SYMBOLS

169 Voiceless labial-velar fricative

170 Voiced labial-velar approximant

171 Voiced labial-palatal approximant

172 Voiceless epiglottal fricative

174 Voiced epiglottal fricative

173 Epiglottal plosive

182 183 Alveolo-palatal fricatives

181 Alveolar lateral flap

175 Simultaneous ʃ and x

Affricates and double articulations can be represented by two symbols joined by a tie bar if necessary.

433 (509)

SUPRASEGMENTALS

501 Primary stress

502 Secondary stress ˌfoʊnəˈtɪʃən

503 Long eː

504 Half-long eˑ

505 Extra-short ĕ

507 Minor (foot) group

508 Major (intonation) group

506 Syllable break ɹi.ækt

509 Linking (absence of a break)

DIACRITICS

Diacritics may be placed above a symbol with a descender, e.g. 119 + 402B

402A	Voiceless	n̥ d̥	405	Breathy voiced	b̤ a̤	408	Dental	t̪ d̪
403	Voiced	s̬ t̬	406	Creaky voiced	b̰ a̰	409	Apical	t̺ d̺
404	Aspirated	tʰ dʰ	407	Linguolabial	t̼ d̼	410	Laminal	t̻ d̻
411	More rounded	ɔ̹	420	Labialized	tʷ dʷ	424	Nasalized	ẽ
412	Less rounded	ɔ̜	421	Palatalized	tʲ dʲ	425	Nasal release	dⁿ
413	Advanced	u̟	422	Velarized	tˠ dˠ	426	Lateral release	dˡ
414	Retracted	e̠	423	Pharyngealized	tˤ dˤ	427	No audible release	d̚
415	Centralized	ë	428	Velarized or pharyngealized	209			
416	Mid-centralized	e̽	429	Raised	e̝	(ɹ̝ = voiced alveolar fricative)		
431	Syllabic	n̩	430	Lowered	e̞	(β̞ = voiced bilabial approximant)		
432	Non-syllabic	e̯	417	Advanced Tongue Root	e͖			
419	Rhoticity	327 aˠ	418	Retracted Tongue Root	e͗			

TONES AND WORD ACCENTS

LEVEL			CONTOUR		
512	519	Extra high	524	529	Rising
513	520	High	525	530	Falling
514	521	Mid	526	531	High rising
515	522	Low	527	532	Low rising
516	523	Extra low	528	533	Rising-falling
517	Downstep		510	Global rise	
518	Upstep		511	Global fall	

SAMPA Austria

p	p	101	r\	ɹ	151	7	ɣ	315	_0	◌̥	402A
b	b	102	r\'	ɻ	152	M	ɯ	316	_v	◌̬	403
t	t	103	j	j	153	1	ɨ	317	_h	◌ʰ	404
d	d	104	M\	ɰ	154	}	ʉ	318	_t	◌̈	405
t'	ʈ	105	l	l	155	I	ɪ	319	_k	◌̰	406
d'	ɖ	106	l'	ɭ	156	Y	ʏ	310	_N	◌̼	407
c	c	107	L	ʎ	157	U	ʊ	321	_d	◌̪	408
J\	ɟ	108	L\	ʟ	158	@	ə	322	_a	◌̺	409
k	k	109	b_<	ɓ	160	8	ɵ	323	_m	◌̻	410
g	g	110	d_<	ɗ	162	6	ɐ	324	_O	◌̹	411
q	q	111	J_<	ʄ	164	{	æ	325	_c	◌̜	412
G\	ɢ	112	g_<	ɠ	166	3	ɜ	326	_+	◌̟	413
?	ʔ	113	G_<	ʛ	168	3\	ɞ	395	_-	◌̠	414
m	m	114	W	ʍ	169	@\	ɘ	397	_"	◌̈	415
F	ɱ	115	w	w	170	"	ˈ	501	_x	◌̽	416
n	n	116	H	ɥ	171	%	ˌ	502	_A	◌̈	417
n'	ɳ	117	H\	ʜ	172	:	ː	503	_q	◌̙	418
J	ɲ	118	>\	ʔ	173	:\	ˑ	504	`	◌˞	419
N	ŋ	119	<\	ʢ	174	_X	◌̆	505	#	◌˞	419
N\	ɴ	120	x\	ɧ	175	.	.	506	@`	ɚ	327
B\	ʙ	121	O\	ʘ	176	|	|	507	_w	◌ʷ	420
r	r	122	|\	ǀ	177	||	‖	508	_j	◌ʲ	421
R	ʀ	123	!\	ǃ	178	-\	◌̯	509	_G	◌ˠ	422
4	ɾ	124	=\	ǂ	179	<R>	↗	510	_?\	◌ˤ	423
r'	ɽ	125	||\	ǁ	180	</>	↗	510	~	◌̃	424
p\	ɸ	126	l\	ɺ	181	<F>	↘	511	_n	◌ⁿ	425
B	β	127	s\	ɕ	182	<\>	↘	511	_l	◌ˡ	426
f	f	128	z\	ʑ	183	_T	◌̋	512	_}	◌̹	427
v	v	129	_>	◌ʼ	401	_H	◌́	513	_e	◌̴	428
T	θ	130	_(	◌͡◌	433	_M	◌̄	514	l_e	ɬ	209
D	ð	131	'	◌ˌ		_L	◌̀	515	_r	◌ɹ	429
s	s	132	t\	ʇ		_B	◌̏	516	_o	◌̞	430
z	z	133	S\	ʃ		!	↓	517	=	◌ˈ	431
S	ʃ	134	k\	ɻ		<!>	↓	517	_^	◌̂	432
Z	ʒ	135	5	ɬ		^	↑	518	_h\	◌ɦ	
s'	ʂ	136	C\	ɕ		<^>	↑	518	_y	◌ʸ	
z'	ʐ	137				<T>	↿	519	_s	◌ˢ	
C	ç	138	i	i	301	<H>	↿	520	_x'	◌ˣ	
j\	ʝ	139	e	e	302	<M>	↼	521	_?	◌ˀ	
x	x	140	E	ɛ	303	<L>	↽	522	_?'	◌ˀ	
G	ɣ	141	a	a	304	<B>	↿	523	_r\	◌ɹ	
X	χ	142	A	ɑ	305	_L_H	◌̌	524	_r'	◌ʳ	
R\	ʁ	143	O	ɔ	306	_/	◌̌	524	_r\'	◌ɻ	
X\	ħ	144	o	o	307	_R	◌̌	524	_R\	◌ʁ	
?\	ʕ	145	u	u	308	_H_L	◌̂	525	dz	ʣ	
h	h	146	y	y	309	_\	◌̂	525	dZ	ʤ	
h\	ɦ	147	2	ø	310	_F	◌̂	525	dz\	ʥ	
K	ɬ	148	9	œ	311	_R_F		528	ts	ʦ	
K\	ɮ	149	&	ɶ	312	E\	ɘ		tS	ʧ	
P	ʊ	150	Q	ɒ	313	i\	ɪ		ts\	ʨ	
v\	ʊ	150	V	ʌ	314						

SYNVO Phonetic Alphabet (SYNVOPA)

Number	IPA	SYNVOPA	Example	
101	p	`p`	Spatz	[ˈʃpat͡s]
102	b	`b`	Ball	[ˈbal]
103	t	`t`	Stier	[ˈʃtiːr]
104	d	`d`	dann	[ˈdan]
105				
106				
107	c	`c`		
108				
109	k	`k`	Skandal	[skanˈdaːl]
110	g	`g`	Gast	[ˈgast]
111				
112				
113	ʔ	`?`	beamtet	[bəˈʔamtət]
114	m	`m`	Mast	[ˈmast]
115	ɱ	`F`		
116	n	`n`	Naht	[ˈnaːt]
117				
118	ɲ	`J`	agneau, vigne	[aɲo], [viɲ(ə)]
119	ŋ	`N`	lang	[ˈlaŋ]
120				
121				
122	r	`r`	Rast	[ˈrast]
123	ʀ	`R`	rue, venir	[ʀy], [v(ə)niːʀ]
124	ɾ	`4`		
125				
126				
127	β	`B`	cabra, Habana	[ˈkaβra], [aˈβana]
128	f	`f`	Fass	[ˈfas]
129	v	`v`	was	[ˈvas]
130	θ	`T`	thin, breath	[ˈθɪn], [ˈbreθ]
131	ð	`D`	this, breathe	[ˈðɪs], [ˈbriːð]
132	s	`s`	Hast	[ˈhast]
133	z	`z`	Hase	[ˈhaːzə]
134	ʃ	`S`	Schal	[ˈʃaːl]
135	ʒ	`Z`	Genie	[ʒeˈniː]
136				
137				
138	ç	`C`	ich	[ˈʔɪç]
139				
140	x	`x`	Bach	[ˈbax]
141	ɣ	`G`	viga, burgo	[ˈbiɣa], [buɾɣo]

Number	IPA	SYNVOPA	Example	
142	χ	`X`		
143	ʁ	`*`		[haːʁə]
144				
145				
146	h	`h`	hat	[ˈhat]
147				
148	ɬ	`K`		
149				
150				
151	ɹ	`P`		
152				
153	j	`j`	ja	[ˈjaː]
154				
155	l	`l`	Last	[ˈlast]
156				
157	ʎ	`L`	figlio	[ˈfiːʎo]
158				
160				
162				
164				
166				
168				
169	ʍ	`M`		
170	w	`w`	well	[ˈwel]
171	ɥ	`H`	huile, nuire	[ɥil(ə)], [nɥiːʁ(ə)]
172				
173				
174				
175				
176				
177				
178				
179				
180				
181				
182				
183				
184				

301	i	i	vital	[viˈtaːl]
302	e	e	Methan	[meˈtaːn]
303	ɛ	E	hätte	[ˈhɛtə]
304	a	a	hat	[ˈhat]
305	ɑ	A	bât, pâte	[bɑ], [pɑt(ə)]
306	ɔ	O	Post	[ˈpɔst]
307	o	o	Moral	[moˈraːl]
308	u	u	kulant	[kuˈlant]
309	y	y	Mykene	[myˈkeːnə]
310	ø	2	Ökonom	[ˌʔøkoˈnoːm]
311	œ	9	göttlich	[ˈgœtlɪç]
312	ɶ	&		
313	ɒ	Q	pot	[ˈpʰɒt] [1]
314	ʌ	V	cut, much	[ˈkʰʌt], [ˈmʌt͡ʃ]
315	ɤ	7		
316	ɯ	W		
317	ɨ	1		
318	ʉ	0		
319	ɪ	I	bist	[ˈbɪst]
320	ʏ	Y	füllt	[ˈfʏlt]
321	ʊ	U	Pult	[ˈpʰʊlt]
322	ə	@	halte	[ˈhaltə]
323	ɵ	8		
324	ɐ	6	Ober	[ˈʔoːbɐ]
325	æ	q	hat	[ˈhæt]
326	ɜ	3	bird, furs	[ˈbɜrd], [ˈfɜrz] [2]
395				
397	ɘ	5		
401	t’	t‘		Ejective
402A	n̥	n_0		Voiceless
403	s̬	s_v		Voiced
404	kʰ	k_h	kalt [ˈkʰalt]	Aspirated
405				
406				
407				
408				
409				
410				
411	ɔ̹	O_)		More rounded

[1] British English

[2] American English

412	ɔ̜	`O_(`		Less rounded
413	u̟	`u_+`		Advanced
414	e̠	`e_-`		Retracted
415	ë	`e_"`		Centralized
416				
417				
418				
419				
420	t^{w}	`t_w`		Labialized
421	t^{j}	`t_j`		Palatalized
422	$t^{ɣ}$	`t_G`		Velarized
	$t^{ʁ}$	`t_*`		Uvularized
423				
424	ɛ̃	`~E`	matin [matɛ̃]	Nasalized
425				
426				
427				
428				
429	e̝	`e_>`		Raised
430	e̞	`e_<`		Lowered
431	n̩	`=n`	baden [ˈbaːdn̩]	Syllabic
432	u̯	`^u`	aktuell [akˈtu̯ɛl]	Non-syllabic
433	t͡s	`t_s`	Zahl [ˈt͡saːl]	Affricates
501	ˈ	`'`	(apostrophe)	Primary stress
502	ˌ	`,`	(comma)	Secondary stress
503	aː	`a:`	Bahn [ˈbaːn]	Long
504				
505				
506	.	`-`	(hyphen)	Syllable boundary
507				
508	‖	`#`	(hash)	Phrase boundary
509	a͡u	`a_u`	Haut [ˈha͡ut]	Linking (absence of a break)
		`t_c`	t	Unvoiced preplosive closure
		`d_c`	d	Voiced preplosive closure
		`/`	(slash)	Speech pause
			(space)	Word boundary
	()	`( )`	petit [p(ə)ti]	Optional phone
		`[ ]`	[1]	Syllable stress tag
		`{ }`	#{P:0}	Phrase boundary tag
		`<xxx>`	<150>	IPA number tag