Learning pronunciation variation

A data-driven approach to rule-based lexicon adaptation for automatic speech recognition

by

Ingunn Amdal

A dissertation submitted in partial fulfilment
of the requirements for the degree of

DOKTOR INGENIØR

Department of Telecommunications
Norwegian University of Science and Technology
N-7491 Trondheim
Norway

2002

Norwegian University of Science and Technology
Department of Telecommunications
N-7491 Trondheim, Norway

Dr. ing.-avhandling 2002:100
Rapport nr.: 420206

ISBN 82-471-5502-8
ISSN 0809-103X

Ever'body says words different ... Arkansas folks says 'em different, and Oklahomy folks says 'em different. And we seen a lady from Massachusetts, an' she said 'em differentest of all. Couldn' hardly make out what she was sayin'.

John Steinbeck
The Grapes of Wrath (1939)

Abstract

In this dissertation a complete data-driven approach to rule-based lexicon adaptation is presented, where the effect of the acoustic models is incorporated in the rule pruning metric.

Robust speech recognition is an important research topic, which can contribute to make systems based on automatic speech technology more user-friendly. To achieve a robust system the variation seen for different speaking styles must be handled. In this dissertation we have therefore investigated how to model pronunciation variation for different speaking styles.

The method presented in this dissertation consists of data-driven solutions to all the steps in rule-based pronunciation modelling:

- First an alternative transcription is generated from phone recognition of each utterance, using the acoustic models in order to observe the variation without the restriction of the recognizer's lexicon. We use the same acoustic models as we later will use in the recognition phase for a consistent rule derivation and assessment.

- Alignment of the transcriptions is performed by the traditional dynamic programming approach or by a time synchronous approach. For the dynamic programming a data-driven method to deriving phone-to-phone substitution costs based on the statistical co-occurrence of phones, association strength, is introduced.

- Rules for pronunciation variation are derived from this alignment. The rules are pruned using a new metric based on the acoustic log likelihood. Well trained acoustic models are capable of modelling much of the variation seen, using the acoustic log likelihood to assess the pronunciation rules prevents the lexical modelling from adding variation already accounted for.

- The pruned rules are then used to generate pronunciation variants and the lexicon is modified.

- Adding variants to the lexicon not only corrects errors, but may also introduce new errors. Controlling the added confusability is therefore important. A framework for confusability measures based on decision theory is introduced.

The experiments start with a general investigation of standard automatic speech recognition techniques for different speaking styles. The speaking styles investigated are read speech and spontaneous dictation from native speakers and read non-native speech. A general purpose pronunciation lexicon containing variants and marked canonical pronunciations is used to compare acoustic modelling and lexical modelling of pronunciation variation. Performance of acoustic models of different levels of complexity is also compared. The results show that the lexical modelling using the general purpose variants gave small improvements, but the errors differed compared with using only one canonical pronunciation per word. Modelling the variation using the acoustic models (using context dependency and/or speaker dependent adaptation) gave a significant improvement, but the resulting performance for non-native and spontaneous speech was still far from read speech. Data-driven pronunciation variation modelling is therefore investigated for these two speaking styles.

For the non-native task data-driven pronunciation modelling by learning pronunciation rules gave a significant performance gain. Acoustic log likelihood rule pruning performed better than rule probability pruning. The largest improvement was seen when incorporating the variation for all the speakers in one lexicon, making it possible to use the same lexicon and acoustic models for all speakers.

For spontaneous dictation the pronunciation variation experiments did not improve the performance. The answer to how to better model the variation for spontaneous speech seems to lie neither in the acoustical nor the lexical modelling. One of the main differences between read and spontaneous speech is the grammar used as well as disfluencies like restarts and long pauses. The language model may therefore be the best choice for more research to achieve better performance for this speaking style.

Finally, alignment methods are compared. The association strength scheme for deriving substitution costs is shown to give better performing rules than other dynamic programming methods. The usual dynamic programming approach has difficulties: 1) for transcription errors and disfluencies (usual in spontaneous speech) and 2) when aligning different words (for confusability measures). We also show examples of alignments where a time synchronous alignment may be beneficial to ensure that we compare the same acoustic segments.

Preface

This dissertation is submitted in partial fulfilment of the requirements for the doctoral degree of *doktor ingeniør* at the Norwegian University of Science and Technology (NTNU). The advisors have been Professor Torbjørn Svendsen and Associate Professor Magne Hallstein Johnsen, both at the Department of Telecommunications, NTNU.

The main work has been conducted in the period from April 1998 to October 2001. In addition to the research activity, the work included compulsory courses corresponding to one year full-time studies, as well as half a year of teaching assistant duties. Physically I have spent approximately half the time at the Signal Processing Group in the Department of Telecommunications, NTNU, Trondheim, and half the time at Telenor Research and Development, Kjeller. In the period from January 2000 to July 2000 I stayed at Bell Labs, Lucent Technologies, Murray Hill, New Jersey, USA and worked under the supervision of Filipp Korkmazskiy, Arun C. Surendran and Chin-Hui Lee. Further, finishing touches of the work were given while working as a research scientist at Telenor Research and Development, Fornebu, as well as at a second stay at Bell Labs from February 2002 to May 2002 under the supervision of Eric Fosler-Lussier.

The work has been funded by the Norwegian Research Council (NFR) as a part of the project SPODIS (SPOken DIalogue Systems for telephone services) with additional funding from Telenor. In addition, for both my stays in USA I received partial funding from Bell Labs.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my advisor Professor Torbjørn Svendsen for his support, encouragement, and guidance throughout this work. My sincere thanks also go to Associate Professor Magne Hallstein Johnsen who has always been available for questions and discussions and for being invaluable help in the finishing phase. In the starting phase of my work Trym Holter was of great help and contributed significantly to this work with his ideas on data-driven pronunciation modelling.

During the course of this work I have had the privilege of working with many helpful colleagues at NTNU, Bell Labs, and Telenor. There is a number of people who have given me a lot of help and encouragement along the way, and provided an enjoyable working atmosphere. I would like to thank each of them and I especially appreciate the efforts by Bojana Gajić, Tor André Myrvoll, Ole Morten Strand, Hallstein Lervik, Kirsten Ekseth, and Arne Kjell Foldvik at NTNU. At Telenor Research and Development, I would especially like to thank Knut Kvale, Britt Kjus, Lars Hafskjær, and Endre Skolt.

I am grateful to Chin-Hui Lee at Bell Labs, Lucent Technologies, New Jersey, who gave me the opportunity to stay with his group for six months from January 2000 to July 2000. My supervisors Filipp Korkmazskiy and Arun C. Surendran contributed significantly to the work in this dissertation. I also want to thank Olivier Siohan for his helpful support. In my second three month stay at Bell Labs from February 2002 to May 2002, Eric Fosler-Lussier provided an excellent discussion partner with his comprehensive knowledge on pronunciation modelling.

Finally, I thank my family for they love and support. My parents have always given their encouragement and support in a non-intrusive way I appreciate deeply. Especially, I am grateful to my husband Øyvind Eilertsen for his love, support and unwavering confidence in me. His contribution in practical matters like proof-reading is greatly appreciated, but most important is his contributions in all the matters that are beyond a dissertation.

Contents

List of abbreviations

ANN	Artificial Neural Network
ASR	Automatic Speech Recognition
BLASR	Bell Labs Automatic Speech Recognizer
BPW	Baseforms Per Word
CART	Classification And Regression Trees
CMN	Cepstral Mean Normalization
CMS	Cepstral Mean Subtraction
CMU	Carnegie Mellon University
CRPR	Combined Rule PRobability (individual alignment and joint rule derivation)
GPD	Generalized Probabilistic Descent
HMM	Hidden Markov Model
HTK	Hidden markov model Tool-Kit
IPA	International Phonetic Alphabet
JRPR	Joint Rule PRobability (joint alignment and joint rule derivation)
LDC	the Linguistic Data Consortium
LL	Log Likelihood pruning measure for an acoustic segment
LLH	Log LikeliHood pruning measure for a rule
LPC	Linear Prediction Coefficients
MAP	Maximum A Posteriori
MCE	Minimum Classification Error
MFCC	Mel-Frequency Cepstral Coefficients
ML	Maximum Likelihood
MLLR	Maximum Likelihood Linear Regression
MMI	Maximal Mutual Information
MRPR	Merged Rule PRobability (merging of individually derived rules)
PER	Phone Error Rate
RPR	Rule PRobability

SAMPA	Speech Assessment Methods Phonetic Alphabet
SI-284	WSJ Speaker Independent 284-speaker set
SI-84	WSJ Speaker Independent 84-speaker set
VTLN	Vocal Tract Length Normalization
WER	Word Error Rate
WSJ	Wall Street Journal

Chapter 1

Introduction

Applications using speech technology have for long been a natural part of many futuristic descriptions in books, films, TV-series, and commercials. The technology has over the last years reached a state where we see speech technology based products also being used in the real world. The acceptance by the public increases as the technology improves, but we are still far from the conversational interfaces encountered in fiction. The speech technology applications in real use are limited in one way or the other to give acceptable performance. Limitations often used are e.g. restricted vocabulary, restricted environments, and systems tailored to one or a few users.

Among the most popular type of applications used by the general public are travel information, call centres, and ordering phones. Many different ways of speaking must be tolerated, and the dialogues are usually kept quite restricted to simplify the task for the system. Another use of speech technology is automatic dictation e.g. for medical personnel, where the system is tailored to one person and thus can manage a larger vocabulary.

1.1 Background

A speech technology based system consists of a speech recognition system for speech input, a speech synthesizer for speech output, and a dialogue manager to handle the dialogue. In this dissertation, only the automatic speech recognition (ASR) system will be considered.

Speech input is preferable when the users need to have their hands and eyes free to do other tasks, as in a car or control room environment, or when the device to be controlled is out of reach. Complex actions like asking for several information items at the same time are easier using speech than a graphical interface. Compared to a graphical interface a speech based system can easily

refer to invisible objects. On the other hand there may be ambiguities in referring to the objects. (If a users says "this" the system must interpret what object "this" refers to.) The advantages by using speech rely on a well functioning speech recognition system. With many recognition errors the disadvantage for the user who must perform error-corrections will outweigh the advantages. The advantages of interaction through speech can best be utilized in a multimodal setting. Multimodal user interfaces give the user the opportunity to choose the most suitable mode depending on the situation.

The goal of all research on ASR is to improve the performance. Improved ASR will help make systems based on speech technology more user-friendly and also increase the number of applications that can be speech-enabled. For languages where the ASR research effort has been large, (e.g. US and UK English, Japanese, German) the current technology has reached such a state that useful applications based on speech recognition are possible, e.g. products for controlled tasks like dictation [130]. In the future more and more advanced speech and language based services are expected [16].

With the promising performance of state-of-the-art ASR systems it is possible to strive for further improvements that will handle more speaker and environmental variation. To make a speech recognition product a success there must be few restrictions on the customer's behaviour and environment. If ASR systems can cope with conversational dialogues it is possible to use mixed initiative, not only machine-driven dialogues [132]. This development calls for a deeper understanding of the underlying principles of spoken language [42]. It also calls for an understanding of the recognizer to ensure that variation in speech is addressed correctly. Changes in one part of the recognition system will influence other parts of the system.

Current ASR systems have difficulties with spontaneous speech encountered in conversational dialogues, as well as accent and dialect variability. The substantial differences between non-native speech and native speech will also challenge a "native" ASR system. More subtle differences that are easily handled by humans (e.g. Australian versus US English) may still cause problems for ASR. The speaker dependent variations will mainly be caused by inter-speaker variability, but we will also encounter intra-speaker variability, i.e. the same speaker may behave differently depending on the context (e.g. environment and task) and over time when getting used to a system. In this dissertation we concentrate on the variability that is not on the acoustic level (e.g. the variability due to the differences in the vocal apparatus between individuals), but on the variation on the lexical level that will be similar for groups of speakers. We call this variation different *speaking styles*.

Our objective is to investigate both the similarities and differences among

the different speaking styles and how to best model the variation seen. Even if there are different types of variation, the methods for treating them may be similar.

1.2 Robust automatic speech recognition

All ASR systems must handle variation. The same word spoken several times by the same speaker will vary both in length and acoustical content. For speaker independent speech recognition the voice quality and characteristics will vary even more. The term *robust* automatic speech recognition is used when we consider variation beyond the inter- and intra-speaker acoustic differences we see even for read speech.

The variation in speech input to a speech technology based service may be divided into three groups:

1. Pronunciation variation
2. Grammar and vocabulary variation
3. Channel and noise variation

Pronunciation, grammar and vocabulary variation will be speaker dependent whereas channel and noise variation will depend on the environment. Trying to make ASR handle both speaker and environment variation is crucial in robust modelling. The research groups interested in one of these two issues are often interested in the other one as well, as both areas must be handled for example in public telephone services based on speech recognition.

Different speakers using different speaking styles will use different pronunciations. Spontaneous speech and different dialects or accents are examples of speaking styles with pronunciations that differ from the canonical ones often found in pronunciation dictionaries. The population of most countries becomes more and more multinational, and non-native users will increase the observed variability in pronunciation even more. There will also be differences between expert and novice users of a speech based service (e.g. fast versus over-articulated speech). User-friendly systems should be able to recognize the pronunciations judged appropriate by the user. One of the eight golden rules in user interface design is to "Support internal locus of control" [102]. The user should be spared surprising system actions when using "non-surprising" speech. This will help the user to keep a consistent mental model of the system, which is of great importance for a well-designed dialogue system. If the recognizer makes errors when the user is using rare words or pronunciations, this will be understandable for the user. To make dialogue systems using

speech recognition more user-friendly, robustness to common pronunciation variation is needed.

The task that the speech based application is intended for gives requirements for the recognizer's vocabulary and grammar, and presents another source of variation. The vocabulary and grammar preferred by the user may vary dependent on e.g. non-nativeness, dialect and sociolect, as well as differences between expert and novice users. One cannot assume that the users of a speech technology application will stick to a well-defined grammar (as perhaps for dictation systems). Users may be unwilling to normalize both pronunciation and grammar. They may also be unaware of their own peculiarities. Many perceive their own speaking style as normalized but all these "normalized" variants differ. There was for example an unexpected amount of variation in pronunciation among professional speakers when searching for "model" speakers for Austrian German [83]. Hesitations, restarts, and other disfluencies are also characteristics of spoken language that vary among speakers and must be handled by the language model in a speaker-independent system. Robustness is therefore also needed in modelling grammar variability.

The third main variable that needs to be addressed in speech recognition-based applications is non-speech variation, e.g. noise and channel variation. Both pronunciation and grammar variation are dependent on the user and therefore quite different from this last type of variation that is dependent on the environment. To control how we model the observed variation, we should treat the environment and speaker variation separately. Acoustic model adaptation techniques can model both speaker and environment variations. Explicitly separating these two effects is recognized as an important area for future research [125].

1.3 Pronunciation modelling

It is desirable for speech recognition systems to manage diverse speaking styles, (e.g. spontaneous speech, accents and dialects, and speech from users with different mother tongues), but such variation in user input is difficult for the current state-of-the-art recognizers. One way of improving this is better modelling of pronunciation variation. The pronunciation dictionary is therefore an important part of the ASR. In this dissertation it is called a *lexicon*, a familiar term in the speech community. A lexicon defines the transcription of the words in terms of the acoustic model units of the recognizer. This transcription will not necessarily look like an entry in a pronunciation dictionary made for humans and not machines. This will be treated in more detail in section 2.2.3.

Pronunciation modelling is by no means a new issue in the ASR community,

early efforts are reported in e.g. [8] and [95]. Pronunciation variation modelling is still an important issue in ASR research, and overviews are for example given in [113] and [114]. More recently multilingual ASR has become an interest [1], which introduces new challenges for pronunciation modelling.

Pronunciation variation can be captured using linguistic knowledge, i.e. specific knowledge about how people with different accents pronounce words, but this knowledge is not always sufficient for pronunciation modelling. As an example, a transcription of spontaneous US English speech (Switchboard) revealed at least 80 variants of the word "and" [43]. Non-native speech varies even more, and the phonological rules governing the variation will probably be different for speakers with different mother tongues. In such cases a data-driven approach may be more suitable where we try to extract information from a database containing the speech we want to model. The resulting pronunciation rules will depend on the database and thus on the language, as well as the task and speaking style. This may be favourable for a tailored system as we then only model the variation seen for this specific task. Nevertheless, a method that is independent of the specific database and language is preferable, as it can be reused for other tasks without major modifications.

Linguistic knowledge does not give sufficient information to optimize an automatic speech recognizer. The knowledge varies from language to language, but even for the most studied languages pronunciations are constantly changing and the number of different speaking styles with their characteristics makes it infeasible to have a complete picture. The speech recognition models used today are therefore based on statistical representation and analysis, which must be kept in mind when optimizing the system. A handbook in dialogue design for speech technology by Balentine and Morgan [12] says (page 70):

> "Such models contain statistical-processing artifacts that bear no direct relations to human hearing, and consequently speech recognition often makes mistakes humans would not make."

All parts of the recognizer except the lexicon are usually optimized with respect to objective criteria. A data-driven approach will enable us to use the same criteria for the lexicon as for the other parts of the recognizer, allowing a unified optimization of the whole system. We therefore believe a data-driven approach to pronunciation modelling should be preferred. There is no reason to ignore linguistic knowledge, but the effects of the pronunciation rules derived using either method should be verified on representative speech data.

The statistically based acoustic models of current ASR systems are capable of handling much of the variation seen in speech, also pronunciation variation.

More complex acoustic models will for example handle many allophonic variations in a suitable way. Adaptation of the acoustic models is a successful method to further improve individual recognizers.

Some pronunciation variation can be described as phonological, e.g deletions, insertions, and larger changes (larger both in length and acoustic variation.) This kind of variation may be better handled at the lexical level. Modelling of a group of very different speakers by adapting the acoustic models may result in diffuse models, and in these cases pronunciation modelling by changing the lexicon may give better performance. The two techniques for capturing variation should be combined using the method that gives the best result; acoustic model adaptation for the pronunciation variation at the allophonic level, and lexicon adaptation for the more phonological variation.

Since large vocabulary recognizers always include a language model, the effect of this model should be incorporated in the pronunciation modelling techniques.

One of the main problems in pronunciation modelling is to make sure that we know which variation we are modelling. The effect of the acoustic models, the lexicon, and the language model will interact. The possibility of adding superfluous complexity by modelling the same variation several times, or even worse, adding contradicting changes, must be avoided.

1.4 This dissertation

In this dissertation the focus is on using the lexicon to capture speaker variation, using the same lexicon and the same acoustic models for all speakers. Experiments on individual lexicon adaptation as well as acoustic model adaptation are also presented.

We believe all parts of the system should be optimized in a consistent way. The training of the acoustic and language parts of an ASR system is based on objective criteria, objective criteria should be used for optimizing the lexicon also. This calls for data-driven methods in pronunciation modelling. Knowledge about human perception and production of speech, as well as linguistics and phonetics, is important, but must be formalized for building the ASR system.

Different measures can be used in data-driven pronunciation modelling. We believe the acoustic likelihood should be utilized as a measure in pronunciation modelling. We then take into consideration the variation already modelled by the acoustic models and thereby give a measure consistent with the optimization of the whole ASR system.

One of the major drawbacks with data-driven modelling is that we are

restricted to variation present in the lexicon adaptation data. Direct modelling of pronunciation variation by deriving alternative pronunciations for words present in sufficient numbers in the adaptation data, has been shown to give improvement, e.g. in [49]. To model pronunciations for words not present in the lexicon adaptation data ("unseen words"), it is necessary to extend the method to modelling pronunciation rules, and thereby generalize the variation seen in the adaptation data. This gives many new challenges on how to derive the rules and how to generate and assess pronunciation variants from them.

One reason for the modest improvements achieved in pronunciation modelling is the lack of a way to control the confusability between pronunciations. To make lexica tailored to a person or group we cannot rely on just adding extra pronunciations, we must also remove confusable ones. The use of discriminative methods in choosing which pronunciations to add to the lexicon is one way of solving this problem.

Pronunciation modelling consists of several steps, including alignment of the reference and alternative transcriptions. Although this is a small part of pronunciation modelling (and therefore the whole system), we have investigated alignment methods based on objective criteria in order to be consistent in all parts of the pronunciation modelling.

Last, but not least, it is important to gain knowledge on the effect of the different variation modelling techniques for different speaking styles. The impact of various standard ASR techniques on different speaking styles has been therefore investigated.

1.4.1 Contributions of the dissertation

- **Data-driven pronunciation rule assessment using acoustic likelihood:**
  In this dissertation the acoustic likelihood derivation and selection of pronunciation variants presented in [49] is expanded to derivation and selection of pronunciation *rules*. The advantage of consistent optimization when using an acoustic likelihood based metric is combined with the possibility to model pronunciations for unseen words.

- **Data-driven alignment using dynamic programming with phone-to-phone substitution costs derived using the data:**
  Current alignment methods use costs for phone-to-phone substitutions based on phonetic knowledge or phone identity only. In this dissertation we present a data-driven approach to derive the costs. This method is inspired by the grapheme-to-phoneme conversion in [68]. These costs may be more suitable than phonologically based costs when the alternative

transcription to be aligned is automatically derived without phonological constraints. This method also gives the possibility of non-symmetric mappings.

- **Data-driven pronunciation rule derivation using time synchronous alignment:**
  Another alignment alternative using the time information provided by the ASR system is presented in this dissertation. Using this alignment method we assure we compare the transcriptions of the same acoustic segments. This method also provides pronunciation rules without the need for an extra rule derivation step. Corresponding acoustic likelihood scores are also derived as a by-product.

- **Comparisons of standard variation modelling techniques for different speaking styles:**
  State-of-the art variation modelling techniques are evaluated in this dissertation for different speaking styles. The study is focused on the comparison of acoustic and lexical modelling.

- **A framework for decision theory applied to pronunciation modelling:**
  An assessment method for confusability is important. In this dissertation known decision theory methods, e.g. from [61] and [67], are used to derive confusability measures given a lexicon and an appropriate lexicon adaptation set.

1.4.2 Outline of the dissertation

Chapter 2 describes the basics of the automatic speech recognition system at the level necessary to understand the experiments in this dissertation. The description starts with a short explanation of elements from phonetics and phonology that are relevant for ASR systems. Three ways of modelling pronunciation variation in different parts of the recognizer are outlined. A section on statistical considerations is also included in this chapter. The kinds of variation in spoken language that can be addressed by pronunciation modelling, in this dissertation called "speaking styles", are described in chapter 3. Language modelling topics concerning speaking style variation are also described. Modelling pronunciation variation is also closely related to the adaptation of the acoustic models; this subject is described in chapter 4. Lexicon adaptation is the main subject of this dissertation and is described in chapter 5. Chapters 3, 4, and 5 give an overview of previous work in the field. The theory underlying the experiments in this dissertation is given in chapter

6. Experiments are described in the result chapters 7, 8, 9, and 10. These 4 result chapters conclude with discussions and short summaries. Finally, a concluding summary is given in chapter 11.

1.4.3 List of publications

Parts of the results presented in this dissertation are published in the following publications:

- I. Amdal and T. Svendsen, "Evaluation of pronunciation variants in the ASR lexicon for different speaking styles," in *Proc. LREC-2002*, (Las Palmas de Gran Canaria, Spain), pp. 1290–1295, 2002.

- I. Amdal, F. Korkmazskiy, and A. C. Surendran, "Joint pronunciation modelling of non-native speakers using data-driven methods," in *Proc. ICSLP-2000*, (Beijing, China), pp. III:622–625, 2000.

- I. Amdal, F. Korkmazskiy, and A. C. Surendran, "Data-driven pronunciation modelling for non-native speakers using association strength between phones," in *Proc. ISCA ITRW ASR2000*, (Paris, France), pp. 85–90, 2000.

The results on maximum likelihood variants in section 8.3.2 are based on similar experiments published in:

- I. Amdal, T. Holter, and T. Svendsen, "Modellering av uttalevariasjon for automatisk talegjenkjenning" in *Nordlyd (Tromsø University working papers on language & linguistics)*, vol. 28, pp. 74-87, 2000.

- I. Amdal, T. Holter, and T. Svendsen, "Maximum likelihood pronunciation modelling of Norwegian natural numbers for automatic speech recognition," in *Proc. Norwegian Signal Processing Symposium (NORSIG)*, (Asker, Norway), pp. 145–150, 1999.

Further work on confusability metrics is presented in:

- E. Fosler-Lussier, I. Amdal, and H.-K. J. Kuo, "On the road to improved lexical confusability metrics," in *Proc. ISCA ITRW Pronunciation Modeling and Lexicon Adaptation (PMLA)*, (Estes Park (CO), USA), pp. 53–58, 2002.

Earlier work with relevance to pronunciation modelling is presented in:

- K. Kvale and I. Amdal, "Improved automatic recognition of Norwegian natural numbers by incorporating phonetic knowledge," in *Proc. ICASSP-97*, (Munich, Germany), pp. 1763–1766, 1997.
- K. Kvale and I. Amdal, *Automatic recognition of Norwegian natural numbers over telephone lines*, Telenor R&D report 19/97, 1997.

Chapter 2

The automatic speech recognition system

Speech technology relies on contributions from many different sciences. There are components from acoustics, auditory perception, signal processing, statistics, mathematics, phonetics and phonology, linguistics, and computer science. This chapter gives a brief introduction to the some of the theory that automatic speech recognition is based on.

2.1 Phonetics and phonology

This section will give a brief introduction to parts of phonetics and phonology needed for the explanation of the automatic speech recognition (ASR) system. For a more thorough description of the subject, refer to for example [72], [71], or [24] (in Norwegian).

The set of distinctive sounds that convey meaning are the building blocks used to describe the pronunciation of a language. The smallest unit that can be used to differentiate between words is called a *phoneme*. Quoting Ladefoged in [71] (page 296):

> "**Phoneme** One of a set of abstract units that can be used for writing a language down in a systematic and unambiguous way."

Phoneme transcriptions are shown using "/ ... /" as delimiters. To identify a unit as a phoneme we must be able to find a pair of words whose pronunciations differ only in this unit, a so-called *minimal pair*. An example is the English words "pit" and "bit", where the only difference is the first unit which we will describe as the phonemes /p/ and /b/, respectively.

The variants of a phoneme are called *allophones*, again quoting Ladefoged in [71] (page 291):

> "**Allophone** A variant of a phoneme. The allophones of a phoneme form a set of sounds that (1) do not change the meaning of a word, (2) are all very similar to one another, and (3) occur in phonetic contexts different from one another — for example, syllable initial as opposed to syllable final. The differences among allophones can be stated in terms of phonological rules."

Different accents of the same language will often contain small variations, e.g. in vowel quality, that may be described as allophonic variation. We call this variation allophonic even if this is not strictly correct according to (3) above. This is in line with the use in the speech technology community, e.g. in [63].

A *phone* is a speech unit as it is pronounced, i.e. the realization of the abstract phoneme, quoting Laver in [72] (page 29):

> "**Phone** A speech event capable of displaying phonetic equivalence between speakers"

Phone transcriptions are shown using "[...]" as delimiters to separate them from phoneme transcriptions. In a phone transcription we can add descriptions of the phonetic features of a sound by using diacritics, i.e. small marks that show modifications of the value of a phone symbol or increase the precision, e.g. rounding and aspiration.

A transcription using a simple set of symbols without diacritics is called a *broad phonetic transcription* and shows the symbolic form of the pronunciations. A transcription showing more phonetic detail, e.g. using diacritics, is called a *narrow phonetic transcription* and shows the *surface* form. As an example of a narrow phonetic transcription we use the English words "pit" and "spit": In the first case the [p] will be aspirated and we may want to transcribe this as [p^h ih t] with a diacritic h on the [p] representing aspiration. The word "spit" has no aspiration and will therefore be transcribed [s p ih t][1].

To classify phones we use *phonetic* (also called *acoustic*) *features*. Depending on your field of study, phones may be classified according to different criteria. In addition, certain phonetic features are more prominent in some languages than others, giving yet more different groupings of phones. One standard set of phonetic features with accompanying symbols is defined by the "International Phonetic Alphabet" (IPA) [57]. Another widely used

[1] The CMU phonetic alphabet defined in appendix A is used in the transcription.

classification is defined in the "Speech Assessment Methods Phonetic Alphabet" (SAMPA), [97], which has the nice additional feature of being computer friendly. i.e. using the ASCII-characters. We have mainly used the Carnegie Mellon University (CMU) phonetic alphabet for the transcriptions [15]. This is an alphabet often used for US English and is described in appendix A.

For English the main phonetic features used in phone classification are:

1. Voiced or unvoiced
2. Manner of articulation
3. Place of articulation.
4. Lip-rounding

At the highest level, phones are separated into consonants and vowels, where the vowels are the nucleus (centre) in syllables. (This is a rule with exceptions.) Consonants are usually described using the first three features, and they are often grouped according to the manner of articulation. The main groups used in IPA are: plosives, nasals, trills, tap/flaps, fricatives, laterals, and approximants. For US English pronunciations these are often, e.g. by SAMPA [97], reduced to four broad classes:

1. Plosives
2. Fricatives
3. Affricates; a plosive followed by a fricative (will sometimes be grouped together with fricatives)
4. Sonorants (including nasals, laterals, and approximants).

Vowels are voiced and have the same manner of articulation, so they are classified by the place of articulation, lip-rounding, and in many languages also length (quantity). For a broad classification they may be treated as one group, possibly distinguishing diphthongs from monophthongs. In English the quantity difference is not always pronounced in such a way that a phonologically short vowel is shorter in the actual acoustic segment than a phonologically long vowel. The difference is therefore often only described as a quality difference. SAMPA uses the terms "checked vowels" for short vowels and "free" for long vowels. When we need a more detailed classification of vowels we use the lip-rounding and describe the place of articulation using a matrix. The horizontal classes (matrix rows) for the place of articulation are front, central, and back, whereas the vertical classes (matrix columns) are close, close-mid, open-mid, and open.

A vowel is fairly stationary and is often modelled as a fundamental frequency (pitch) transmitted through a filter describing the vocal tract. This filter can be approximated using an all-pole filter, the basic assumption of linear prediction. Linear prediction is an important model and is used both in speech analysis (e.g. ASR) and speech coding. The frequency response of the vocal tract filter will have distinct peaks called *formants* which characterize the vowels. A plot of the centroid positions of the two first formants (F_1 and F_2), shows how the vowels can be classified according to the formants, see figure 2.1. The first formant is related to the close/open dimension, since a low F_1 corresponds to a "close" articulation. The second formant (or more precisely the difference between the second and first formants) is related to the front/back dimension, since a low F_2 corresponds to a back articulation. This figure also shows the vowel triangle which gives borders of the place of articulation of vowels. The lower right corner is [aa], which is back open, the upper corner is [iy], which is close front, the last corner is [uw], which is back close. A fourth corner may be defined at [ae], which is front open. This is of course a simplified view since we only consider centroids; if we looked at many realizations of the different vowels we would still see a grouping, but not disjoint classes.

Different phone classes have very different acoustic properties. For ASR purposes we consider the speech signal to be slowly varying with time and approximated by a stationary random process for a given (short) time window [91]. This assumption is acceptable for vowels, but does not hold for many of the consonants. The waveform of a speech signal is shown in figure 2.2 to illustrate the differences. We see the characteristic closure and release phases of the plosive [t], which look like a silence (the closure) and a burst (the release phase, which will be influenced by the following phone). The fricatives, such as [f], behave like high frequency "noise"-like signals. The sonorants, such as as [m] and [n], are similar to vowels, but the nasals are special since the air then flows through the nasal tract and the resonant frequencies of the oral cavity will appear as anti-resonances, which gives zeros in the resulting transmission filter. We also note that vowels have more energy than consonants.

An automatic segmentation algorithm based on acoustic similarity of speech segments and a phone recognition system was presented in [70]. The acoustic segments were defined as either transient or steady. A segment was classified as transient when the spectral change exceeded a threshold. A steady segment was defined as a sequence of frames between transient segments. It was shown that when the number of acoustic segments was twice the number of phonemes in a sentence, most of the acoustic segments could be given a phonetic interpretation. This supports the assumption that speech can be

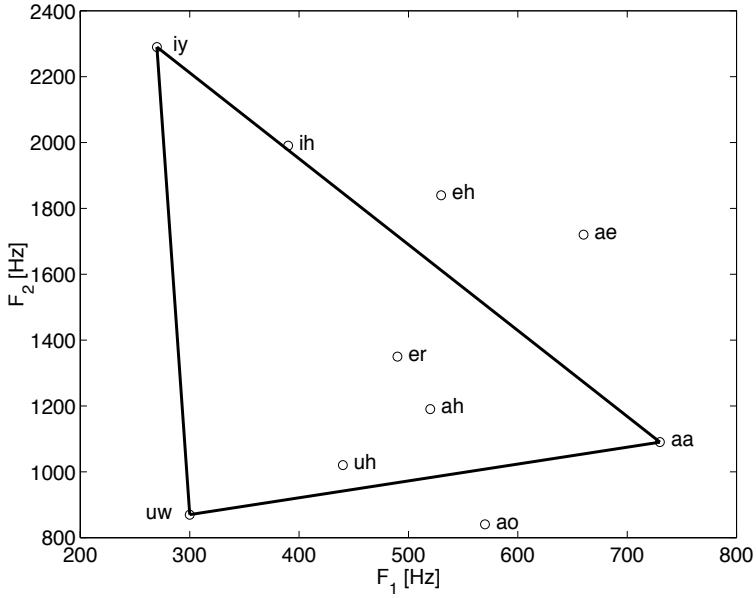

Figure 2.1: First and second formants of US English vowels, after [90] (page 28).

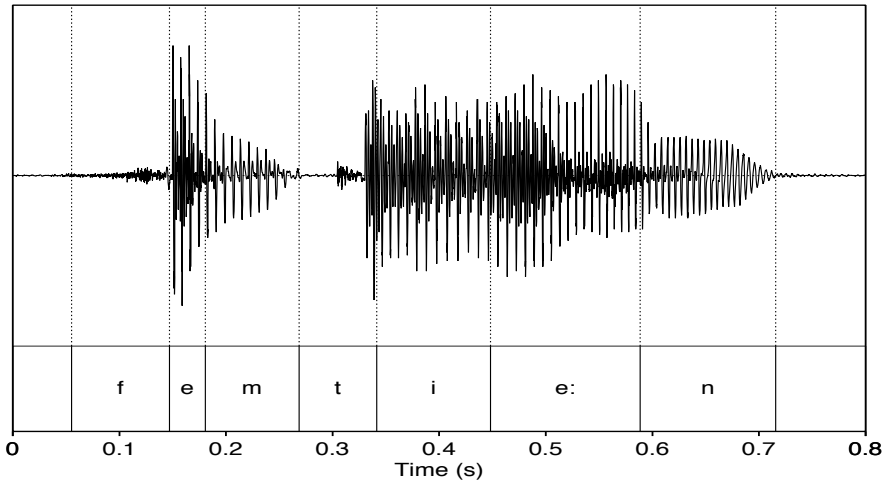

Figure 2.2: Waveform of the utterance "51" in Norwegian, SAMPA transcription.

divided into acoustically fairly stationary segments.

Suprasegmental phonetic features like syllable effects and prosody have an impact on human perception, but are currently not considered in most speech recognizers.

2.1.1 Sub-word units for automatic speech recognition

The *phoneme inventory* of a language is the smallest set of units that is sufficient to define the pronunciations of the words in the language. Since these basic sounds convey difference in meaning, they represent the sounds a recognizer ideally should be able to distinguish from each other. The phonemes are therefore often used as the basis for defining the sub-word speech units for ASR. Depending on the language, the phoneme inventory contains about 30-50 units. The phoneme inventory of one language will usually differ for different dialects (or even accents). Still, it is more convenient to have only one set of units for the speech recognizer even for a system designed to cover several dialects. The variation will then have to be modelled in the acoustic models or the lexicon.

We have in this dissertation chosen to use the term for a realization of the sound, *phone*, to describe the acoustic models. Thereby we have reserved the term *phoneme* for abstract units as used in phonology. All transcriptions are shown using the phone delimiters "[...]". In the ASR community the acoustic models are called phonemes or phones (or even phone-like units) interchangeably. The reason to prefer phone is that the mapping from phonemes to acoustic speech units is not a one-to-one relationship. Some allophones belonging to the same phoneme may be easier to model separately, while other phonemes are better modelled as a joint model. We need a certain number of examples to train a speech unit in the recognizer properly, and lack of coverage for a phoneme may force us to group it with another phoneme.

2.2 Basics of automatic speech recognition (ASR)

In the following, we give a general overview of automatic speech recognition and introduce important terms and results that will be referred to in this dissertation. The presentation given here will focus on the statistically based hidden Markov model (HMM) approach, since this is the most popular technique for ASR systems today. For a more comprehensive treatment of both HMMs and other techniques, see [90], [76] or [52].

The process of automatic speech recognition can be divided into two phases: the training phase and the operational phase. The recognition system is first

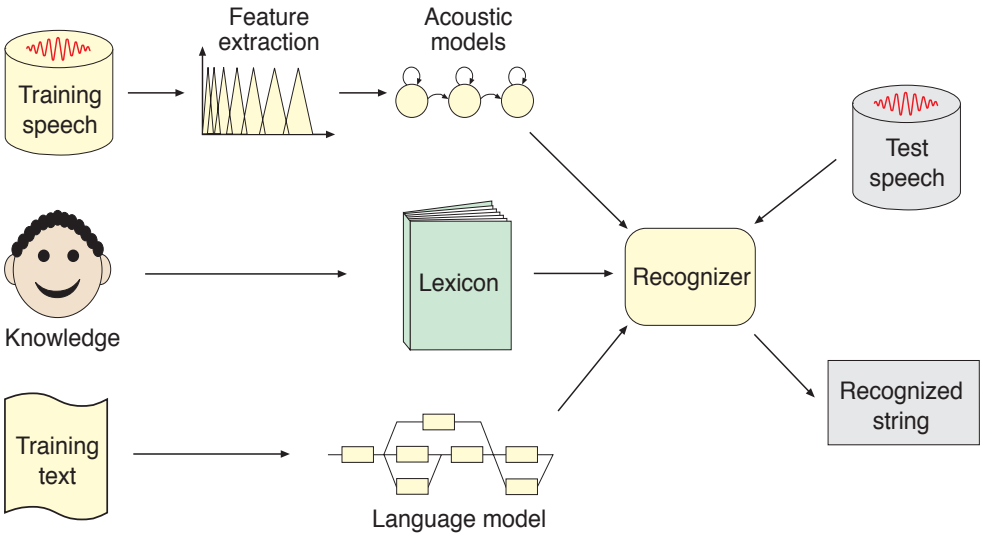

Figure 2.3: Automatic speech recognition system.

trained statistically on sufficient amounts of relevant data and then used to translate input speech into text utterances.

The recognition system may be divided into three main elements as shown in the three lines of figure 2.3:

1. From speech via acoustic features to sub-word units: acoustic models
2. From sub-word units to words: lexicon
3. From words to sentences: language model

The three modules shown in figure 2.3 constitute an automatic speech recognizer. Each part will be explained in more detail in the following sections. A fourth step is the translation from sentences to meaning, but such semantic parsing is not considered in this dissertation.

For a recognizer operating in a real speech-based application, the user input will replace the "test speech" in figure 2.3. The decoder computes a score for the possible sentences consisting of a combination of probabilities for the corresponding acoustic model units, lexicon entries (pronunciations), and language model. We get a hypothesized recognized string as output, possibly several strings ordered in an N-best list or a word lattice and possibly including a confidence measure. In figure 2.3 the "test speech" input is used to show that the performance of a recognizer is evaluated on a well defined test set that makes it possible to compare different recognizers.

Mathematically the system can be described as a classifier. We observe a sequence (i.e. a feature vector representing speech) $\mathbf{O} = (\mathbf{o}_1, \mathbf{o}_2, \ldots \mathbf{o}_T)$ and want to find the word (sequence) W that maximizes the *a posteriori probability* (MAP) which will give us the *Bayes classifier*:

$$\widehat{W} = \underset{W}{\operatorname{argmax}}\, P(W|\mathbf{O}) = \underset{W}{\operatorname{argmax}}\, \frac{p(\mathbf{O}|W)P(W)}{p(\mathbf{O})} \tag{2.1}$$

The observation $\mathbf{O}$ is fixed and $p(\mathbf{O})$ is independent of W so we get the equation:

$$\widehat{W} = \underset{W}{\operatorname{argmax}}\, p(\mathbf{O}|W)P(W) \tag{2.2}$$

$P(W)$ is generally called the probability of the *language model*, while $p(\mathbf{O}|W)$ is the probability density of the *acoustic model*. $P(W)$ and $p(\mathbf{O}|W)$ are estimated from the training set and we get a plug-in maximum a posteriori decision rule. Usually parametric estimation is used, i.e. we assume the form of the distributions and estimate the parameters of these distributions instead of estimating the distributions directly.

2.2.1 Speech pre-processing

The first part of the recognition process is the translation of the speech signal into acoustic features that efficiently represent the information needed by the recognizer. Knowledge about human speech production and perception has been important in finding the best techniques for deriving the features from the speech waveform. For pronunciation modelling it is important to understand the limitations of the speech pre-processing.

Mel-frequency cepstral coefficients (MFCC) are the most widely used acoustic features at present. Other popular features are linear prediction coefficients (LPC) and perceptually weighted linear prediction coefficients (PLP). The process of converting the speech waveform into cepstral based acoustic features can be divided into the following steps:

1. Windowing and pre-emphasizing
2. Converting to a spectral envelope representation
3. Converting to a cepstral representation including cepstral liftering
4. Adding dynamic features

The waveform is often sent through a simple high-pass filter to emphasize the high frequency portion of the signal since the low frequency part has much

more energy. This is called *pre-emphasizing*. The signal is then divided into segments using windowing, such a segment of speech is often called a *frame*. The window size is chosen to give fairly stationary segments.

For the spectral envelope representation in MFCC we use a filter bank with bands divided according to a Mel scale. Division into bands of equal number of Mels corresponds to division into critical bands which are designed according to partial loudness. The critical bands correspond approximately to articulation bands which are based on speech perception, see e.g. [2]. These types of filter banks have been shown to correspond to spatial distances along the basilar membrane in the human ear.

The cepstrum is the inverse Fourier transform of the logarithm of the speech spectral representation. It was originally used in order to separate the source (the glottal excitation) and the filter (the vocal tract) as additive elements. The convolutional distortion from the transmission channel will be additive in the cepstral domain. A popular way of removing channel dependency is therefore to normalize the cepstral coefficients, so-called Cepstral Mean Normalization (CMN) or Subtraction (CMS). This technique was first proposed in [7] but did not gain popularity until in the early 1990s, according to [52]. The low- and high-order cepstral coefficients have been shown to have higher sensitivity due to sources of variability not essential in representing the phonetic content of a sound. Liftering of cepstral coefficients (i.e. emphasizing the middle-order coefficients) is therefore used to give more reliable estimates.

The temporal change of speech has been shown to be important, and the static parameters extracted using MFCC are augmented using dynamic parameters. These are usually the first and second order time-derivatives and are often called delta and delta-delta parameters, respectively.

The resulting feature vector of coefficients is then a representation of the information in one speech frame that is believed to be relevant for ASR. A recognizer front-end will typically be described by the window size, frame rate, filter bank type, the number of (static) coefficients, type of energy, and the use of deltas and delta-deltas.

Limitations

The search for better features is an important subject for the ASR community, especially finding features that are more robust to environmental variation (e.g. noise and reverberation). Transcriptions of spontaneous speech show the limitations of current speech pre-processing also regarding variation in pronunciation. Current speech pre-processors treat all segments in the same way, although we know that the stationarity assumption is not valid for other units than vowels, while the window length is either too short or too long to

capture all speech effects. The different phones have very different properties, see section 2.1, so to model e.g. plosives we may need shorter windows, while modelling syllable effects demands longer windows.

Using features more closely related to the auditory properties of human perception is interesting, especially for improved performance in noisy and reverberant environments. The problem is that auditory model methods usually are computationally expensive. A related approach with complexity comparable to conventional front-ends was presented in [34].

2.2.2 Acoustic models

The acoustic models constitute the main part of the recognizer. Much of the variability in the realization of a speech unit is here modelled in a statistical framework that combines and quantifies the variation using probabilities. The variation that the acoustic models handle can be as different as environmental variation and variation between speakers. The same unit uttered by the same speaker on different occasions will also vary in a way that we conveniently can model by statistics; e.g. duration and timbre. Traditionally, much of the pronunciation variation is modelled in the acoustic models.

Acoustic models are trained on acoustic features derived from recorded speech with accompanying transcriptions, telling the recognizer which acoustic segment corresponds to which unit. These models represent units of speech, usually words or sub-word units, as mentioned in section 2.1. In a sub-word based recognizer we need a pronunciation lexicon to state the correspondence between the sub-word units and the words we wish to recognize.

The phone-like units are often context-dependent, i.e. different models are used for the same unit depending on its neighbours. A set of context-independent phones for the acoustic model inventory is often called *monophones*. A context-dependent phone called *triphone* is a phone given its immediate left and right neighbours. This is perhaps the most popular acoustic model unit because it can handle allophonic variation and coarticulation effects better than context-independent models. Longer contexts like quintphones have also been tried, e.g. in [25], showing especially advantageous results for spontaneous speech. Not all possible triphones will be present in the training data and we need tying of parameters between the models to account for the unseen triphones.

The most popular acoustic models are the statistically based Hidden Markov Models (HMMs) and artificial neural networks (ANNs). Both models have their advantages and disadvantages. A system for combining several ASR outputs called ROVER [27] has shown that combinations sometimes can achieve better results than the best individual system.

Hidden Markov Models

The use of hidden Markov processes in ASR was first reported in [11] and [59]. The introduction of HMMs was a revolution in the ASR history. Many limitations and inherently non-valid assumptions about the speech signal have been addressed, see e.g. [120], but no competing model has had the same impact, yet. Given sufficient relevant training data this statistically based model has made more and more sophisticated ASR applications possible.

The description of HMMs given here uses notation and inspiration from Rabiner and Juang in [90]. For each acoustic frame we will have an observation $\mathbf{o}_t$. We describe an HMM by its states $\mathbf{q}$ and parameters $\theta = (A, B, \pi)$:

- The state at time t is denoted q_t. An observation sequence $\mathbf{O} = (\mathbf{o}_1, \mathbf{o}_2, \ldots \mathbf{o}_T)$ will correspond to a state sequence $\mathbf{q} = (q_1, q_2, \ldots q_T)$
- The transition probabilities are $A = \{a_{ij}\}$, where $a_{ij} = P[q_{t+1} = j | q_t = i]$
- The observation probability distributions are $B = \{b_j(\mathbf{o}_t)\}$, where $b_j(\mathbf{o}_t) = p(\mathbf{o}_t | q_t = j)$
- The initial state distribution is $\pi = \{\pi_i\}$, where $\pi_i = P[q_1 = i]$

The basis of a Markov model is a finite state machine where the probability of being in a given state depends only on a fixed number of previous states. For ASR we usually use first order Markov models where the probability depends only on the previous state:

$$P[q_{t+1} = j | q_t = i, q_{t-1} = i_1, \ldots] = P[q_{t+1} = j | q_t = i] \tag{2.3}$$

The model is called *hidden* because the state sequence $\mathbf{q}$ in the Markov model is linked to the observation through a stochastic process, i.e. "hidden". We therefore have a double stochastic process, first the stochastic Markov state model and then the stochastic process from the states to the observations. The observation stochastic process is described by the observation probability densities $b_j(\mathbf{o}_t)$. If we have a discrete alphabet of observation symbols $\mathbf{v}_k$ corresponding to discrete observation densities, we have $b_j(k) = P[\mathbf{o}_t = \mathbf{v}_k | q_t = j]$. In many ASR systems the observation density is continuous and consists of a mixture of Gaussian components. Usually diagonal covariance matrices with several mixtures are used rather than full covariance matrices with fewer mixtures.

The realization of a speech unit (e.g. word or phone) will vary in duration from utterance to utterance. It is virtually impossible for a speaker to utter the same pronunciation several times with each segment having exactly the same

length. The statistical framework of transition probabilities a_{ij} takes care of this variability. For normal speech sounds we often only allow transitions to higher numbered states, and skipping states may be prohibited. The transition matrix therefore usually has some elements set to zero.

The number of states in the HMM and the connection between them are called *topology* decisions. A phone HMM will often consist of three states, allowing state transitions to higher numbered states or a loop to the same state only. This is called a left-right topology and the start, the middle, and the end of the phone can then be modelled differently. We will often chose other topologies for silence, noise, and filler models than for ordinary phones. For non-speech units like these it is reasonable to add more transitions to allow more variation in length and in the order of the states.

The HMMs are usually trained in a maximum likelihood (ML) framework, i.e. the parameters $\theta = (A, B, \pi)$ are unknown constants:

$$\widehat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\mathbf{O}|\theta) \tag{2.4}$$

Assuming statistical independence of observations we can express the probability of the observation sequence $\mathbf{O}$ given the model as a joint probability over all possible state sequences:

$$p(\mathbf{O}|\theta) = \sum_{\text{all } \mathbf{q}} p(\mathbf{O}|\mathbf{q}, \theta) P(\mathbf{q}|\theta) \tag{2.5}$$

The two stochastic processes of the model are given by $P(\mathbf{q}|\theta)$ and $p(\mathbf{O}|\mathbf{q}, \theta)$. The probability of the observation given the state sequence $p(\mathbf{O}|\mathbf{q}, \theta)$ will be a function of the observation probabilities $b_j(\mathbf{o}_t)$, while $P(\mathbf{q}|\theta)$ will be a function of the transition probabilities a_{ij} and the initial state distribution π_i:

$$p(\mathbf{O}|\mathbf{q}, \theta) = b_{q_1}(\mathbf{o}_1) \cdot b_{q_2}(\mathbf{o}_2) \cdots b_{q_T}(\mathbf{o}_T) \tag{2.6}$$

$$P(\mathbf{q}|\theta) = \pi_{q_1} \cdot a_{q_1 q_2} \cdot a_{q_2 q_3} \cdots a_{q_{T-1} q_T} \tag{2.7}$$

Log likelihood is used instead of likelihood since the probabilities will be small numbers. This also conveniently gives summing of the observation probabilities $p(\mathbf{o}_t|q_t, \theta) = b_{q_t}(\mathbf{o}_t)$ instead of multiplication:

$$\log p(\mathbf{O}|\mathbf{q}, \theta) = \log \prod_{t=1}^{T} p(\mathbf{o}_t|q_t, \theta) = \sum_{t=1}^{T} \log p(\mathbf{o}_t|q_t, \theta) \tag{2.8}$$

To estimate the parameters of the model we use a training set of speech signals with accompanying transcriptions. There are several well known estimation strategies for this that will not be described here. The most popular

methods are segmental K-means and reestimation with Baum-Welch (the EM algorithm). These two methods may be used alone or in combination. Further details can be found in e.g. [90].

For decoding, i.e. finding the output symbol that best represents the utterance, we find the HMM that best explains the observation. This is done by computing the log likelihood for each HMM and selecting the one that gives the highest value. The HMM is represented by its set of parameters θ:

$$\widehat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\mathbf{O}|\theta) = \underset{\theta}{\operatorname{argmax}} \sum_{\text{all } \mathbf{q}} p(\mathbf{O}|\mathbf{q},\theta) P(\mathbf{q}|\theta) \tag{2.9}$$

This corresponds to the acoustic model part of equation (2.2). For isolated word recognition using word HMMs a maximum likelihood for HMM θ_i will correspond to recognizing a word W_i. For continuous speech recognition (or when using sub-word units) the recognized output will correspond to a sequence of HMMs.

In practice the Viterbi algorithm, which is an efficient way of finding the best path of a network, is used instead of this maximum likelihood approach. Instead of summing the likelihoods for all sequences we use the likelihood of the best path to represent the likelihood of the model:

$$\widehat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\mathbf{O}|\theta) \approx \underset{\theta}{\operatorname{argmax}} \left(\max_{\mathbf{q}} \left[p(\mathbf{O}|\mathbf{q},\theta) P(\mathbf{q}|\theta) \right] \right) \tag{2.10}$$

The Viterbi algorithm is more efficient because we only need to keep track of one path at each state. Even this will require many computations, and usually some kind of pruning is performed. The Viterbi solution will in this case be suboptimal. With pruning, we either keep only the paths with likelihoods better than a likelihood offset relative the best path, or we limit the computations to a fixed number of best paths. Even if we maintain all paths, the solution will not necessarily be the ML solution. In [60] a "forward" decoding using total likelihood score performed significantly better than Viterbi for discriminatively trained models. The same effect was presented in [98], where using all pronunciations in decoding instead of only the best gave increased performance.

Limitations

An HMM based speech recognition system consists of a large number of parameters to make sure that all variabilities are represented. For large vocabulary continuous speech recognition in the order of a few million HMM parameters are needed [74]. To train all these parameters large amounts of

speech are needed. The best performance is obtained with speech recorded in the same type of environment in which the recognizer will operate. This may include background noise like car noise, music, or the extra variation introduced by hands-free microphones. Collection of environment specific databases is a tedious task (see e.g. [48]), as illustrated by the pricing of such databases. The SpeechDat-Car database is recorded by using subjects in real car environments and in various situations. This database costs about four times as much as the ordinary SpeechDat database which is recorded over a fixed telephone network where the subjects mostly call from home or an office, [23]. Human speech recognition does not suffer from the same demand of relevance in training. Learning a new language from clean speech is actually preferred even when later hearing this language in noisy environments.

2.2.3 Lexicon

If other units than words are used for the acoustic models, we need a correspondence between the acoustic model units and the words in the vocabulary. When phones form the basic acoustic model units, this corresponds to, but is not necessarily equal to, a pronunciation dictionary. In the ASR community the term *lexicon* is often used instead of dictionary for this link. The middle line in figure 2.3 shows that the lexicon is often based on knowledge contrasting the optimization based on speech data and objective criteria that is used in the other parts of the recognizer.

We use the word *baseform* for the connection between acoustic models and a lexicon entry. The baseform describes a word as a sequence of acoustic models which is not necessarily equal to the form that we find in a pronunciation dictionary made for humans and not machines. The baseform is the lexical item representation and we reserve the word *pronunciation* for the realization. Up till now most ASR systems use only one canonical or a few baseforms per word, and these baseforms have typically been hand-made using phonetic knowledge.

Some examples of entries in the CMU lexicon are given in table 2.1. A number after the word indicates a non-canonical baseform. The numbers after the vowels indicate stress, not all ASR lexica provide this. Syllable information is also given in some ASR lexica. First of all we note that the word class is usually not indicated. A consequence of this is that there is no distinction between *homographs* (words with the same spelling but different meanings) with different pronunciations, such as the two tenses of "read" in table 2.1. Often an ASR lexicon will have only one entry in such cases. For ASR lexica with multiple entries, several baseforms can be given for one word, e.g. "realize". There is usually only one entry for words with different

Word	*Baseform*
READ	`r eh1 d`
READ (2)	`r iy1 d`
READABILITY	`r iy2 d ah0 b ih1 l ih0 t iy0`
READABLE	`r iy1 d ah0 b ah0 l`
READER	`r iy1 d er0`
READER'S	`r iy1 d er0 z`
REALIZE	`r iy1 l ay2 z`
REALIZE (2)	`r iy1 ah0 l ay2 z`
RIGHT	`r ay1 t`
TOO	`t uw1`
TWO	`t uw1`

Table 2.1: Examples of ASR lexicon entries

senses, but with identical spelling and pronunciation. An example of this is "right" which can mean both a direction and a notion of correctness in addition to other senses[2]. This is called *homonymi* and can be a problem because information that could be useful for language modelling and semantic parsing is hidden. *Homophones* are words that have different meanings (and usually different spellings), but that have the same pronunciations. These words will have separate entries, as shown for "two" and "too", but the ASR system must rely on the language model to resolve which word is recognized. This is the same as for human speech recognition, except that we usually have more contextual information available, such as the setting and theme for the spoken utterance and the identity of the speaker (this is a topic of pragmatics). In ASR such knowledge is incorporated by using task and dialogue state dependent language models.

When multiple baseforms are used for each word, pronunciation probabilities may be used to inhibit confusions due to rare pronunciations. Pronunciation probabilities are often defined as part of the language model instead of the lexicon. From a hand-labelled part of Switchboard 36 different pronunciations were found for "the" in the test set, and 38 different pronunciations in the training set. Only half of the variants found in the training set were also observed in the test set. The confusability by adding all observed variants can also be illustrated by the 35 different words that had the pronunciation `[ax]` (schwa) [82].

[2]The Concise Oxford Dictionary lists 6 adjectives, 3 adverbs, 4 nouns, and 2 verbs for the word "right".

2.2.4 Language model

The last module in the recognizer comprises the combination of words into sentences as shown in line three of figure 2.3. The language model used in ASR may be linguistically derived, but usually statistical grammars trained on large text corpora are used.

N-gram is a statistical grammar where the probability of a word given the N previous words, the *word history*, is calculated from the training material.

$$\widehat{P}(w_i|w_{i-1},\dots w_{i-N+1}) = \frac{\text{count}(w_i, w_{i-1},\dots w_{i-N+1})}{\text{count}(w_{i-1},\dots w_{i-N+1})} \tag{2.11}$$

Unigram (1-gram) uses no word history and is the same as word frequency. Bigrams or trigrams are used in most medium and large vocabulary recognizers. *Perplexity* is a measure of the predictability in the language model. The perplexity is related to the entropy of the grammar and can roughly be explained as the average number of possible words that can follow a given word. If no language model is present, the perplexity will equal the number of words in the vocabulary. Lower perplexity means that the move from word to sentence level in the recognition process will be less complex.

Huge amounts of data are needed to train a statistical language model, and even then all word combinations will not be found. The problem of sparsity is even more explicit for language modelling than for acoustic modelling because there are many more words than phones. Normally, smoothing techniques have to be used for the unobserved N-grams. In the backoff strategy the corresponding probabilities are computed from probabilities in lower order models.

A recognition network is made by combining the acoustic models into a network defined by the language model and the lexicon. The larger the language model, the larger the recognition network, and the more important are efficient search algorithms. A static decoder (generating the entire recognition network before decoding) is infeasible when using large language models. In this case dynamic decoders are necessary, since they are faster and need less memory during decoding. Often a first pass with a simpler language model (and acoustic models) is used to make a word lattice that can be rescored using more advanced models in a second pass (possibly after local adaptation of acoustic models.)

2.3 Discriminative methods in ASR

For a speech recognizer the most important evaluation measure is the error rate or corresponding continuous measures [61]. Minimum error rate is a more

important goodness criterion for the performance of a recognizer than correct class modelling, and it should also be the metric used in the training of the recognizer. This calls for the use of decision theory in optimizing the ASR system, i.e. discriminative modelling.

Discriminative methods in ASR have mainly been utilized in optimization of the acoustic models. One important advantage with ANN acoustic models is that discriminatively trained models are easy to implement, but HMMs or a combination may also be used. Minimizing classification error (MCE) using the generalized probabilistic descent (GPD) method was introduced in [61] and [64]. A conditional maximum likelihood (CML) for discriminative modelling of the acoustic models was introduced in [60]. This is similar to maximum mutual information (MMI) but does not assume a fixed language model. The main difference between GPD and MMI type modelling is that GPD uses a smoothed error count instead of a linear function of a misclassification measure. Other optimization techniques directly related to the error rate include corrective training and minimum empirical error.

Discriminative training can give better or comparable performance with less complex acoustic models. The popularity of discriminative training in ASR systems has however been modest. For high complexity systems, maximum likelihood training methods are unbeaten, while discriminative approaches are useful in real-time applications, since the system requires fewer parameters.

The basis of discriminative methods is an optimizing criterion that is directly linked to error rate. We model not only according to the right class (word), but also the competing classes. In this way we put more emphasis on the estimation of the borders between classes than on modelling the typical sample.

Two basic problems generally arise when training a speech recognizer:

1. Unseen data: We need sufficient data to achieve sufficiently dense sampling of the model we want to build.

2. Systematic errors: We need relevant data that represents the speech we want to recognize to avoid systematic errors.

We will usually not have sufficient amounts of data for all cases that we want to model, and it is therefore important that the chosen estimation method generalizes well. For a discriminative approach we use each data sample to estimate several parameters, because the sample will appear not only in the training material for the correct utterance but also for the competing (incorrect) utterances. On the other hand, the relations between the classes are important, and shortage of training samples for one class will have a bad effect not only on that class, but also the other classes. Lack of data can be

alleviated using model assumptions which decrease the number of parameters that need to be estimated (or tying of parameters). The crucial point is then whether the model assumptions are correct or at least reasonable.

The second problem is best solved with carefully chosen training data. Systematic mismatches between the training data and the data on which the classifier will be used are difficult to handle even with well-designed training methods. If a smaller set of more relevant data is available, speaker or environment adaptation is one solution.

2.4 How to model pronunciation variation in ASR

Pronunciation variation modelling can be implemented in different parts of the speech recognizer. Different kinds of pronunciation variation can be modelled in the three parts shown in figure 2.3, but the same variation can also be modelled differently in different parts. Different pronunciations of phonemes (allophonic variation) can e.g. be handled using either more acoustic units, more complex models (but less units), or more baseforms. The allophonic variation can also be handled by adapting the models to one speaker or a more homogeneous subset of speakers.

Adaptation of the acoustic models is a successful method to make speaker-dependent recognizers with improved performance compared with speaker independent recognizers. Task adaptation or adaptation to a group of speakers (e.g. dialect adaptation) is possible, but the adaptation target should be kept homogeneous. Modelling large variations within the same model by broadening the distributions or adding more components in the Gaussian mixture will give more diffuse models which may be more overlapping, [119]. The success of this type of speaker adaptation also depends on the match between the actual pronunciations and the transcription.

Some of the pronunciation variation is caused by speaking style (including dialects, non-native mother tongue etc.), and may be better handled by careful design of the pronunciation dictionary, i.e. pronunciation modelling, [62]. The most common way of dealing with pronunciation variation is to put several baseforms in the ASR lexicon. These baseforms are also often used to retranscribe the speech corpus before a retraining of the acoustic models. Using the lexicon to capture speaker variation makes it possible to model several speakers simultaneously, thus using the same lexicon and the same acoustic models for all speakers. A multiple baseform lexicon will affect the language model. We can introduce pronunciation probabilities in addition to the word probabilities given by the language model, but we must take care not to change word probabilities depending on the number of baseforms.

High quality recognizers always include a language model, and it should therefore be incorporated in the pronunciation modelling techniques. For large vocabulary speech recognition a well designed language model may lessen the negative impact of a mismatch between the speaker and the acoustic models and explicit pronunciation modelling may be less important. If the speaking style we try to model has special language model characteristics, they may be incorporated directly into the language model, e.g. the hesitations and restarts of spontaneous speech.

One of the main challenges in pronunciation modelling is to make sure that we know which variation we model. The effects of the acoustic models, the lexicon, and the language model will interact. Even the choices at the speech pre-processing stage will influence the variation modelling. Superfluous complexity may result from modelling the same variation several times or, even worse, adding contradicting changes.

The two main techniques for capturing variation, acoustic model adaptation and lexicon adaptation, should be combined using the method that gives the best result: Acoustic model adaptation for the pronunciation variation that can be described as allophonic and lexicon adaptation for the more phonological variation like deletions and insertions.

Although not in widespread use, some experiments using discriminative modelling of pronunciation variation have been reported, see [67] and [99]. In section 6.1 we give a framework for decision theory applied to pronunciation modelling.

2.5 Statistical considerations

The standard evaluation metric for an automatic speech recognition system is the *word error rate* (WER). The word error rate is computed by aligning the reference and hypothesized utterances and counting substitutions, deletions, and insertions. All these are counted as errors and we get:

$$\text{WER} = \frac{\text{Substitutions} + \text{Deletions} + \text{Insertions}}{\text{Total number of words}} \times 100\% \tag{2.12}$$

Note that the WER may be larger than 100%. Sometimes the word *accuracy* is used, this is 100% − WER. The result may also be given as percentage *correct*, which is used when insertions are not counted.

The size of the test set will influence the significance of the increase or decrease in word error rate. This subject has been addressed in the speech community in several publications, e.g. [39], [47], and [115].

In this dissertation, confidence intervals for word error rates are computed using hypothesis testing assuming that word errors are independent. This

is a standard statistical method described in e.g. [56] and translated to a speech recognition framework in [47]. For tests using a language model the assumption is not true and these confidence intervals will give a too small region. The confidence intervals for the test sets used in this dissertation are given in appendix C.

Other tests more suitable when a language model is in use are available from NIST [85] and in methods proposed in [115]. The NIST methods are based on sentence error rate since sentences can be considered to be independent, even when using a language model. The test sets reported in this dissertation contain too few sentences to use sentence error rate as a useful measure.

Looking at the errors corrected versus the errors introduced is also useful [123], and this kind of error analysis has been utilized for many of the experiments reported. Comparing the errors that differ between two systems, the McNemar test can be used to assess the significance of the differences in word error rate, [39]. The McNemar test requires that errors are independent, which is not the case in our setups since we have applied a bigram. Still, the test give more information than only word error rate comparisons, and we have chosen to use the term "significant" when the McNemar test gives a p-value less than 0.01.

Chapter 3

Speaking styles

In this dissertation the term *speaking style* is interpreted widely. We do not only cover read versus conversational speech but include descriptions of dialects and accented speech as well as non-native speech. The reason is that all these different "speaking styles" deviate from a canonical lexicon and language model and are therefore parts of the variation that a robust ASR system must handle. In [1] Adda-Decker gives an overview, although focused towards multilinguality, that also covers the evolution of ASR systems to capture variability. She divides the main challenges for ASR in the four axes shown in figure 3.1, where speaking style is one of the axes. Using the wider definition of speaking styles as in this dissertation we also include some of the challenges on the "speaker" axis.

The first ASR systems only considered restricted speaking styles, i.e. careful articulation of isolated or connected words. The increased modelling capacities of current ASR systems also manage the looser articulation of continuous speech. The continuous speech can be divided further into read, prepared, and spontaneous speech (with increasing degree of variation encountered). The DARPA and NIST evaluations [85] mirror the progress in the ASR community as the evaluation tests have evolved from constrained tasks and read speech to broadcast news transcriptions which include both prepared and spontaneous conversational speech, as well as non-native speech and a variety of environment challenges (noise, music, background talk etc.)

Making speech technology based applications more widespread has several consequences for demands on ASR systems regarding speaking styles:

- When ASR systems are used in new domains, we need new and larger vocabularies and grammars.

- When more natural language dialogues are used, the ASR must handle

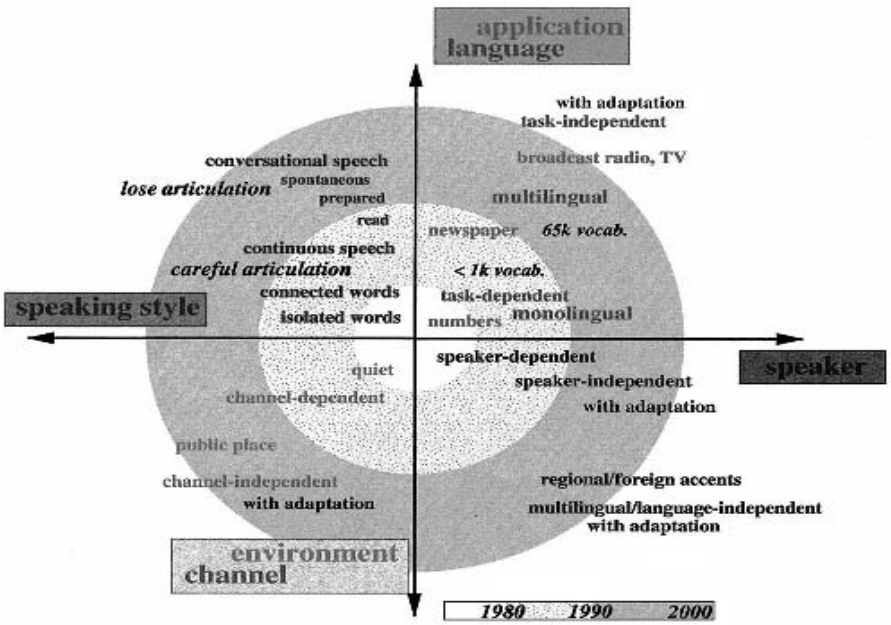

Figure 3.1: The ASR Big Bang, after [1] (page 9).

more conversational speech.

- Non-trained (non-expert) users may increase the use of dialects and accented speech in speech technology applications.
- A more international community increases the portion of non-native speakers in the general public and therefore in the "pool" of potential users of speech technology applications.

One example of the last issue is name-dialling applications that are gaining more and more popularity. For most larger companies the employees come from different countries. They will not always know the pronunciation of the native names and in addition they introduce non-native names that natives are unfamiliar with.

3.1 Dialects and accented speech (including non-native speech)

In [119] Van Compernolle has given a review of the problems when recognizing speech from non-natives as well as accents and dialects. This section is mainly

based on Van Compernolle's article, but is also inspired by Mayfield Tomokiyo, [81].

Accents will only show minor differences at the phonemic (abstract) level. Most of the variation will be (small) phonetic variations, like changes in vowel realizations. Accented speech may therefore be described as a shift of classes maintaining the separability between classes. It is thus an information preserving transformation. Usually accent variation will be similar for rather large groups of speakers, making it possible to train speaker independent, but accent dependent models in the same way as making different acoustic models for different languages. Experiments with the number of different models for Dutch and Flemish are referred in [119]. The conclusion was that too refined regional clustering is difficult because of lack of data, while three to four accent models were optimal. The main accent variation can be interpreted as allophonic, and acoustic model adaptation should therefore be a good way to take care of the variation. Experiments on dialects [20] and accented speech [53] have confirmed that acoustic model adaptation will handle some of the mismatch. The results are still not as good as for non-accented speech.

For non-native speech one of the big problems is that non-natives often project sounds onto a subspace of the native sounds [119]. The reasons are both that some native sounds can be difficult to pronounce, and that some sounds are hard to distinguish for a non-native. One example of the latter is the different feature dimensions for vowels for the different European languages. Quality differences may be difficult to perceive for a non-native whose mother tongue does not discriminate with respect to this feature. Non-native speech may still not be too hard to understand for a native as both vocabulary selection and grammar often will be simpler. For the ASR system this potential advantage becomes a disadvantage since it will result in a mismatch to the language model trained on native speech. The progress in acoustic modelling for native speech is based on more detailed modelling using more data which creates sharper distributions. The performance gain by using e.g. context-dependent modelling is often not present for non-natives as the more specified models have less tolerance for the non-native variation. Speaker adaptation will improve the modelling but even after adaptation the recognizer usually performs significantly worse on non-native speech than the native speech. In addition to the loss of information due to the lower dimension of the sound space, words unknown to the non-native may cause "random" pronunciation errors in speaker adaptation systems. [107] reported approximately the same gain for unsupervised as for supervised adaptation for non-native speech, which may be due to this effect. (The supervised adaptation contained many out-of-vocabulary words that also may explain the result.)

Mayfield Tomokiyo reported in [81] an extensive study on Chinese and Japanese speaking English. As expected the speaking rate was lower for the non-natives, mostly due to more frequent pauses. Coarticulation effects across word boundaries as modelled by a native ASR system, may therefore not be suitable for non-natives. For the lexical distribution she found that the native speakers used contracted forms more often. The non-native speech could not be described as more "restricted" as is often the case. Although the grammar was poor, unconventional word choices and unique ways of asking the same question can explain why the vocabulary growth was higher for non-natives in tourist information queries. As expected the perplexity was lower for non-native speech. Disfluencies were more frequent in non-native speech than native speech, but the Chinese speakers had substantial fewer disfluencies than the Japanese.

3.2 Spontaneous speech

The number of studies on spontaneous speech aiming to improve recognition of conversational speech has increased over the last years. One example is the recent interest in Broadcast News encouraged by the NIST evaluations [85]. Switchboard is an US English database especially created for research on robust ASR [40]. It was collected by Texas Instruments in the early nineties and sponsored by DARPA (Defence Advanced Research Projects Agency). Switchboard contains telephone conversations with the usual hesitations, restarts, etc. found in spontaneous speech. This makes Switchboard suitable for studies and experiments on spontaneous speech, see [44] and [58].

Spontaneous speech of this kind is difficult for the current state-of-the-art recognizers. An indication of the less structured speech encountered is that we usually talk about *utterances* instead of *sentences* for spoken language compared with written language. When speaking spontaneously people give even less attention to structure. It might be argued that "Switchboard"-type of human-human dialogue is too difficult, but surely this database contains features of spoken language that a recognizer should be able to deal with to make ASR systems really user-friendly. Many other available databases consist of read speech that has too little variation compared with real life for an ASR service. The ASR performance in lab versions is always better than in "real life", and we believe that some of this deterioration is caused by pronunciation variation.

In [121] Weintraub et al. investigated the effects of spontaneous speech on speech recognition performance. A three-way test using conversational speech, "acted" conversational speech, and read conversational speech gave

higher word error rates the more casual the speaking style. The results showed that the speaking style is a dominant factor in determining the performance. Apparently the acoustic realizations of word sequences were not modelled satisfactory by current ASR systems.

3.3 Language modelling and pronunciation variation

In pronunciation modelling the influence from the language model is mostly considered using *pronunciation probabilities*. A language model consists of the word probabilities, usually given some word history. When adding more pronunciations for each word, we can make the word probability dependent on the pronunciation probability. The ASR classifier equation (2.2) can then be decomposed to include the baseforms $\mathcal{B}$[1] for the word W:

$$\begin{aligned} p(\mathbf{O}|W)P(W) &= \sum_{B \in \mathcal{B}} p(\mathbf{O}|B,W)P(B|W)P(W) \\ &\approx \max_{B \in \mathcal{B}} [p(\mathbf{O}|B,W)P(B|W)P(W)] \end{aligned} \tag{3.1}$$

The last line is the Viterbi approximation of using only the best baseform as we do when using only the best path in equation (2.10). The pronunciation probability $P(B|W)$ can be defined as a part of the language model and we get the language model probability $P(B|W)P(W)$.

Different speaking styles will give different pronunciation probabilities. As shown for "ordinary" (considering words only) language modelling, the statistically based methods are most successful and therefore most popular in order to model pronunciation probabilities in the language model. When reliable estimates of the pronunciation probabilities are available, incorporating them has been shown to give improvements, see e.g. [66] and [129]. The importance of pronunciation probabilities increased when the number of pronunciations per word was high. In section 5.4.5 we give a short description of techniques to estimate pronunciation probabilities for pronunciation variant pruning.

In a review of large vocabulary speech recognition by Young [130], there is a section on spontaneous speech, focusing on how to modify the language models used in current ASR. The language modelling for spontaneous speech is in an early stage according to [130], referring to one of the oldest studies on spontaneous speech for ASR by O'Shaughnessy [86]. The big problem of making a spontaneous speech language model is that transcribed spontaneous

[1] A baseform is the entry in the ASR lexicon for a pronunciation.

speech in sufficient quantities is not easy to obtain. Most work on language modelling for spontaneous speech has therefore been concentrated on modelling of disfluencies.

3.3.1 Spontaneous speech characteristics in human-human versus human-machine dialogues

Studies on the differences between human-human and human-machine dialogues are important when making language models for ASR systems. Much of the theory on language modelling is based on written or spoken language between humans. To be able to use this knowledge we must understand the differences in human behaviour when speaking to a machine instead of a human.

Shriberg [103] has studied disfluencies in the human-human dialogues of Switchboard compared with other human-human dialogues (the database AMEX) and human-machine dialogues (the database ATIS). Both ATIS and AMEX belong to the air travel domain, which is more restricted than Switchboard. A more elaborate analysis of disfluencies in ATIS was performed by Nakatani and Hirschberg [84]. Shriberg grouped the disfluencies in "filled pauses" (e.g. "uhmm"), repetition, substitution, insertion, deletion and speech error. The human-human dialogues showed a lot more filled pauses, repetitions and deletions, probably because these disfluencies have a turn-taking function between humans. The probability of a sentence being fluent was found to be decaying roughly exponentially as a function of sentence length. The average number of words affected by each disfluency was surprisingly low (1-2 words) and constant over the corpora. Shriberg inferred from these results that there is a universal constraint regarding how many words that can be produced or understood during disfluent speech.

3.3.2 Language modelling and disfluencies

In [87] Oviatt found a correlation between disfluencies and both utterance length and structure in the task. The data were collected during empirical studies resulting in a total of forty-four subjects. The author concluded that the user-interface should be designed to elicit structure and short sentences to give less recognition errors. The users preferred a more structured form-filling type of dialogue over the more user-driven "unstructured" type.

Verbmobil is a German project for automatic translation between German, English, Japanese and Spanish. As a part of this project Schultz and Rogina [100] reported experiments on different "noises", e.g. breathing and paper rustle during spontaneous speech. Usually, such noises are deleted before

training the language model, but Schultz and Rogina showed that the perplexity decreased when the noises were treated as ordinary words. This means that some words are more likely to occur after such noises than others.

Shriberg and Stolcke have done work on statistical language modelling for spontaneous speech with the Switchboard database. They found that the usual scheme of removing disfluencies prior to language modelling gave lower perplexity, but no improvement in recognition rate [112]. The lack of recognition improvement was due to the very low baseline performance causing too few correct words to make the language modelling work properly. Removal of deletions and repetitions had the largest effect to lower the perplexity. This is therefore not a contradiction to the results in [100] that were concerned with noise. Shriberg and Stolcke warned against totally "cleaning up" the transcription, because in another study [104], they found a connection between hesitations and word predictability.

Removing the filled pauses gave no reduction in perplexity [112]. The authors explained this as caused by the acoustic segmentation of the speech at turns and pauses. Filled pauses often mark linguistic boundaries and these may come in the middle of the acoustic segments. If the filled pauses were present they might help to reveal the linguistic boundaries. This was investigated further by Shriberg and Stolcke in yet another study [111]. The ASR system gave too unreliable transcriptions so hand-labelled material was used instead. A trained bigram and trigram model on this hand-labelled transcription found about 70% of the linguistic boundaries. When including turn-taking information and part-of-speech labels, results were improved further. To the authors' surprise, using only part-of-speech labels and no word identity gave worse results than only word identity.

Another continuation of the studies on human-human dialogues was done by Siu and Ostendorf [108], who did language modelling using disfluencies as speech markers. Treating these speech markers like special words in the N-gram modelling gave reduced perplexity.

Chapter 4

Acoustic model adaptation

The acoustic models used in ASR systems today are capable of handling a lot of variation, as long as the variation is present in the training data. The most popular models are the statistically based HMMs, described in section 2.2.2. Using a mixture of Gaussians (or another suitable distribution) for the observation probability we can model both intra- and inter-speaker variation as well as environmental variation. With context-dependent models we can model allophonic pronunciation variation and many coarticulation effects. For pronunciation modelling purposes it is important to understand the capabilities of the acoustic models to take care of the variation.

The problem is the "curse of dimension" [74]. Models that can handle more variation require more parameters and training data. One solution is acoustic model adaptation. Usually this is performed on a speaker-by-speaker basis, but groups of speakers or specific environments can also be the objects of adaptation.

Speaker specific ASR with adapted acoustic models will encounter less variation in pronunciation and may need fewer variants. We should therefore be able to gain extra performance by speaker dependent lexical adaptation, i.e. chosing the optimal baseforms given the speaker dependent acoustic models. The performance of the adaptation depends on the quality of the transcriptions (given or obtained automatically). Lexicon adaptation *before* the acoustic model adaptation may therefore also be beneficial.

The description given in this chapter is mainly taken from two recent reviews on adaptation for ASR: Lee and Huo in [74] and Woodland in [125]. As in these two articles we only consider adaptation of HMM based systems.

There are several ways to group the various acoustic model adaptation techniques. [125] defines three families:

1. MAP adaptation (direct adaptation)

2. Linear transformation (indirect adaptation)

3. Speaker clustering/speaker space

Another division can be made based on how the adaptation is performed. If we have a transcription for the adaptation data, we call this *supervised* adaptation. If no transcription is known, usually the recognizer's output (often with some confidence checking) is used and we have an *unsupervised* adaptation. A further division can be made between *static* and *dynamic* adaptation. In *static* (or block) adaptation all the adaptation data available is used at once. In *dynamic* (or incremental) adaptation, we feed the system one utterance at the time and we have an intermediate adapted system available at any time.

Usually only the means of the (Gaussian) observation densities are adapted. Some experiments also report adaptation of the variance of the observation densities.

4.1 Linear transformation (MLLR)

Transform based adaptation is also called *indirect* adaptation. The most popular linear transformation is the maximum likelihood linear regression (MLLR) adaptation method introduced in [79]. The parameters (e.g. the vector of means $\boldsymbol{\mu}$ of the observation densities) are updated according to an affine transform:

$$\widehat{\boldsymbol{\mu}}_{\mathrm{MLLR}} = \mathbf{A}\boldsymbol{\mu} + \mathbf{b} = \mathbf{W}\boldsymbol{\xi} \tag{4.1}$$

Where $\mathbf{W}$ is an $n \times (n+1)$ matrix and $\boldsymbol{\xi}$ is the extended mean vector

$$\boldsymbol{\xi}^T = [1 \ \mu_1 \ \mu_2 \ \ldots \ \mu_n] \tag{4.2}$$

The transformation matrix $\mathbf{W}$ is estimated using the adaptation data. The transformation is tied over a cluster of parameters. For adaptation of the mean of the observation densities the clusters are usually derived from a tree built on acoustical similarity.

For small amounts of data MLLR perform better than many other adaptation techniques, but as the amounts of data increase it is not guaranteed to reach the ML estimate. Over-training is a problem that can be overcome by careful selection of the number of transforms.

4.2 MAP adaptation

Maximum *a posteriori* (MAP) adaptation is the most popular form of *direct* adaptation of the acoustic models. It was first presented in [75] and later

expanded to cover Gaussian mixtures in [36] and [37].

To estimate the parameters of the recognizer the traditional maximum likelihood (ML) approach in equation (2.4) is substituted by a maximum a posteriori probability approach:

$$\widehat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\theta|\mathbf{O}) = \underset{\theta}{\operatorname{argmax}}\, \frac{p(\mathbf{O}|\theta)P_0(\theta)}{P(\mathbf{O})} \tag{4.3}$$

The observation $\mathbf{O}$ is fixed and $P(\mathbf{O})$ is independent of θ and we get the equation:

$$\widehat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\mathbf{O}|\theta)P_0(\theta) \tag{4.4}$$

The a priori distribution of the parameters $P_0(\theta)$ is called the parameter *prior*. MAP is a Bayesian approach to parameter estimation where we combine the information in the adaptation data with prior knowledge.

The three key issues in MAP estimation are [74]:

1. Definition of prior densities
2. Estimation of the prior density parameters, often called *hyperparameters*
3. MAP estimation solution

A convenient choice of prior distribution is *conjugate priors* [75] which gives the same distribution family of the prior density as the posterior distribution. The hyperparameters are estimated from the data giving an empirical Bayesian approach (for a true Bayesian approach the hyperparameters are known.) The choice of prior densities and the estimation of the hyperparameters are therefore strongly related.

One example to illustrate the combination of information from the data and the prior is the estimation of the mean μ of a single Gaussian (observation) density using conjugate priors [75]. The variance σ^2 of the Gaussian is assumed known and fixed. The prior distribution $P_0(\mu)$ is defined by its mean ν and variance τ^2 (which in this case are the hyperparameters). We then get the estimate for μ:

$$\widehat{\mu}_{\mathrm{MAP}} = \frac{n\tau^2}{\sigma^2 + n\tau^2}\,\bar{x} + \frac{\sigma^2}{\sigma^2 + n\tau^2}\,\nu \tag{4.5}$$

The sample mean is given by $\bar{x}$, and n is the total number of training samples. The influence from the data is given in the first term while the influence of the prior mean ν is given in the last term. We see that the contribution from the two terms is controlled by the number of samples n and the prior variance τ^2.

The MAP approach is well defined mathematically and converges to the ML estimate. In the original formulation only observed parameters in the adaptation data are updated and the performance gain for small amounts of data is therefore limited.

4.3 Other methods

An extension of MAP to update unobserved parameters is the structural MAP (SMAP) approach proposed in [101]. We want to adapt as small clusters of parameters as possible dependent on the amount of adaptation data. The HMM parameters (Gaussian densities) are therefore organized in a tree structure. For each cluster the Gaussian components are summarized in one Gaussian density that can be updated using MAP adaptation. We then have adaptation information for a cluster of Gaussian components. If we have sufficient adaptation data for all parameters, SMAP will be the same as ordinary MAP. This is generally not the case and we use hierarchical priors to control the adaptation of the parameters where we have insufficient data.

According to [125] a recent trend is to combine the strengths of different methods. The most popular adaptation methods MLLR and MAP are best for smaller and larger amounts of data, respectively. A combination trying to get the best of both methods is therefore desirable. A promising approach is the SMAPLR adaptation presented in [107]. This is a combination of MAPLR (which uses MAP to update the affine transforms in MLLR) and SMAP, i.e. using hierarchical priors.

Speaker clustering/speaker space

Traditionally ASR systems have been made using gender specific models, this can be expanded to more speaker types. Two recent techniques are Eigenvoices and cluster adaptive training (CAT). More about these and other techniques including references can be found in [125].

Speaker normalization

Cepstral mean normalization (CMN) has been a standard feature normalization technique for many years. The main motivation is to remove the effect of the channel but it will also give a speaker normalization effect. More recently speaker normalization using vocal tract length normalization (VTLN) has been proposed in [77] and [78]. The frequency axis is re-scaled using a warping factor that is derived for each speaker. The warping factor is estimated using the mismatch between the input utterance and the speech recognition models.

Histograms show that the average warping factor is higher (corresponding to frequency expansion) for males than females.

4.4 Acoustic models and pronunciation variation

4.4.1 Speaker adaptation of acoustic models

Mayfield Tomokiyo reported in [80] experiments on Japanese reading English news. All transcriptions used in rule derivation were made using speaker adapted models. Pronunciation variants were obtained by an unrestricted phone recognition pass and later filtered using both knowledge and data-driven measures. Knowledge-based variants were also added. During testing a word lattice was made and the pronunciation variants were added to the lattice before rescoring. As the lexical modelling was performed on pre-made lattices, the joined effect of acoustic model and lexicon adaptation can not be derived.

Another experiment using speaker adaptation before pronunciation modelling was reported by Goronzy et al. in [41]. The task was native English speakers speaking German. For each of the 44 speakers in the adaptation set an average of 180 utterances was available. Unsupervised speaker adaptation using MLLR as well as speaker dependent pronunciation rules showed an improvement. Speaker independent pronunciation rules gave a deterioration compared with the baseline due to added confusability. The combination of speaker dependent pronunciation rules and MLLR showed the best performance on average. For five out of eight speakers the combination of pronunciation modelling and MLLR gave better results than MLLR alone.

4.4.2 Accent and dialect adaptation of acoustic models

Accent adaptation using a combination of lexical and acoustic model adaptation was investigated by Humphries and Woodland in [53] (also referred in chapter 5). The authors investigated adaptation of British English to US English using relatively small amounts of data (500 utterances). Both data-driven rule based pronunciation modelling and speaker independent MLLR was performed. The results showed improvement both using lexicon adaptation and acoustic model adaptation alone as well as an increased improvement when using a combination by applying the adapted lexicon prior to the MLLR adaptation.

Adaptation of the acoustic models from one dialect to another was explored in a study by Diakoloukas et al. [20]. The database used for the experiments contains Swedish speech. The original models were trained on speech from the Stockholm area, while speech from Skåne (an area in the south of Sweden)

was the target dialect. The approach was to use small amounts of Skåne data and speaker adaptation techniques published earlier by one of the authors [21]. The number of speakers seemed to have less effect than the (total) number of sentences. The Skåne adapted models gave a much lower error rate than the original Stockholm models using only 200 adaptation sentences. Increasing the number of adaptation sentences to more than 500 gave only small improvements. Adaptation performed better than training for small amounts of data, while for more than 1000 sentences training performed better.

4.4.3 Choice of speech recognition units

Using other units than phones as the acoustic model inventory has been an interest for a long time and has been investigated for example in [116] and [9].

Holter and Svendsen did a study described in [50] on optimal acoustically based units (i.e. shorter than phones). The motivation was that the recognition systems are neither truly phonetically nor acoustically consistent and that a joint optimisation of both units and pronunciation would be wise. The main problem with acoustically based units is that there is no one-to-one correspondence to any linguistic unit, making it difficult to find the baseforms to put in the lexicon. The resulting units should be robust to both inter- and intra-speaker variations. The algorithm for finding the units was therefore constrained to assign similar label-sequences to the same utterance, independent of speaker. This also solved the problem of finding the baseform. The vocabulary used for the experiments consisted of 20 US English words (digits and computer-related words). Recognition using the acoustically based units gave the same performance compared with whole-word models and phone models. The same effect was also observed for a larger vocabulary [49].

Another more recent approach for automatically finding an optimal phone set, used the phone based HMMs as a starting point. Monte Carlo simulations were used to compute the distance between phones by Singh et al. in [106]. An earlier work tried to generate the phone set based on the maximum likelihood of word segmentation [105]. In the same line Printz and Olsen presented a method of computing acoustic perplexity directly from the HMMs in [88]. The acoustic perplexity was used to compute a synthetic acoustic word error rate that measures the capabilities of the acoustic channel giving a better way to evaluate language models.

A compromise between word and phone models is to use syllables as the unit, giving longer contexts than phones, but fewer models than words. The use of syllables as the speech recognition unit can also be advantageous from a linguistic point of view [43]. The choice of recognition unit can also be language dependent. Chinese (Mandarin) has 1200 (tone-dependent) syllables

and Japanese only 50 [52]. For these and related languages the syllable is therefore a very interesting unit. Syllables are not so popular recognition units for English since it is believed that the syllabic structure is too complex. The many different syllable structures in English give a large number of different syllables; e.g. for the Switchboard corpus 47406 different syllables were found [43]. But all these syllables are not equally frequent: in the lexicon 75% of the syllable types were of the form consonant-vowel, vowel-consonant, vowel, or consonant-vowel-consonant. Investigating the transcriptions this percentage was even higher: 84% of the syllable types in the Switchboard corpus had one of these "simple" structures.

The use of syllables, and more specific syllable onsets, in speech recognition was explored by Wu et al. [128]. A number corpus collected over telephone lines with a vocabulary of 32 words was chosen for the recognition experiments. An automatic syllabification algorithm was used to find the syllables, 30 for a single-pronunciation lexicon and 118 for a multi-pronunciation lexicon. The multi-pronunciation lexicon was derived by data-driven methods based on a (hand-made) phonetic transcription. This multi-pronunciation lexicon improved the performance by itself. Syllabic models were derived from the basic phone units using a syllable-grammar. During the combination of syllables to words, the recognizer was restricted to linking the start of the syllabic models to marked syllabic onsets. In this way the temporal properties of the syllables were forced upon the recognizer, although the basic units were not strictly syllabic but syllables made up of phones. First the syllabic onsets were found artificially from the text, this increased the recognition greatly. Secondly these onsets were found by training a neural network, the increase in the recognition performance was now "not quite reaching statistical significance". Wu et al. still found the results promising and anticipated possible improvements of the segmentation algorithm responsible for finding the onsets. A more recent syllable approach was reported in [35] with promising experiments on Switchboard and Alphadigit tasks.

4.4.4 Dynamic HMMs

An approach on the border between acoustic and lexical modelling of pronunciation variation was presented by Hain and Woodland in [45] and [46]. A new layer of statistics was introduced for the link between acoustic models and baseforms. The stochastic relation between the HMMs and the lexical units was called Hidden Model Sequence (HMS). In the first approach the length of the sequence was restricted by the original baseform, later experiments investigated the possibility of deletions and insertions. The HMS-HMMs were tested on RM and Switchboard and gave increased performance, the addition

of deletions and insertions gave only small additional improvements. One of the problems with the insertion models was that they became too broad, further work is needed to make more compact distributions.

Chapter 5

Lexicon adaptation

Lexicon modification is the most popular way of modelling phonological pronunciation variation. Segmental variation, such as allophonic variation, can be handled by the acoustic models using training or adaptation on the target speech. Other types of variation may be better handled at the lexical level, e.g. insertions, deletions, and variation that is present for a group of speakers (e.g. dialects), or is typical for a speaking style. Lexical modelling accommodates longer contexts than acoustic modelling, permitting modelling of syllables and even entire words or phrases [62]. This chapter will give an overview of the current status of pronunciation variation modelling by lexicon adaptation. Our approach is given in chapter 6.

There are two main directions in finding pronunciation variations, each involving different problems:

1. **Knowledge based methods** where we try to find the best pronunciation rules by applying phonetic and linguistic knowledge. The main problem occurs if the knowledge does not cover the variation we want to model. We may then have too many or too few variations and we may not know how frequent they are.

2. **Data-driven methods** where we use databases of real speech to find the variations present. The problem is that the variations based on a given database may give a result too specific for that database. One of the advantages is that we may compute probabilities for the variants, as opposed to the knowledge based methods.

For both these methods we can distinguish between *direct* and *indirect* modelling. The pronunciation variants can either be derived directly for each word or indirectly by deriving pronunciation rules and using these rules to generate new pronunciations. Data-driven direct modelling limits us to model

only words observed sufficiently many times in the adaptation set, whereas for indirect modelling (both for data-driven and knowledge based) care must be taken in the generalization from the observed variation. If a certain variation appears in very different contexts in the adaptation data compared with the test data the generalization may not be valid. An example of this is that variation observed in function words in the adaptation data is not necessarily a variation appropriate for content words in the test data even if the phone context is the same. One example of this is the function word "for" with the canonical baseform `[f ao r]` and the alternative baseform `[f er]`. This transformation is not equally probable for the noun "forest" with the canonical baseform `[f ao r ah s t]`.

The main reasons to use indirect modelling in a data-driven approach are:

- The vocabulary of the data used for rule derivation can be different from that of the test data. Rules help us generalize the variation seen in the adaptation data to words not present ("unseen words").
- Rules depend on smaller segments than words and will occur more often, giving more reliable estimates.
- A possible extension to cross-word rules will be easier.

Some of the pronunciation variation will be present across word boundaries [38]. Using rules makes the extension to cross-word pronunciation modelling easier, although multi-words makes it possible to model cross-word effects also when dealing directly with baseform variants [32]. *Multi-words* are new lexical items formed of several words, e.g. "going to" can be treated as one word to account for the baseform variant `[g aa n ax]`.

Decision tree modelling, also called CART modelling (Classification and regression trees) is a popular method for deriving pronunciation rules. It is described in some of the earliest pronunciation modelling approaches e.g. [10] and [95]. Usually one tree is built for each phone. The number of rules is controlled by limiting the number of mappings in each leaf and the leaf probabilities can be used as rule probabilities.

There has been a migration from knowledge based methods to data-driven methods. In [31] and [33] it was shown that general purpose lexica do not model spontaneous speech sufficiently well. Only 33% of the pronunciations found in the hand-labelled part of Switchboard were present in the Pronlex dictionary. The non-canonical pronunciations showed an 11% increase in word error rate over canonically pronounced words. Another observation was that frequent segments showed more variation. Data-driven methods will model frequently occurring segments better. This might be an advantage, as frequent

words will have a larger influence on the WER. Besides, ASR is based on statistics, and the differences and similarities perceived by humans might not be the most useful for ASR. A combination is thus usual, using some kind of data-driven method to verify the rules and find probabilities for them.

Most experiments report the need for several iterations. When new pronunciations are found, retraining of the acoustic models using the new pronunciations, by retranscription of the training set, will usually give increased performance. Another iteration using the retrained models for a new transcription to derive yet another set of pronunciations may give further increase in performance. Modelling pronunciation variation in the lexicon should reduce the need for modelling variation in the acoustic models, and a retraining after lexicon modification should give better performance because the models will be more focused.

In the presentations of previous work we will focus on the characteristics of the different methods used for modelling pronunciation variation. Improvements in error rate compared with a baseline system is the usual way of assessing the performance of a method. Comparing the performances of the different methods is difficult as different complexity, different languages (although mostly English), and different corpora are used. The amount of training data should also be taken into account when comparing. This makes it difficult to do a fair comparison, and we have therefore chosen to cite no error rate numbers.

The effect of workshops on the subject is apparent. Much of the work reported here has connections to workshops dedicated to pronunciation modelling. The focus on one topic helps to get people interested and is an arena to exchange new ideas and advance cooperation between different research groups. Many of the experiments cited here were presented at the workshop "Modeling Pronunciation Variation for ASR" organized in Rolduc (the Netherlands) in 1998 or the workshop "Adaptation Methods for Speech Recognition" organized in Sophia-Antipolis (France) in 2001. The two reviews from these workshops have been useful for this overview, [113] and [114].

5.1 Knowledge based pronunciation modelling

The use of linguistically based transformation rules to generate decision trees for alternate pronunciations was described by Finke and Waibel in [26]. Forced alignment was used to choose among the rules in the training corpus, which was the Switchboard database (using a speaker adapted recognizer). These data-verified rules were modelled in a decision tree, taking into account the context dependency of phonetic neighbours, word type, speaking rate, average

word/phone duration, vowel stress, pitch, and computed probabilities. A rule probability was estimated from the relative frequency of the use of each rule. The resulting transformation rules were interpreted as speaking mode dependent. The lexicon was expanded using baseforms found by forced alignment, i.e. direct modelling, but with indirect modelling as an intermediate step. The variants found using the derived rules did not increase the performance as much as when selecting variants from the baseline dictionary. The authors interpreted this as due to added confusability. Baseform weighting using the speaking mode dependent decision trees increased the performance. The starting point for this experiment was knowledge based, but data were used for verification.

Using hand-labelled data is also a kind of knowledge based method, but if the pronunciations found are used in retranscriptions, the categorizing is less clear cut. One example of this is experiments performed during the Johns Hopkins summer workshop in 1997 described by Byrne et al. in [13] and Riley et al. in [94]. The results show that pronunciation modelling techniques using automatically labelled data performed better than hand-labelled. The experiments first used indirect pronunciation modelling using the rules to generate variants for unseen words. Decision trees were used to model the phonological rules seen in hand-labelled speech material, i.e. which phones can be neighbours dependent on lexical stress and distance from word boundary. From these decision trees a small network of possible alternate transcription was made for each word. Using all of these alternative pronunciations for recognition decreased the performance. This is the same effect as was shown in [26] and shows the need for some way to choose which pronunciations to accept in the lexicon. The alternative pronunciations from the decision trees were then used in a forced alignment to retranscribe the corpus. Using new decision trees based on this automatic transcription the performance increased. The authors' interpretation was that there was a mismatch between transcription by human perception and machine perception. Another reason may be that the hand-labelled material was much smaller than the automatically transcribed since the process of hand-labelling is time consuming and expensive.

Direct modelling with hand-labelled data as boot-strap was also investigated in the experiments of the Johns Hopkins summer workshop in 1997 using an "explicit dictionary expansion", [94] and [13]. Pronunciations found sufficiently often in the hand-labelled or the automatically transcribed corpus were put in the lexicon with weights based on relative frequency. This means fewer variations for the recognizer, and the performance increased, as expected. Multi-words were used to model coarticulation effects between some of the words. The conclusion of the experiments was that cross-word context is

not necessary except for some words. Using the initial decision tree rules to retranscribe the training set and train new acoustic models increased the performance. Using these new models for a new retranscription making new decision trees and new explicit lexicon expansions, improved the performance even more. These experiments also showed the need for care when deriving the weights for the alternate pronunciations (i.e. pronunciation probabilities).

Five known pronunciation rules for Dutch were investigated by Kessens et al. in [66] and Wester et al. in [123] (4 deletion rules and 1 insertion rule). Improvements were shown by incorporating them in known contexts. Modifying the acoustic models by retranscribing the training data using the variants, gave increased improvement. Language model modification was done by incorporating pronunciation probabilities (computed from forced alignment of training data) and gave further improvement. Cross-word rules were investigated by both including "border" versions of pronunciations and multi-words. In both cases this was limited to frequently occurring variations. A data-driven approach was compared with this knowledge based approach by Kessens et al. in [65]. Deletion rules were found by allowing deletions in an alternative transcription, in order to let the acoustic models decide where to delete phones. The knowledge based approach and the data-driven approach gave about the same performance, but the data-driven approach resulted in a smaller lexicon. As the data-driven rule derivation was controlled by frequency counts, the most frequently occurring variations (the most important ones) were favoured. There was a 96% overlap in transcriptions by the two approaches. The data-driven rule context was phone identity, whereas the knowledge rule context contained broader groups of phones. The knowledge based rules will therefore be applied more often, even if the same transformation (in this case deletion) is described.

5.2 Direct data-driven pronunciation modelling

One of the major challenges in pronunciation modelling is to decide which baseforms to include in the lexicon to get the best ASR performance. Most ASR lexica in use today are based on linguistic knowledge only and are not optimized with respect to ASR performance. Data-driven pronunciation modelling focuses on finding the "best" baseforms given an objective criterion.

An overview of the recognition system with adapted lexicon based on data-driven methods is shown in figure 5.1. The new lexicon is affected by both the adaptation data and the acoustic models, and for most approaches also the knowledge based reference lexicon. For a consistent approach the language model should also be considered. This is shown by a dashed arrow. The

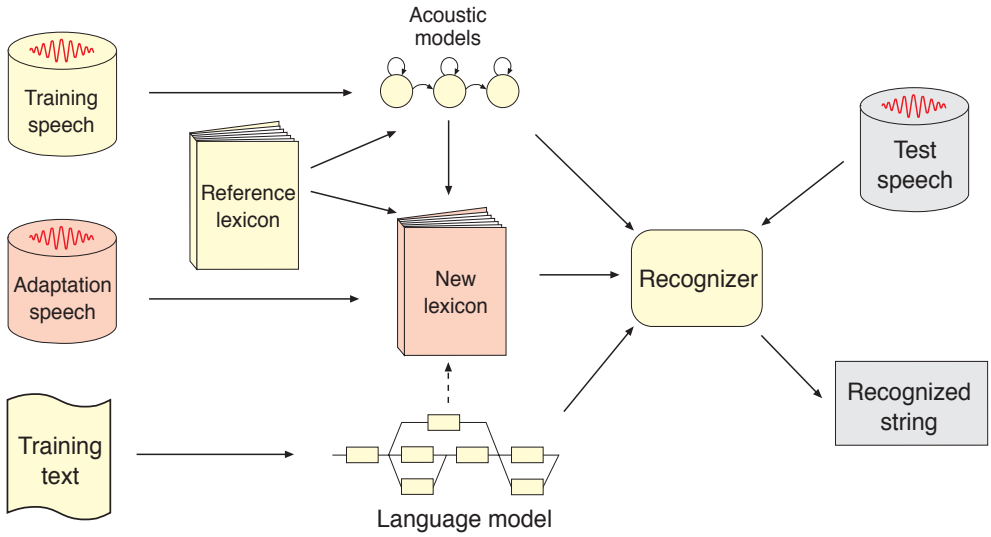

Figure 5.1: Recognition system with adapted lexicon.

acoustic models may also be adapted by the new data, possibly using the new lexicon. The new lexicon may also influence the language model. For an optimal system the influence between all parts of the system should be considered giving a joint optimization.

Pronunciation variation modelling can be described in two steps:

1. Find pronunciation variants

2. Assess the variants and modify the lexicon

A truly data-driven method was reported by Holter and Svendsen in [51], using some of the methods derived for finding optimal baseforms presented in [50]. For the experiments an US English database with a 991 word vocabulary was used. No rules or hand-labelled data were used, only a baseline recognizer. This recognizer was used to make an N-best list of pronunciations without any prior knowledge of the vocabulary other than the number of words and the boundaries of each word, which is usually present in an orthographic transcription. A subset of these pronunciations was chosen through a maximum likelihood algorithm doing joint optimization for all the utterances of each word. A clustering procedure chose which variants to add in the lexicon to ensure distinct baseforms, still using the maximum likelihood metric. The appeal of this idea is that it uses objective criteria for the optimization of all parts of the recognizer.

Another approach to finding variants using data-driven methods was presented by Fosler-Lussier in [29] and [32]. First an alternative transcription based on a bigram phone grammar was used to derive rules using decision trees. Then the training set was retranscribed using these rules to get a "smoothed transcription" and the pronunciation variants that occurred frequently enough were added to the lexicon. An added feature in this approach is the use of "dynamic" lexica, including word frequency, word trigram probability, word length, and speaking rate measures in the pronunciation modelling.

A similar approach was described for the Verbmobil project by Wolff et al. in [124]. A phone recognizer generated alternative hypotheses that were aligned with a transcription based on the canonical lexicon entries. Baseforms occurring 8 times or more were included in the lexicon and gave an increased performance. A measure of confusability called "consolidation" showed that words occurring 15 times or more had reached a "stable" set of pronunciations. This threshold excluded 85% of the lexicon adaptation material. The algorithm was therefore expanded to incorporate generalization by using frequent sequences instead of words only.

5.3 Indirect data-driven pronunciation modelling

Generating pronunciation variants using rules that are automatically derived from data, is frequently used. These rules should ideally capture the difference between the reference pronunciation of a word and the actual pronunciation used by the speakers. This approach is similar to the first step in the variant generation of [32] and [124], as well as the approach based on hand-labelled transcriptions in [94].

Humphries and Woodland did experiments on British English accent modelling in [55]. This is an example of indirect modelling, deriving pronunciation rules from data. No hand-labelled transcription was used, so this is an entirely data-driven approach. Alternate transcriptions were found by allowing all vowels to be substituted and then using a forced alignment. Vowel transformations are an important difference between British English accents. The three best transcriptions were used to derive rules using context dependent decision trees including leaf probabilities. A new pronunciation dictionary was then made for the new accent. Typically an average of 4 pronunciations per word were found to be effective. The test was performed on a 2000 word vocabulary, and adding pronunciations increased the performance.

Humphries and Woodland have also done experiments on British versus US English in [53] and [54]. The recognizer was here used to perform a free (and erroneous) transcription. An acoustic confidence measure was used to filter the

transcribed data before rule derivation. This transcription was aligned with the canonical baseforms in the British English lexicon. A list of possible phone transformations (substitutions, insertions and deletions) was generated and incorporated in a decision tree. Examination revealed interesting correlations with linguistic analysis of the differences between British and US English, e.g. transformation of `[t]` to `[d]` in certain contexts. The British-trained recognizer was tested on US English speech using US-adapted pronunciations, and this gave an increased performance compared with using British pronunciations [54]. There was no difference between using British pronunciations and adapted US pronunciations when training new acoustic models on US English speech [53]. The performance was worse but comparable to US models trained on "real" US pronunciations. In the case of sufficient training material, the extra phonological information was of less value. The authors suggested that in this case the pronunciation variation was taken care of by the acoustic models.

An example of the migration from knowledge based to data-driven modelling, is the work of Cremelie et al. Their first work was based on hand-labelled data [17]. Later work in [18], and [129] was based on automatically derived "expert" transcriptions where a constrained speech recognizer was used. In the improved version [18] more linguistic constraints were put into the automatic transcription, a wider context was used, and negative rules were allowed. Negative rules means that variation is prohibited. In [129] further improvements gave a significant performance gain. Pronunciation rules were derived from alignment of the reference and "expert" transcriptions, and each acoustic segment was only allowed to count for one rule. Rule probabilities were found by frequency counts and the rules were ordered in a rule hierarchy which favoured more specific rules. The most important rules were the coarticulation rules between words, most other experiments do only consider intra-word variations. These coarticulation rules were incorporated in the language model (since cross-word rules cannot simply be added to the lexicon). For the experiments they used two databases with different vocabularies for both English and Flemish, i.e. four databases in total. Cross-checking between the two databases for the same language was done to investigate possible corpus-specific rules. The results showed improvement for both languages and about the same level of improvement for rules based on either automatically or hand-made transcription. The automatic transcription version was best in some tests, the same effect as observed by Byrne et al. [13]. According to the authors, the reason could be that the automatic transcription caught the peculiarities of the recognizer and therefore gave rules better suited for this particular recognizer.

5.4 Step-by-step data-driven rule derivation

Data-driven pronunciation rule derivation is the technique used in this dissertation and the standard steps are therefore described in more detail referring to previous work. The same steps will be used when describing our own work in section 6.3. We have chosen some of the approaches referred in sections 5.1–5.3 to give examples of previous work in data-driven pronunciation rule modelling. The approaches chosen to illustrate rule derivation are: Cremelie et al. [18], Fosler-Lussier et al. [32], Humphries et al. [54], and Riley et al. [94].

Pronunciation variant generation by using data-driven rule derivation can be described in five steps:

1. Automatically generate alternative transcriptions
2. Align the reference and alternative transcriptions
3. Derive rules from the alignment
4. Assess and prune the rules
5. Generate baseform variants from the rules, assess the variants, prune or assign weights, and modify the lexicon

As we can see the first step in direct pronunciation variant modelling is replaced by a 4-step rule derivation and a step 5 to generate variants from the rules. Step 4 is not trivial, rules may interact and one rule may change the context affecting another rule. The rule pruning will control the number of variants indirectly, but we need a step 5 to assess the variants. We may also add a step 6 performing retranscription of the lexicon adaptation material and iterate the process.

The 4 (3) first steps may be replaced by phonological rules if we have knowledge about the variants we want to describe. Step 5 (and 4) should be performed anyway to ensure that we control the performance of the recognizer in a consistent way.

5.4.1 Reference and alternative transcriptions

The first step in rule generation is finding an alternative transcription that can reveal the true pronunciations of the speakers. We can also use knowledge: if we have hand-labelled data, pronunciation rules can be derived from comparing this transcription with the reference. The alternative transcriptions may contain two types of errors. Either they can be too similar to the reference transcription and hide the differences really existing in the data, or they may

contain transcription errors. The reference transcription will also often be derived semi-automatically. Usually only a word transcription exists and if the reference lexicon contains several baseforms, the recognizer is used to choose baseform by forced alignment.

There are two dominant ways of finding the alternative transcriptions automatically [114]. A phone recognizer is used employing either:

- A phone loop (no grammar) [54], or
- A restricted phone grammar [18], [32]

Another way to generate alternative transcriptions is to bootstrap using hand-labelled data as in [94].

For the experiments in [54] a phone loop approach was used since a restricted phone grammar without the possibility of deletions and insertions performed worse. In [18] the alternative transcription was found by allowing each phone in the reference transcription to be deleted or substituted by a phone of the same class. Insertions were also allowed and probabilities were assigned to both deletions, insertions, and substitutions. In [32] a phone bigram was used. All these approaches use the recognizer, i.e. apply data-driven generation of the alternative transcription.

As an example, the two transcriptions for the utterance
"Paramount pictures expected eight ..." are:
Reference transcription (canonical baseforms):
`[p eh r ah m aw n t p ih k ch er z ih k s p eh k t ah d ey t]`
Alternative transcription (phone loop on spontaneous dictation):
`[p eh r m aa m p ih k ch er z ah k s p eh k t ih t ey iy t]`

5.4.2 Alignment

The usual approach when aligning the two transcriptions is to use dynamic programming. The difference between the methods lies in how the costs for the phone-to-phone mappings are assigned. For the experiments we refer to, three different kinds of costs were used:

- Uniform [54]
- Phonetic feature vector [94], [32]
- Four phone groups (and silence) [18]

The two last methods used phonetically based costs. In [94] and [32] each phone was classified according to a set of phonetic features. The cost for a

phone-to-phone mapping was derived by comparing the phonetic feature vectors for the two phones. In [18] this was simplified to classify the phones in four groups (vowels, sonorants, fricatives, and plosives). Both these methods rely on knowledge about phone similarity and the assumption that the probability of phone-to-phone mappings due to pronunciation variation will follow phone similarities. We can observe a correspondence between the chosen methods for transcription and alignment as no a priori assumptions are made neither in deriving the alternative transcription nor in performing the alignment in [54].

The example transcriptions for "Paramount pictures expected eight ..." can be aligned as follows:

```
[p eh r ah m aw n t p ih k ch er z ih k s p eh k t ah d ey    t]
[p eh r    m aa m   p ih k ch er z ah k s p eh k t ih t ey iy t]
```

5.4.3 Rule derivation

Rules representing the pronunciation variation can be extracted from the alignment of the two transcriptions. A usual approach in rule based pronunciation modelling is to let the rules express phone-to-phone mappings (allowing deletions and insertion). It has been shown [18], that deletion rules are frequent, especially in spontaneous speech [31]. The rules are usually defined as dependent on a specified context. The width of the context has to be decided as well as which other information to include. The most frequently used context is one phone neighbour to each side of the phone(s) affected by the rule. Other contexts that are shown to have effect include word frequency [32], lexical stress, and syllabic information [94]. Using more complex contexts demands either more data to estimate the rules properly or a generalization of the contexts. CART trees can be used to generalize the context of a rule in an automatic way. For cross-word rules the word boundary information must be included in the context.

From the example transcriptions for "Paramount pictures ..." we can derive several rules. Examples of word internal phone-to-phone-mappings with context given by phone identity are:
`[r ah m]` maps to `[r m]` (a deletion)
and
`[m aw n]` maps to `[m aa m]` (two substitutions)
In the last case a word-external rule may be more appropriate ($ marks the word border):
`[m aw n t $ p]` maps to `[m aa m $ p]`

To derive and cluster rules from the aligned transcriptions, both [54], [94] and [32] used decision trees. All three approaches started modelling only word

internal rules. To account for cross-word context, multi-words were added in [94] and [32].

In [18] both word internal and external rules were derived. The algorithm also included negative rules that prohibit variation to be applied. For the cross-word rules, the index of the rule was attached to the variant to ensure compatibility with the previous or next word affected by the rule.

5.4.4 Rule assessment and pruning

Most rule based pronunciation modelling techniques need some kind of pruning to control which of the alternative pronunciations that should be included in the lexicon. Rarely used pronunciations may introduce more errors than they correct. We can use a threshold based either on the rule probability or the pronunciation probabilities or both to control the number of new pronunciations to add.

In [54], [18], and [94] the rules were pruned using rule probability based thresholds. The rule probabilities were found from the decision trees (i.e. a kind of entropy measure) in [54] and [94], or frequency counts in [18].

5.4.5 Pronunciation variant generation, assessment and pruning

From the resulting set of selected rules new pronunciations are derived. Using several rules for each baseform will result in a huge number of new baseforms:

$$\#\text{ baseforms} = (\#\text{ rules for phone 1}) \cdot (\#\text{ rules for phone 2}) \cdots \tag{5.1}$$

The resulting pronunciation probability derived from the rule probabilities would be very low for most of the multi-rule derived baseforms. One way around this is to use estimated pronunciation probabilities as a threshold to limit the number of variants instead of, or in addition to, the rule probability threshold.

When we have little data to derive pronunciation from, it may also be necessary to merge the reference lexicon and the new baseforms [54]. In [32] merging was found beneficial even if a 100-hour adaptation set was used, because the pronunciations were modelled directly and the infrequently occurring words got too "noisy" pronunciations.

The pronunciation probabilities can be estimated directly by counting the different pronunciations chosen when retranscribing the adaptation data by forced alignment (restricted by the initial rules). This was described as "smoothed transcription" in [32] and "explicit" dictionary expansion in [94].

Estimating pronunciation probabilities is not trivial for unseen words. Rule probabilities make it possible to estimate the probability for unseen words and pronunciations, but care must be taken to not generalize too much. In [54] the pruning threshold for the pronunciation probabilities was derived from the rule probabilities that were given by the leaf probabilities of the decision tree. In [117] the problem of combining the rule probabilities to word probabilities was discussed. Pronunciation probabilities derived from the decision trees were compared with pronunciation probabilities estimated by frequency counts in [94]. These two ways of estimating the pronunciation probability did not give the same result.

5.4.6 Retranscription

The rule derivation can be done iteratively by retranscribing the adaptation data using a grammar restricted by the initial rules. Retranscription, realignment, and new rule derivation can be done iteratively until only small changes are observed in the resulting rule set. This will be similar to the "smoothed" transcription and "explicit" dictionary expansion referred to earlier, [32] and [94], where only variants seen sufficiently often in the training data were used.

Retranscription will give fewer transcription errors, but at the expense of restricting the alternative transcription to be more similar to the reference transcription. The limitation imposed on the transcription will ensure more instances of each rule. We will not get any new rules from this approach, but we may have more confidence in the rules we get.

5.5 Confusability reduction

Confusability reduction is closely linked to the pruning of rules and pronunciation variants. As shown the assessment of pronunciations is usually done by one or a combination of these methods:

- Assess pronunciation rules (the resulting variants will then be assessed indirectly)
- Assessing each pronunciation variant directly

The rule hierarchy used in [18] and [129] is an indirect way of reducing confusability by restricting confusable rules. The consistency and consolidation measures in [124] could also be used to control the confusability.

Modelling pronunciation variation by adding several baseforms for each word in the lexicon should be done with care. More variants, and probably more similar variants, will increase the lexical confusability and the error

rate. A method of balancing the "old" errors corrected and the "new" errors introduced should be included in the algorithm. Rule and variant assessment only considers the "goodness" of each pronunciation alone and how it performs on the word which the baseform belongs to. The pronunciations will interact and a more global approach to assess the total set of pronunciations also taking into account the effect on the other words in the vocabulary, should be beneficial. This calls for discriminative techniques incorporating misclassification measures.

Torre et al. explored in [118] measures of word confusability by first estimating a phone confusion matrix. The phone confusions were combined with word confusions that were used in vocabulary selection. If we have several synonyms, this algorithm can be used to choose the best pronunciations among synonyms. With basis in the same algorithms, automatic alternative transcription generation was also examined with promising preliminary tests. Phone confusion matrices were also used by Sloboda and Waibel in [109]. First the confusion matrix was used to smooth a phone bigram that was used to automatically find variants for frequently misrecognized words. For the variants found, homophones as well as variants that only differed in confusing phones were eliminated.

Discriminative model combination was used for pronunciation modelling experiments by Schramm and Beyerlein in [99]. An expression for the word error count including the pronunciation weights was derived. Minimizing this function with respect to these weights gave an iteration formula for updating the weights. The update function depends on the frequency of occurrence of the baseform in the true word as well as for competing words. The technique presented is similar to the one presented by Korkmazskiy and Juang in [67], where GPD adaptation of pronunciation weights were used.

A confusability measure based on substring matching was presented by Wester and Fosler-Lussier in [122]. All possible word pronunciations that matched substrings in the reference transcription were used to make a lattice of confusable words. The confusability metric was calculated by considering the number of words that corresponded to each phone. Baseforms with high confusion counts were then removed. This method was later extended to incorporate acoustic confusability by Fosler-Lussier et al. in [30].

The suggestions for confusability measures mentioned above have shown modest improvement in WER. The final criterion for selecting which pronunciation variations (or rules) to use in the recognition system should ideally be consistent with the optimization on the rest of the system, i.e. considering both acoustic models, the lexicon, and the language model.

Chapter 6

Acoustic log likelihood based pronunciation modelling

We believe that in addition to knowledge based methods we should incorporate data-driven methods, at least to assess the baseforms. The acoustic models used in a recognizer do not necessarily correspond to the abstract linguistic units used to describe the pronunciation of a language [12]. There may also be a mismatch between the speaker independent acoustic models and the speaker. For a consistent approach to automatic speech recognition all parts of the recognizer including the lexicon should be optimized using objective criteria.

The focus of this dissertation is lexicon adaptation, i.e. pronunciation modelling by changing the baseforms in the ASR lexicon. It is then important not to add baseforms for the variation that is already modelled by the acoustic models. We believe that this can be achieved using acoustic log likelihood as a pruning measure both for variants and rules. It is crucial that the acoustic models used in pronunciation modelling are the same as the ones used for the ASR system in operation.

Rule probability estimation from an automatic transcription is sometimes referred to as maximum likelihood pronunciation modelling. Since the frequency counts are obtained from a maximum likelihood transcription this is a fair use of the term, but in this dissertation we reserve the term "maximum likelihood" for optimization with respect to acoustic log likelihood.

This chapter starts with a formulation of the pronunciation modelling problem using decision theory. Then our approach to both direct and indirect pronunciation modelling is described using the same outline as in the description of previous work in chapter 5. The last section is devoted to the important issue of confusability reduction.

6.1 Decision theory applied to pronunciation modelling

Translating the decision theory elements to ASR, the classifier is the recognizer and the classes are the words, sentences, or meanings that we can extract from an utterance. The classifier/recognizer consists of acoustic models, the pronunciation lexicon, and the language model, see section 2.2. All three (high-level) parts of the recognizer may be optimized discriminatively. In this dissertation the pronunciation lexicon is the main issue and we therefore translate the general decision theory equations to incorporate the parts of the ASR system that depend on the pronunciation lexicon. The notation is the same as in the description of general decision theory in appendix D.

We use the following notation for the classification system:

- The classifier $C(\cdot)$
- A set of M different words (classes) $W_j \in \{W_1, W_2, \ldots, W_M\}$
- A set of baseforms $\{B_j^{\mathrm{id}}\}$ for each word where `id` shows the identity of the baseform.
- A set of N training samples $\mathbf{o}(n) \in \{\mathbf{o}(1), \mathbf{o}(2), \ldots, \mathbf{o}(N)\}$ where each $\mathbf{o}(n)$ is an array of feature vectors representing an acoustic segment belonging to a word [1].

When we need to distinguish between several baseforms B_j this will be indicated with a superscript. The baseform $B_j^{\max}$ will for instance be the pronunciation with highest log likelihood for word W_j.

When the index of a sample is not important, we omit the n and write $\mathbf{o}$. Indicating which word an acoustic segment belongs to will be shown using subscript [2]: observation $\mathbf{o}_j$ belongs to word W_j. The number of samples for one word W_j is N_j. N is the total number of training samples irrespective of word, i.e. $\sum_{j=1}^{M} N_j = N$. The set of all samples belonging to one word is then:

$$\{\mathbf{o}_j(n)\} = \{\mathbf{o}\ ;\ \mathbf{o} \in W_j\},\ n = 1, \ldots, N_j \tag{6.1}$$

In the following equations we also need an indicator function defined as:

$$1(\xi) = \begin{cases} 1, & \text{if } \xi \text{ is true} \\ 0, & \text{otherwise} \end{cases} \tag{6.2}$$

[1] The length of the array will depend on the length of the acoustic segment and will vary both within words and between words. For simplicity and readability we will omit the length.

[2] This is different from equation (2.2.2) where the subscript on $\mathbf{o}$ was used to show time dependency. Since we no longer keep track of the length we reuse the subscript.

Discriminant function

First of all we define a discriminant function g which measures how well an acoustic segment $\mathbf{o}$ matches a given baseform B_j and the recognizer's parameters θ. Discriminant functions are explained in more detail in appendix D. Here we use the log likelihood as discriminant function:

$$g'_j(\mathbf{o}|\theta) = \log p(\mathbf{o}|B_j, \theta) \tag{6.3}$$

For an ASR system the parameter set θ will correspond to the acoustic model parameters, such as HMM parameters in the observation densities and transition probabilities (and pronunciation and language probabilities). For simplicity and readability we will ignore the parameters θ in the discriminant notation in later equations. In the case of pronunciation modelling we want to optimize with respect to the baseform B_j and not θ. (Which is "usual" in discriminative training since the acoustic models then are the targets for optimization.) Normally we will use log likelihood per frame to be able to compare log likelihood for segments of different lengths.

For a true Bayesian approach we should use the a posteriori probability as the discriminant function:

$$P(B_j|\mathbf{o}) = \frac{p(\mathbf{o}, B_j)}{p(\mathbf{o})} = \frac{p(\mathbf{o}|B_j)P(B_j)}{p(\mathbf{o})} \tag{6.4}$$

We want to find an objective measure consistent with the error rate of the system. This measure can then be used in a minimization with respect to the baseforms we want to assess to bring the system towards a minimum error rate. The measure should be consistent with the recognizer and we have therefore chosen to use the the acoustic log likelihood. In both the training phase and recognition phase (classifier operation) log likelihood is used. Using log likelihood instead of the a posteriori probability means that we ignore the probability $P(B_j)$.

A misclassification measure for introducing a new baseform B_j^{alt} for another word W_j may then be expressed as:

$$\log p(\mathbf{o}_i|B_j^{\text{alt}}) - \log p(\mathbf{o}_i|B_i^{\max}) \tag{6.5}$$

This can be used to assess the performance of a new baseform B_j^{alt} by combining the improvements (a positive misclassification means that B_j^{alt} is preferred) for all instances of the word W_j in an adaptation set. We then get a measure of how much better B_j^{alt} models the variation given the adaptation set. When using indirect modelling by deriving pronunciation rules we must take into account that each rule probably will be applicable to several words. A way to

use this measure to evaluate a pronunciation rule is to consider all the words for which the rule in question is applicable. This is used in section 6.3.4.

To estimate the errors introduced, we need to combine the loss for all instances of all words in the vocabulary. One problem with this measure is that some errors will be new while others are not. We should therefore include all the baseforms B_j^m for the word W_j in the expressions. We redefine the discriminant function to be the log likelihood using the baseform that gives the highest score:

$$g_j(\mathbf{o}) = \log p(\mathbf{o}|B_j^{\max}) = \log p(\mathbf{o}|W_j) \tag{6.6}$$

Note that the true $\log p(\mathbf{o}|W_j)$ would be a sum of the likelihoods for all baseforms for the word W_j, but we have chosen to let $\log p(\mathbf{o}|W_j)$ denote the Viterbi approximation using only the best baseform $B_j^{\max}$.

6.1.1 Misclassification measures for one baseform

Using the same formalism as in equation (D.10) we define the difference between the discriminant functions for two baseforms as a misclassification measure. In equation (D.10) the difference was defined with respect to the best competitor, here we define the difference with respect to *any* competitor. The misclassification measure in equation (6.5) using the word based discriminant function in equation (6.6) is then:

$$d_{j|i}(\mathbf{o}_i) = g_j(\mathbf{o}_i) - g_i(\mathbf{o}_i) \tag{6.7}$$

If this measure is positive for at least one of the competing vocabulary words W_j we have a classification error. In the case of an error the discriminant function of the true word W_i gives a lower value than the competing word W_j for the segment $\mathbf{o}_i$,

One possible loss function is to count all errors making no assumptions on how well the absolute value of the misclassification measure represents the "amount" of error. This is the same zero-one loss function as shown in equation (D.11), repeated here:

$$l_{j|i}(\mathbf{o}_i) = \begin{cases} 0, & d_{j|i}(\mathbf{o}_i) \leq 0 \\ 1, & 0 < d_{j|i}(\mathbf{o}_i) \end{cases} \tag{6.8}$$

To express the expected loss as a function of the correct class W_k we combine the loss with respect to all classes. Summing over competing classes is equivalent to summing over all classes since the loss $l_{j|k}$ is zero for the correct

class:

$$R(C(\cdot), W_k) = \sum_{j=1}^{M} l_{j|k}(\mathbf{o}) P\{C(\mathbf{o}) = W_j | W_k\} \tag{6.9}$$

To estimate this value we use the N_k training samples $\mathbf{o}_k(n)$ of word W_k:

$$\begin{aligned} l_{j|k}(\mathbf{o})\widehat{P}\{C(\mathbf{o}) = W_j | W_k\} &= \frac{1}{N_k} \sum_{n=1}^{N_k} l_{j|k}(\mathbf{o}_k(n)) \cdot 1 \left[g_j(\mathbf{o}_k(n)) > g_k(\mathbf{o}_k(n)) \right] \\ &= \frac{1}{N_k} \sum_{n=1}^{N_k} l_{j|k}(\mathbf{o}_k(n)) \end{aligned} \tag{6.10}$$

If we use a different loss function $l_{j|k}(\mathbf{o})$, the last equality does not hold unless the loss for the correct class is zero. The estimate of the expected loss for word W_k corresponding to equation (6.9) is then:

$$\widehat{R}(C(\cdot), W_k) = \sum_{j=1}^{M} \frac{1}{N_k} \sum_{n=1}^{N_k} l_{j|k}(\mathbf{o}_k(n)) \tag{6.11}$$

We estimate the total expected loss by summing over all classes:

$$\begin{aligned} \widehat{R_0}(C(\cdot)) &= \sum_{k=1}^{M} P(W_k) \sum_{j=1}^{M} \frac{l}{N_k} \sum_{n=1}^{N_k} l_{j|k}(\mathbf{o}_k(n)) \\ &= \sum_{k=1}^{M} P(W_k) \frac{1}{N_k} \sum_{n=1}^{N_k} \sum_{j=1}^{M} l_{j|k}(\mathbf{o}_k(n)) \end{aligned} \tag{6.12}$$

This is an estimate of the *upper bound* on the average probability of error because we sum errors from all the competing classes. If there are many overlapping errors, we will not estimate the true error rate, but the measure can still be useful.

Counting only the true errors will mean that we only consider the best competitor and thereby may hide useful information. Improving the performance with respect to other competitors may give better results, since the best competitor in the training set may not represent unseen data well. A slightly different training set may give a completely different result. Alternative measures based on estimating the error rate that takes better care of this problem will be presented in the next section.

The word probability $P(W_k)$ can be computed from the language model training text, giving a language model dependency. For a unigram language

model this is a constant, but higher order N-grams require word history which complicates the matter.

The expression in equation (6.12) gets impractical when we introduce several new baseforms simultaneously. If we compute the measure for each new baseform, it is not clear how the numbers should be combined. Besides, the triple sum requires a lot of computation. Using pronunciation rules, each rule will probably bring up several new baseforms which must be evaluated. We will therefore present several redefinitions of the misclassification measure in equation (6.7) that better handle sets of baseforms.

6.1.2 Misclassification measures for sets of baseforms

Baseforms will interact, and we should therefore optimize on sets of baseforms (or rules). Besides, one pronunciation rule can give many baseforms, and assessing one rule may therefore mean assessing several baseforms. Errors may be both corrected and introduced, and we should consider the combined effect. We also want to reduce the complexity in the computation of the estimated error by combining the distances between the correct word and all competing candidates. The misclassification measure in equation (6.7) is therefore modified in several variants. This will give other estimates of expected loss than the upper bound shown in equation (6.12).

We introduce a misclassification measure where all competitors are taken into account through a function of all discriminant functions:

$$d_i(\mathbf{o}_i) = f(g_1(\mathbf{o}_i), g_2(\mathbf{o}_i), \ \dots \ g_M(\mathbf{o}_i)) \tag{6.13}$$

The first possibility is to expand equation (6.7) to consider the best competitor among all other words (this is the misclassification measure used in appendix D):

$$d_i^1(\mathbf{o}_i) = \max_{i \neq j} g_j(\mathbf{o}_i) - g_i(\mathbf{o}_i) \tag{6.14}$$

Using a zero-one loss function corresponding to equation (6.8):

$$l_i^1(\mathbf{o}_i) = \begin{cases} 0, & d_i^1(\mathbf{o}_i) \leq 0 \\ 1, & 0 < d_i^1(\mathbf{o}_i) \end{cases} \tag{6.15}$$

we get an estimate of the average probability of error:

$$\widehat{R_1}(C(\cdot)) = \sum_{k=1}^{M} P(W_k) \frac{1}{N_k} \sum_{n=1}^{N_k} l_k^1(\mathbf{o}_k(n)) \tag{6.16}$$

This estimate will only count one error per acoustic segment, whereas the estimated risk in equation (6.12) will count several errors per sample if several

classes have better score than the correct one. In speech recognition we will often see that if the correct word does not get the highest score, several other words will have higher score. Equation (6.16) will therefore give a better estimate than (6.12).

One problem when using error counts as in equation (6.14) is that we do not take into account the unseen errors as we optimize on the training set. The classifier performance on the borders between classes is more interesting for optimization than the best competitor, which may be an outlier. We want to express the amount of error as a function of the discriminant functions. One solution is to choose a discriminant function with an output that represents the amount of correctness.

The next suggestion for a misclassification measure is therefore to combine the distances from one word W_i to all the other words W_j, normalized with the number of other words $M-1$:

$$\begin{aligned} d_i^2(\mathbf{o}_i) &= \frac{1}{M-1} \sum_{j,j\neq i}^{M} [g_j(\mathbf{o}_i) - g_i(\mathbf{o}_i)] \\ &= -g_i(\mathbf{o}_i) + \frac{1}{M-1} \sum_{j,j\neq i}^{M} g_j(\mathbf{o}_i) \end{aligned} \tag{6.17}$$

The problem with the measure in equation (6.17) is that large (negative) values of $g_j(\mathbf{o}_i)$ will dominate the sum. Large negative values of the log likelihood mean that the competing baseforms are far from W_i, and we do not want a too large influence from these competitors.

One possibility is to restrict the sum to the words $\mathcal{M}$ that give errors according to equation (6.7), i.e. $g_j(\mathbf{o}_i) > g_i(\mathbf{o}_i)$. This will also reduce the computational complexity.

$$d_i^{2b}(\mathbf{o}_i) = \frac{1}{M'-1} \sum_{j\in\mathcal{M}} [g_j(\mathbf{o}_i) - g_i(\mathbf{o}_i)] \tag{6.18}$$

The disadvantage of this measure is that it may give problems with convergence as the set we sum over can be different for each baseform introduced.

The third measure we introduce is the smoothed misclassification measure used in GPD (Generalized Probabilistic Descent) [64]:

$$d_i^3(\mathbf{o}_i) = -g_i(\mathbf{o}_i) + \log \left[\frac{1}{M-1} \sum_{j,j\neq i}^{M} e^{g_j(\mathbf{o}_i)\cdot\eta} \right]^{1/\eta} \tag{6.19}$$

For large η the last term will be dominated by the best competitor and this measure will be equal to the misclassification measure in equation (6.14).

Large negative values of $g_j(\mathbf{o}_i)$, i.e. word baseforms far from W_i, will give minor contributions. The largest contributions to this misclassification measure will come from the words (baseforms) with best log likelihood score on the acoustic segment $\mathbf{o}_i$. This measure as well as only looking at the best competitor, equation (6.14), are used in section 8.3.7 to compare lexica derived by different rule derivation methods.

A new baseform introduced will belong to the baseforms of W_j, and to assess the performance we combine the misclassification measures for all words W_k in the vocabulary in a loss function. To find an estimate of the average probability of error we use the expected value of the loss function. We may introduce a doubt decision in the loss function to make the results more consistent with unseen events as shown in equation (D.12). A loss function often used in combination with the misclassification measure in equation (6.19) is a *sigmoid* function which gives a smoothed zero-one cost:

$$l_i^s(\mathbf{o}_i) = \frac{1}{1 + e^{-\xi d_i(\mathbf{o}_i)}} \tag{6.20}$$

The advantage of this loss function is that it quantifies the amount of error without putting to much emphasis on classifications far from the border between the classes. The constant ξ controls the region around the border. Both equations (6.19) and (6.20) are differentiable. An alternative loss function that emphasizes the border behaviour even more is a double sigmoid function:

$$l_i^{s2}(\mathbf{o}_i) = \frac{1}{1 + e^{-\xi d_i(\mathbf{o}_i)}} - \frac{1}{1 + e^{-\xi(d_i(\mathbf{o}_i)-\zeta)}} \tag{6.21}$$

Here, we ignore not only correct classifications, but also misclassifications far from the border. The motivation is that these cases are difficult to handle anyway so we might as well ignore them. However, using this kind of loss function we need to be very careful that we do not optimize on the wrong borders. We found no difference in performance between the two loss functions in preliminary experiments and have therefore only used the "single" sigmoid loss function $l_i^s(\mathbf{o}_i)$.

To estimate the probability of misclassification $\mathrm{pmc}(W_k)$, we sum over all segments belonging to word W_k and normalize by the number of samples/segments:

$$\widehat{\mathrm{pmc}}(W_k) = \frac{1}{N_k} \sum_{n=1}^{N_k} l_k^s(\mathbf{o}_k(n)) \tag{6.22}$$

To get an estimate of the total expected loss we take the expectation over

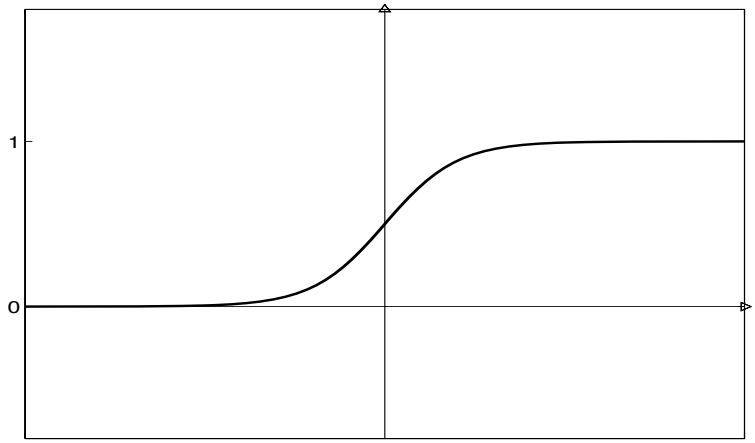

Figure 6.1: Sigmoid loss function.

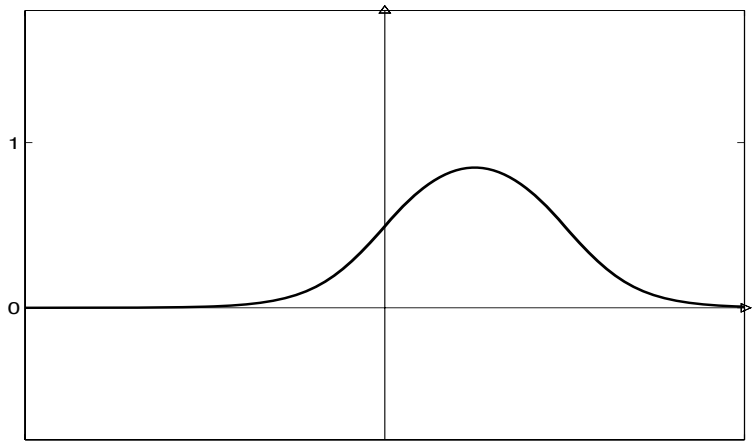

Figure 6.2: Double sigmoid loss function.

all words as in (D.15):

$$\begin{aligned}\widehat{R}(C(\cdot)) &= \sum_{k=1}^{M} P(W_k)\widehat{\mathrm{pmc}}(W_k) \\ &= \sum_{k=1}^{M} P(W_k)\frac{1}{N_k}\sum_{n=1}^{N_k} l_k^s(\mathbf{o}_k(n))\end{aligned} \tag{6.23}$$

This loss function gives us a method of assessing a lexicon given the recognizer's acoustic models and the acoustic training data. We may then compare a lexicon made using a combination of alternative baseforms or pronunciation rules to another baseline lexicon. The measure will not tell us which rules, or even baseforms, to keep or discard. Ideally, we should search for all possible combinations of baseforms. Data-mining algorithms like greedy search or evolutionary algorithms can be used to make a directed search instead of an exhaustive one.

6.1.3 Maximum mutual information

An alternative to minimum error rate as an objective function for optimization is maximal mutual information (MMI). Baseform assessment using MMI is presented for comparison.

In the maximum mutual information scheme we want to find the baseform $\widehat{B}$ that maximizes the mutual information between an acoustic segment $\mathbf{o}$ and a baseform B in the set of baseforms $\mathcal{B}$. The parameters of the recognition system are given by θ.

$$\begin{aligned}\widehat{B} &= \operatorname*{argmax}_{B\in\mathcal{B}} I(\mathbf{o}, B; \theta) \\ &= \operatorname*{argmax}_{B\in\mathcal{B}} E\{\log\frac{p(\mathbf{o}, B; \theta)}{P(B)p(\mathbf{o}; \theta)}\} = \operatorname*{argmax}_{B\in\mathcal{B}} E\{\log\frac{p(\mathbf{o}|B; \theta)}{p(\mathbf{o}; \theta)}\}\end{aligned} \tag{6.24}$$

For instantaneous mutual information we assume that the samples are representative and omit the expectation operator. The MMI can then be used as a

misclassification measure[3]:

$$\begin{aligned} I(\mathbf{o}_k, B_k) &= \log\left[\frac{p(\mathbf{o}_k|B_k)}{p(\mathbf{o}_k)}\right] \\ &= \log\left[\frac{p(\mathbf{o}_k|B_k)}{\sum_{j=1}^{M} p(\mathbf{o}_k|B_j)P(B_j)}\right] \\ &= \log\left[\frac{p_k(\mathbf{o}_k)}{\sum_{j=1}^{M} p_j(\mathbf{o}_k)P(B_j)}\right] \end{aligned} \tag{6.25}$$

Combining the scores for all classes gives the total mutual information:

$$I(\mathbf{o}, B) = \sum_{k=1}^{M} \log\left[\frac{p_k(\mathbf{o}_k)}{\sum_{j=1}^{M} p_j(\mathbf{o}_k)P(B_j)}\right] \tag{6.26}$$

This gives a normalization of the baseform B against all other baseforms applied to the acoustic segments that belong to the word we want to model.

In [64] the connection between MMI and the smoothed error count used in GPD is shown. Maximizing the mutual information is the same as minimizing its negative value:

$$\begin{aligned} -I(\mathbf{o}_k, B_k) &= \log\left[\frac{\sum_{j=1}^{M} p_j(\mathbf{o}_k)P(B_j)}{p_k(\mathbf{o}_k)}\right] \\ &= \log\left[P(B_k) + \frac{\sum_{j,j\neq k}^{M} p_j(\mathbf{o}_k)P(B_j)}{p_k(\mathbf{o}_k)}\right] \\ &\geq \log\left[\frac{\sum_{j,j\neq k}^{M} p_j(\mathbf{o}_k)P(B_j)}{p_k(\mathbf{o}_k)}\right] \\ &= -\log p_k(\mathbf{o}_k) + \log\left[\sum_{j,j\neq k}^{M} P(B_j)p_j(\mathbf{o}_k)\right] \end{aligned} \tag{6.27}$$

This equation is similar to the misclassification defined in equation (6.19). The main difference between MMI and the GPD-inspired discriminative techniques is therefore the smoothed error count used in GPD, equation (6.20). The MMI uses a linear function of the misclassification measure similar to equation (6.17) with the problems of optimizing directly on the sum of the log likelihoods. The MMI formulation has no loss function which gives an optimization function less comparable with WER. MMI therefore relies on the assumption that log likelihood is a good measure to assess the discriminative power of the lexicon.

[3]Here we switch to using class distributions $p_j(\mathbf{o}_k)$ for the conditional distributions $p(\mathbf{o}_k|B_j)$ and omit θ for readability

6.1.4 Limitations and practical considerations

One problem when using an adaptation set to measure the performance of an ASR system is that we will generally not have all the baseforms needed in operational mode for the system available (in sufficient numbers.) Care must be taken when generalizing from the misclassifications in an adaptation set, especially when we model variation by rules.

For speech recognition we will always have some model assumptions (e.g. HMMs). We have to decide how much trust we have in the log likelihood as a measure of the amount of correct classification when choosing the form of and the connection between the misclassification measure and the loss function. An alternative is to use error counts directly, but then we will only optimize with respect to the best competitor. Considering all competitors in a smoothed error count can be done by using a smoothed misclassification as in equation (6.19) and a loss function as in equation (6.20).

Computing the distance between all words in the vocabulary is very computationally intensive, and for many words the distance will be so large that it is of no interest. A less computationally intensive method is to restrict the total loss to include only the confusable words. If all words for which we do not compute the distance in equation (6.19), have a negative distance, a zero-one loss function will give the same result with this simplification as without. Using some of the correct classifications (the ones closest to the border) and a sigmoid loss function may give a better performance since we then include "almost" misclassifications. The drawback is that we need a threshold to decide which correct classifications to include. Another possibility is to use an N-best list to represent the competing hypotheses, this is used in sentence based discriminative training [14] and discriminative language modelling [69]. Another approach may be to look at the N competitors closest to the correct word. This will ignore misclassifications far from the border and have similar effect as the double sigmoid function shown in figure 6.2.

In the outline of measuring confusability of a lexicon the language model is included by $P(W_k)$ in equation (6.23). If we only consider unigram language modelling a simplified version of this equation can be found by using the usual estimate for the word probability:

$$\widehat{P}(W_k) = \frac{N_k}{N} \tag{6.28}$$

This estimate will give the unigram probabilities from the acoustic training set and will not necessarily be the same as a unigram computed on a language

model training set. Using this estimate the average error rate becomes:

$$\begin{aligned}\widehat{R}(C(\cdot)) &= \sum_{k=1}^{M} P(W_k)\frac{1}{N_k}\sum_{n=1}^{N_k} l_k^s(\mathbf{o}_k(n)) \\ &= \sum_{k=1}^{M} \frac{N_k}{N}\frac{1}{N_k}\sum_{n=1}^{N_k} l_k^s(\mathbf{o}_k(n)) \\ &= \sum_{k=1}^{M} \frac{1}{N}\sum_{n=1}^{N_k} l_k^s(\mathbf{o}_k(n))\end{aligned} \tag{6.29}$$

The total number of samples N will be the same for all lexica and can be ignored. The resulting estimate of the total number of errors for a lexicon is:

$$\widehat{\text{Error}}(\text{Lexicon}) = \sum_{k=1}^{M}\sum_{n=1}^{N_k} l_k^s(\mathbf{o}_k(n)) \tag{6.30}$$

This simplified estimate depends on comparable word distributions in the training and test sets.

6.2 Direct modelling of pronunciation variants

We have used a data-driven approach which utilizes the same Maximum Likelihood (ML) criterion for the lexicon design as for the training of the recognizer presented in [51]. The optimal baseform for each word is defined as the baseform B_j' that maximizes the acoustic likelihood of a set of sample utterances $\mathcal{T}_j = \{\mathbf{o}_j(n)\}$ of the word W_j, given a set of valid baseforms $\mathcal{B}$ and an HMM set defined by its parameters θ:

$$B_j' = \underset{B\in\mathcal{B}}{\text{argmax}}\{p(\mathcal{T}_j|B,\theta)\} \tag{6.31}$$

The grammar defining $\mathcal{B}$ in [51] was a monophone loop grammar independent of the word W_j. Other ways of deriving candidate baseforms include defining a word-specific set $\mathcal{B}_j$ with information from the canonical baseform. The method includes a clustering procedure also using the ML criterion to add several baseforms for each word to the lexicon.

Instead of allowing all possible candidate baseforms (i.e. all possible phone sequences of any length), we have restricted the search space to candidate baseforms obtained by an N-best phone loop transcription of the word examples. The same method using a 10-best phone loop has been shown to give improvements for pronunciation modelling of Norwegian natural numbers [3].

One problem with this direct approach is that we are restricted to modelling the variants seen in the training data (in sufficient numbers) as discussed in chapter 5. We have therefore expanded this theory to maximum likelihood based rule modelling.

6.3 Indirect modelling using pronunciation rules

Baseform variant generation by using data-driven rule derivation can be described in five steps, see section 5.4.

6.3.1 Reference and alternative transcriptions

For most experiments we have chosen to use a phone loop grammar as in [54] to make the alternative transcriptions. This will give many transcription errors, but the transcription will contain less influence from the canonical baseforms, and the lexicon will restrict the alternative transcription only via the acoustic models. The "best" baseforms do not always follow known phonological rules and systematic "errors" may be variation that we want to model. It is therefore important not to exclude possible rules before they are evaluated by the final assessment criterion. In our case the association based alignment will discard chance errors. The influence from the reference lexicon can be increased by using phone grammars trained on the lexicon, and we have performed some experiments with phone bigrams. This technique will still be completely data-driven.

For the word based transcriptions we have used an N-best phone loop. Usually the difference in log likelihood between the top words in an N-best list is not very large, so the second best transcription may also contain useful information. The most confident part will not change and will get a higher weight (count). For sentences an N-best transcription is of less use since usually only a few phones will change and the effect will almost be the same as repeating the sentence N times.

We have used both triphone and monophone models, this is described in the corresponding result chapters. Monophones are believed to better show pronunciation variation since triphones will contain, and therefore hide, some of the variation. On the other hand, triphones will usually give less transcription errors. For acoustic log likelihood based modelling the same acoustic models should be used to control the transcription as the ones that will be used in test.

Phone error rate is used to adjust the insertion rate in the automatic transcriptions. This is not consistent with the belief that these transcriptions

convey a truer transcription, some of the "errors" counted are the variation we want to model. As we have found no better measure, phone error rate (PER), is used (usually measured on the adaptation set). Phone error rate is computed in the same way as word error rate, see equation (2.12), counting both substitutions, deletions, and insertions as errors.

6.3.2 Alignment

We have used several alignment methods, first of all dynamic programming approaches with different ways of defining the substitution costs, but also a time synchronous alignment. Usually the substitution costs of the dynamic programming are either uniform or based on phonology. The deletion and insertion costs are usually set in advance to "reasonable" values.

The reference transcription will often be obtained automatically and will therefore contain errors. Manually obtained transcriptions will usually also contain labelling errors. Often only canonical pronunciations are used in the reference transcriptions and variation in pronunciation that deviates from this will also be "errors". The alignment based on such reference transcriptions is a challenge for the alignment algorithm. In our case the alternative transcriptions based on a phone loop will not follow normal phonotactical rules. For this "non-phonological" transcription errors a phonologically based alignment may not be appropriate. Still, some mappings are obviously more probable than others (e.g. a vowel mapped to another vowel instead of a plosive) so a uniform cost may not be the best, [19]. One important aspect is that the phone-to-phone mappings of pronunciation variations are not necessarily symmetric. (It is highly probable that a `[t]` is flapped, but less probable that a flap becomes a `[t]`.)

We want to estimate the mapping probabilities using the information in the data to govern the estimation. We therefore propose a new method called *association strength* based on statistical co-occurrences of phones. This data-driven alignment approach contributes to making the entire rule derivation data-driven.

The problems of alignment can be illustrated with the word "litter" with the reference transcription `[l ih t er]`. One of the (non-native) alternative transcriptions obtained using a phone loop was `[uw d iy jh er]`. Dynamic programming using phone group based costs gave the alignment:

```
[l ih t       er]
[  uw d iy jh er]
```

whereas the association based costs gave the alignment:

```
[   l ih t  er]
[uw d iy jh er]
```

Inspection of the timing of the alignments suggests that the second was a more reasonable alignment. The reason for the difference is that the mappings `[l -> d]`, `[ih -> iy]`, and `[t -> jh]` all were assigned high association strengths (low costs). The mapping `[iy -> uw ]` which may seem more phonetically likely than `[l -> d]`, did not appear frequently enough in the transcriptions to be assigned any association strength and was therefore given the default substitution cost. Even if the start of the word in this example may look like a transcription error that we do not want to model, the last part of the word may contain a useful rule. It should be noted that the alignment also will depend on the chosen insertion and deletion penalties.

It is important to align the same acoustic segments when deriving pronunciation variation. Irrespective of the substitution costs in the dynamic programming there will be "misalignments". One way to reduce this problem it to use word boundary information to avoid that a misalignment will be propagated to the next word. We have also investigated an alternative *time synchronous* alignment method where we align the reference and alternative transcriptions for each sentence using the time information given from the acoustic models. Using the log likelihood information from the transcriptions we can then automatically derive rules.

Association strength

A data-driven alternative to finding the phone-to-phone mapping costs is an hypothesis based estimate for phone-to-phone mappings called *association strength* based on statistical co-occurrences of phones introduced in [4].

The association method was first applied in [68] to generate grapheme-to-phoneme rules. We now apply it to find probabilities of phone-to-phone mappings by comparing two sets of phone transcriptions and finding systematic relations. The cost of a phone-to-phone mapping is set inversely proportional to the probabilities given by the association strength.

The differences between the reference transcription and the alternative transcription can be divided into systematic and unsystematic differences. The differences that are due to transcription and segmentation errors will not be systematic, but the differences due to baseform variation will be systematic. We want an algorithm that can capture all the systematic differences and filter out the random ones. If, for example, a speaker often substitutes `[s]` with `[z]`, we will often see an occurrence of `[z]` in the alternative transcription when `[s]` is present in the reference transcription. We want the algorithm to assign a high association value to the relation between these two phones. We would like to point out that these mappings are not necessarily symmetric, the probability of a mapping from phone A to phone B is estimated independently

of the mapping from phone B to phone A.

The method can be explained in a statistical framework as hypothesis testing of the mean of a binomial distribution [28]. If the occurrences of a specific phone (event A) in the reference transcription and a specific phone (event B) in the alternative transcription are independent, the events are subjects to the binomial distribution. The transcription was segmented and several occurrences of the same phone in each segment were counted as one occurrence. The segments can be words, phone groups, or if we have an initial alignment (as for the iterated version), one reference phone. We can estimate the probability of event B by dividing the number of alternative transcriptions that contain B by the total number of transcriptions. We call this estimate p. The number of occurrences of event A in the reference transcription is n, and k is the number of occurrences of event B in the alternative transcription given event A in the reference transcription. The probability of the number of occurrences of B given A can then be computed using a binomial distribution:

$$P(K = k) = b(k; n, p) = \binom{n}{k} \cdot p^k \cdot (1 - p)^{(n-k)}. \tag{6.32}$$

When the observed value of k is much higher than the expected value $n \cdot p$, the independence assumption is wrong. The probability of the number of occurrences of B given A using the binomial distribution formula will then be small:

$$k \gg n \cdot p \quad \text{and} \quad P(K = k) = b(k; n, p) < \epsilon \tag{6.33}$$

In the implementation we chose to use the negative logarithm of this probability as the association strength between event A and B:

$$S(A \Rightarrow B) = -\log[b(k; n, p)]. \tag{6.34}$$

In order to avoid a high association strength when there is a negative correlation between phones, we chose the restriction $k > n \cdot p$. Another option is to use the positive logarithm, in this case to get a low (negative) association strength.

Finally the algorithm may be iterated. Information from the alignment by applying the association based costs can be used to estimate the co-occurrences.

Which phones that are substituted because of baseform variation, may depend on both the language and the individual speakers. When capturing the phone-to-phone relations from the data we can easily find different sets for different speakers or groups of speakers.

To ensure that we find phone-to-phone mappings that are statistically significant, we choose to have a threshold on the association strength. If

the association strength is low this means k is close to $n \cdot p$. The probability in the binomial distribution is dependent on n, and we will get high strength for low probability mappings for large n. The threshold is therefore set according to the association strength for the mean $n \cdot p$ for the phone-to-phone mapping with the highest number of occurrences. Using the same threshold for all mappings will lead to that phones with low n need more occurrences to get a high strength. As we have lower confidence in phones with low n this will not be a severe problem.

One limitation is that the association strength relies on an estimate of the mean in a binomial distribution. The significance of the measure will be low when we have few samples n to estimate this mean from.

Time synchronous alignment

One reason for introducing a new time based alignment is that we want to ensure that we align transcriptions occurring approximately at the same time. Spontaneous speech often contains long pauses where the alternative transcription inserts spurious phones that should be discarded. An automatically derived reference transcription may contain errors compared with what is really said, especially in the case of restarts and incomplete words. Using the time information a misalignment for one segment does not effect the next segment.

We start with comparing the end times of the first phones in the two transcriptions. If the end times are *equal* (one frame deviation is allowed), this is counted as a substitution (or correct if the phones are identical) and we proceed to the next phones. If the end of the alternative transcription comes *before* the reference transcription, we investigate the neighbouring segments closer to decide if this is an insertion. If the end of the alternative transcription comes *after* the reference transcription, we investigate the neighbouring segments closer to decide if this is a deletion. When we observe equal end times and identical phones in the two transcriptions, this is marked as a possible rule border. In the reference transcription word boundaries are marked so the resulting time alignment also contains word boundaries.

An example of a case where time alignment gives useful information is the name "Clinton" with the canonical baseform `[k l ih n t ah n]` where we in the spontaneous dictation observed the phone loop transcription `[k l ah n]`. A dynamic programming using phone groups gave the alignment:

```
[k l ih n t ah n]
[k l        ah n]
```

whereas the time alignment gave:

```
[k l ih n t ah n]
[k l ah n       ]
```

which means that the second syllable is deleted, and not the first, even if the vowel of the alternative transcription is the same as the vowel in the canonical second syllable.

Another benefit is that we get initial rules directly by observing the segments where the two transcriptions differ. The combined alignment and rule derivation makes it easy to identify segments of no variation that can serve as borders for rules. We have defined segments with identical phones and time as "border" phones that give the context of the rule. For the example above the time segments differed for the `[l]` phones so it should not be used as a border phone, the rule should therefore be `[k l ih n t ah n] -> [k l ah n]` and not `[l ih n t ah n] -> [l ah n]`. Deriving rules across word borders using this method is straightforward. We have chosen to treat the variants occurring across word boundaries without silence in the same way as word internal rules. This is because we assume they will behave in the same way except for phone combinations not found inside words (in which case they make non-applicable rules that do not hurt the performance).

Last but not least, from the alignment we can also compare the log likelihood score for the reference and alternative transcriptions. From each instance of each initial rule we can compute an associated log likelihood ratio as explained in section 6.3.4. The major difference is that we use the acoustic segment for the rule and not the entire word. The derived log likelihood measure can be used directly in rule pruning or to discard low score rules before continuing with word based log likelihood pruning as in equation (6.36).

6.3.3 Rule derivation

We have adopted a rule notation similar to [18]: If a phone string `F` in the reference transcription is transformed to another phone string `T` in the alternative transcription, we write this as: `F` $\rightarrow$ `T`. `F` is called the *focus* of the rule. The transformation is dependent on the context surrounding the focus. A *rule* is therefore defined as a *transformation* `F` $\rightarrow$ `T` given left context `L` and right context `R`:

$$\text{Rule } r\text{: } \texttt{L-F+R} \rightarrow \texttt{T} \tag{6.35}$$

The reference transcription part of the rule, `L-F+R` is called the *rule condition*. All parts of the rule may include one or more phones. If the rule describes a deletion, the transformed phone is "empty", symbolized with `DELETED`.

We illustrate the derivation of rules from an alignment with two examples: 1) If the reference lexicon baseform of the word "states" is `[s t ey t s]`,

while the alternative `[s d ey t s]` is observed in the alternative transcription, we identify this variation as the rule: `s-t+ey` $\rightarrow$ `d`.
2) The word "bands" has the reference baseform `[b ae n d z]` and we observed alternative `[b ae n z]`. The rule in this case is: `n-d+z` $\rightarrow$ `DELETED`.

For the rules derived by the time synchronous alignment we may have a longer rule focus and there is no one-to-one correspondence between the phones in the reference and alternative pronunciation. Instead of a deletion symbol, we write this as the phones-to-phones mapping where the numbers of symbols in the focus and transformations are not necessarily the same. One example of this is the word "December" with the reference lexicon entry `[d ih s eh m b er]`. In the alternative transcription we observed `[dh iy s ah m er]` and the timing of the `[s]` was equal in the two transcriptions. This will give two rules: `d-ih+s` $\rightarrow$ `iy` and `s-eh m b+er` $\rightarrow$ `ah m`.
This last rule will be a combination of substitution and deletion.

6.3.4 Rule assessment and pruning

Log likelihood

We propose to use log likelihood improvements as a rule pruning measure, because this is more consistent with training of the rest of the recognizer. The acoustic models are trained using a maximum likelihood formulation and we have therefore chosen to use the same metric for rule pruning. The capability of the acoustic models to model the variation will then be incorporated. We compare the log likelihood of the baseforms affected by each rule with the log likelihood of the corresponding reference transcription using the misclassification measure from equation (6.5). The measure will thus be a log likelihood ratio, giving a normalization that makes rule comparison easier. The rule pruning measure $\mathcal{LL}$ for an acoustic segment $\mathbf{o}_i$ affected by a rule using log likelihood improvement is:

$$\begin{aligned}\mathcal{LL}(\mathbf{o}_i) &= \log[p(\mathbf{o}_i|B_i^{\text{alt}},\theta)] - \log[p(\mathbf{o}_i|B_i^{\text{ref}},\theta)] \\ &= \log\left[\frac{p(\mathbf{o}_i|B_i^{\text{alt}},\theta)}{p(\mathbf{o}_i|B_i^{\text{ref}},\theta)}\right]\end{aligned} \tag{6.36}$$

Here, B_i^{ref} is the reference baseform for the word W_i belonging to $\mathbf{o}_i$, and B_i^{alt} is the alternative baseform for the same word after modification according to the rule we want to assess. θ is the set of parameters of the recognizer used in computing the log likelihoods.

To assess each rule we use an improvement measure similar to the error estimate in equation (6.30). The most important difference is that we only

use the word in question and not the competing words, thereby removing the outer sum. The other difference is that instead of summing over the *loss* for each instance of the word we now sum over the *improvement*. For each rule several words will be affected and the improvement is computed as a sum of the positive improvements for each word. This gives the total improvement measure for rule k:

$$\mathcal{LLH}(\text{rule } k) = \sum_{W_j} \sum_{k(j) \in K(j)} \mathcal{LL}(\mathbf{o}_{k(j)}) \tag{6.37}$$

Here, $\mathbf{o}_{k(j)}$ is an acoustic segment for a word W_j affected by rule k. For each word W_j we sum over all acoustic segments $K(j)$ where $\mathcal{LL}(\mathbf{o}_{k(j)}) > 0$. We do not include negative log likelihood contributions because we always keep the reference transcription in the lexicon, and $\mathcal{LL}(\mathbf{o}_{k(j)}) < 0$ will mean that the reference baseform will be chosen. To get the total measure for each rule we sum over all affected words. All positive contributions are added, favouring the rules that are applied more frequently. This is deliberate, because we assume that the rule conditions which are most frequent in the adaptation also are most useful in testing. If this is not the case, a weighting factor should be applied. This will be a rough approximation of including word unigrams as explained in section 6.1.4.

The rules are sorted according to the $\mathcal{LLH}$ criterion and we can indirectly control the number of baseforms added by restricting the number of rules we apply. Because of the word unigram factor, frequently occurring rule conditions will be favoured and we will get more variants per rule on average than other methods where the language effect is not considered.

We will have no pronunciation probabilities directly from this measure.

Pronunciation rule probability

Estimated rule probability by frequency counts is a popular pruning measure. We have used two kinds of estimated rule probabilities as a sorting criterion comparing different thresholds governing the selection of the rules.

From the alignment we count the occurrences of all rules as well as the occurrences of all rule conditions. An estimate of the rule probability (RPR1) is the ratio between these counts:

$$\widehat{P}(\texttt{x1-A+x2} \rightarrow \texttt{B}) = \frac{\text{count}(\texttt{x1-A+x2} \rightarrow \texttt{B})}{\text{count}(\texttt{x1-A+x2})} \tag{6.38}$$

As a phone loop transcription contains errors, it can be advantageous to apply some kind of confidence measure to the alignment [54]. One possibility is

to limit the rule extraction to the segments with identical contexts in reference and alternative transcriptions. In this case the estimated rule probability (RPR2) is:

$$\widehat{P}(\texttt{x1-A+x2} \rightarrow \texttt{B}) = \frac{\text{count}(\texttt{x1-A+x2} \rightarrow \texttt{x1-B+x2})}{\text{count}(\texttt{x1-A+x2})} \tag{6.39}$$

The advantage is that we are more confident that we do not count alignment "errors" as rules, but on the other hand we have fewer occurrences of each rule. An "error" in `x1` or `x2` will wrongly inhibit the rule being counted.

Using this kind of estimation of the rule probability we will have probabilities for all segments present in the adaptation data. The sorting criteria before choosing which rules to use will be a combination of absolute and relative thresholds as well as a threshold for pronunciation probability derived by combining the effects of rules applied.

We have also tried sorting by occurrences of rule condition segment in the test lexicon, as well as word frequency estimated from the language model training text. This will include the language model in the optimization.

6.3.5 Pronunciation variant generation, assessment and pruning

There are two approaches to pruning that are often combined:

- Rule pruning before variant generation
- Variant pruning after variant generation

In the previous sections we have shown how rules can be pruned according to a log likelihood measure or a rule probability estimate. The resulting rules will then be used to derive variants. Multiple rules in each variant will give many variants, thereby increasing the confusability as discussed in section 5.4.5. We have chosen to use only one rule at the time. This is a suboptimal solution, but we assume the variants with only one rule applied are the most important ones and should contribute most. For the time synchronous rule derivation we consider a longer rule focus, and one rule may consist of several transformations. Another possibility is to use all rules to make a phone network for each word and assess the possible variants using equation (6.31).

Variant assessment is closely related to confusability reduction and will be treated in section 6.4.

6.3.6 Retranscription

Retranscribing the adaptation data using the derived rules as a restriction, is one way of obtaining an alternative transcription with less transcription errors. When using rule probability as the major sorting criterion the confidence in the transcription and alignment is important. This is because the estimate of the rule probability is based on counting occurrences in the alignment or on using the alignment for training CARTs. For acoustic log likelihood based rule pruning the usefulness of a rule is assessed by the rule pruning measure and not by the alignment. The importance of the alignment in the log likelihood rule pruning scheme is therefore first of all to find all possible good rules. Reducing the number of bad rules in the transcription and alignment phase speeds up the computation, but it is not crucial.

The procedure for retranscription can be illustrated for the word "numerous" with the reference baseform `[n uw m er ax s]`. From a 5-best phone loop transcription for one of the speakers we observed a transform from `[uw]` to `[ow]`, transforms from `[er]` to `[aa r]` or `[er r]`, and a deletion of the `[ax]`. The transform from `[er]` to `[aa r]` was discarded in the rule pruning. In addition, other rules concerning the phones in this word were found in other words. This gave a total of 2 possible transforms for `[uw]` (`[uw]` and `[ow]`), 8 for `[m]`, 2 for `[er]`, and 3 for `[ax]`. All rules were used to make a word lattice for this word. A forced alignment was used to choose from these $2 \cdot 8 \cdot 2 \cdot 3 = 96$ possibilities. The 5-best retranscription for the speaker gave only the transforms `[uw]` to `[ow]` and `[ax]` to `[ah]` or `[z]`, i.e. one transform that occurred in the phone loop transcription and two new transforms. Compared with the phone loop transcription 2 transforms were discarded (in addition to the one discarded in the rule pruning).

6.4 Confusability reduction

Confusability reduction techniques are often independent of how the variants are obtained, whereas they are found using knowledge or by direct or indirect data-driven modelling. The most usual confusability reduction method is to use variant probabilities to either reduce the number of variants or give a lower weight to less likely ones. When we assess pronunciation rules by acoustic log likelihood the variation already modelled by the acoustic models is included in the assessment. This should prevent adding superfluous variants and confusability.

These approaches only consider the quality of the baseforms for one word. We should ensure that the confusability measure used to assess the variants

corresponds to the WER in some way. To achieve this we must consider the baseforms of the entire lexicon. In section 6.1 we have used decision theory for pronunciation modelling to find an expression for the estimated total loss that can be used as a confusability measure. One problem with this approach is that we loose the unified optimization as we use different measures in the rule selection compared with the final rule or variant pruning.

An obvious measure for misclassification is the word error rate. This can be computed by recognition tests on adaptation data. The drawback, however, is that we only optimize with respect to the best competitor. An alternative is the smoothed error count explained in section 6.1 where we take into account the amount of correctness for all competitors. Different lexica for the same task can then be compared. The search for the best baseforms will be computationally expensive. Data-mining algorithms like greedy search or evolutionary algorithms can be used to make a directed search instead of an exhaustive one.

We would also like to evaluate the baseforms for the words not present in the adaptation set. This task can be solved if we are able to estimate the confusability for unseen words. One way to do this is using a phone confusion matrix. A distance measure for confusing the phone p_i with p_j is:

$$E\{\log p(\mathbf{o}_i|p_i,\theta) - \log p(\mathbf{o}_i|p_j,\theta)\} \tag{6.40}$$

Here $\mathbf{o}_i$ is an acoustic segment belonging to phone p_i.

To use this measure we have to decide how to combine the phone confusions to find the misclassification measure comparing two words. Summing up the distances of each phone will only be an approximation to the "true" misclassification. One way to reduce the error introduced by this is to use the phone confusions only to compute the distance for the phones that differ in the words we want to compare. The phone distance measure will in this way serve as a correction term for the difference introduced by for example a pronunciation rule. Note that to find the correction term, the phones of the two transcriptions must be aligned and deletions and insertions must be treated specially.

Another problem with the phone distance estimation is that the reference transcription will contain errors due to the mismatch we try to reveal through pronunciation modelling. Estimating the phone confusions from this transcription we find the distance between a reference transcription phone and an alternative transcription phone. In the same way as for pronunciation rule derivation, the context should be included in the phone confusion matrix.

Chapter 7

Comparison of ASR performance for different speaking styles

The main motivation for the experiments reported in this chapter was to investigate how standard ASR techniques perform for different speaking styles. Our focus is on the effect of pronunciation variants, but when using a general purpose lexicon. In the experiments we compare the use of one canonical baseform for each word with the use of baseform variants found in a standard ASR lexicon, as well as different levels of acoustic modelling.

Pronunciation variation may be handled in different parts of the ASR system, usually by acoustic or lexical modelling. The question of how to model the pronunciation variation has been addressed in e.g. [62] and [123]. The choice of method may depend on speaking style. In [82] investigations on acoustic model performance for read and conversational speech were presented. We present results on both acoustic and lexical modelling of pronunciation variation for read, spontaneous, and non-native speech. Our comparisons of different levels of acoustic modelling are especially interesting for spontaneous dictation and non-native speech and may contribute to better understanding of ASR performance for these speaking styles.

A second motivation for the experiments in this chapter was to find the optimal setting for a baseline for experiments on spontaneous dictation. We have therefore done extensive tests for the system used in chapter 9.

7.1 Introduction

Adequate handling of various speaking styles is one of the main challenges for ASR [1]. Expanding from the domain of read native speech, ASR systems will encounter more variability in the speech, e.g. more pronunciation variants. How we should model the pronunciation variation may depend on the speaking style. One of the important questions in pronunciation modelling is to decide which variation to model at the lexical level and what can be handled by the acoustic models [113].

Two main reasons for the improvement in performance of state-of-the-art ASR systems are more relevant training data and fast speaker adaptation methods. More complex acoustic models need more training data to get reliable estimates for the larger number of parameters. The acoustic models can be made more detailed by using more components in the observation probability density mixtures. Context-dependent acoustic models like triphones and quintphones are capable of modelling allophonic pronunciation variation. Usually the same canonical transcription is used in training irrespective of the speaking style, and we rely on the acoustic models to handle the deviation in pronunciations. The drawback of using more complex acoustic models to model the variation is that these models will be specific to the training data, and the ASR system may encounter problems in recognizing new speakers that do not fit into the same category as the training data.

For heavy accents, e.g. non-native speakers, the performance gain when introducing context dependency in the acoustic models has been small [119]. Speaker dependent adaptation of the acoustic models has given large improvements in recognition performance, but for non-native speakers the performance after adaptation is still far from comparable with native speech. Speaker variation among natives (accents) is usually more systematic, and model adaptation may give the needed shift in classes. For non-native speech there may be no systematic shift and the distributions are smeared out [119]. This kind of variation may be better handled in other ways, e.g. lexicon modelling.

A handcrafted lexicon will generally outperform general purpose lexica on the task for which it is optimized. However, the production of a manually optimized lexicon is costly and in many cases not feasible. It is therefore interesting to evaluate pronunciation variation issues using only publicly available resources. This is particularly interesting regarding portability issues and for languages where available handcrafted resources are limited.

We have investigated the use of pronunciation variants both in the training of the acoustic models and in testing for the different speaking styles. In this way we can compare acoustic and lexical modelling of the variation.

Acoustic model training

We have trained two sets of acoustic models:

1. "Canonical": trained using transcriptions based on a canonical lexicon
2. "Variant": trained using transcriptions based on a lexicon with variants

The "Canonical" set will model all the variation in the acoustic models. The "Variant" set will have less variation in the acoustic models, leaving more to lexical modelling. This scheme should in theory give more consistent training data for the recognizer and better acoustic models. Both monophone (context-independent) and cross-word triphone (context-dependent) models were trained.

Acoustic model adaptation

Acoustic model adaptation is one way to handle variation; this technique may also depend on speaking style, since the seed models we use for adaptation will fit the speaking styles differently. This is especially true for unsupervised adaptation where we rely on the transcription given by the original acoustic models. The performance of acoustic model adaptation is also dependent on the amount of available adaptation data. We have looked at adaptation using both one sentence, which is suitable for on-line adaptation, and 20 sentences, enough to achieve a substantial improvement for most adaptation methods.

Lexical task adaptation

Including pronunciation probabilities has given increased performance in many experiments, e.g. [123]. One way to derive the probabilities of pronunciation variants is to perform a forced alignment on a development set and use frequency counts to estimate the probability. For the non-native speakers we have an adaptation set available, and we have therefore used this speaking style to investigate the effect of using pronunciation probabilities.

7.2 Experimental procedure

7.2.1 Speaking styles

The task was chosen from the Wall Street Journal (WSJ) corpus available from LDC [127]. WSJ consists of several test sets with different speaking styles represented. The two "hub" tests `h1` and `h2` consist of read speech of the same type as the training set. In addition we have the "spoke" tests where we have

Speaking style	*Code*	*Grammar and Vocabulary*
Read, native	`h1`	20k, open (nvp)
Spontaneous dictation, native	`s9`	20k, open (vp)
Read, non-native	`s3`	5k, closed
Read, native	`h2`	5k, closed

Table 7.1: Speaking styles tested

used the tests `s3` with non-native speech and `s9` with spontaneous dictation, see table 7.1. The `h1` and `s9` tests both have a 20k open vocabulary. The `s9` test has verbal punctuation, which is not the case for `h1`. This is denoted by (vp) and (nvp) respectively in table 7.1. For each test we have used the corresponding test specific bigram supplied with the WSJ distribution.

The test sets consist of approximately 200 sentences from 10 speakers, except for the non-native test set which consists of 400 sentences (still 10 speakers).

7.2.2 Lexica

The CMU lexicon is a popular pronunciation lexicon for US English and available for free [15]. Alternate baseforms are marked so the lexicon can be used both as a "surface" lexicon with baseform variants and as a canonical lexicon by removing the alternatives. The basis of the lexicon is 20k words extensively proofed and used by the Carnegie Mellon University (CMU). Additional words and baseforms are added from several unproofed sources, but only words or baseforms that are found in two or more sources have been used, see a more detailed description in appendix B. In the subset used for the 20k vocabulary, 3174 of the words have baseform variants. The maximum number of variants for one word is 6; see table 7.2. In the subset of the lexicon used for the 5k vocabulary 1060 words have multiple baseform variants. On average there are 1.2 variants per word.

We also did some comparative tests using Pronlex available from LDC [89]. This lexicon is also widely used and it is claimed to be more consistent, but is not free. Pronlex is based on more phones than the CMU lexicon; 42 versus 39. The average number of baseform variants per word in Pronlex is 1.1 and the canonical baseforms are not marked. Because of this we trained only context-dependent models with baseform variants when using Pronlex. The results showed hardly any difference in performance between the two lexica. This indicates that the CMU lexicon is a state-of-the-art non-adapted lexicon.

Number of baseforms	*Number of words*
6	13
5	4
4	107
3	228
2	2822
1	16826

Table 7.2: Number of baseforms per word in the CMU lexicon for the 20k WSJ vocabulary

7.2.3 The HTK reference recognizer

We have trained a baseline recognizer using the HTK toolkit [131], using fairly standard methods [126]. We have used the standard WSJ training set consisting of read speech from 284 speakers (SI-284), a total of 37500 utterances. We used MFCC feature vectors with 13 elements, including normalized energy, and added the first and second derivatives giving a total of 39 elements. Each feature vector was derived from a speech frame with a Hamming window of length 25 ms and a 10 ms frame rate.

The RM-models supplied with HTK were used as seed models. The CMU phone set has 39 phone models and we used two silence models, giving a total of 41 monophone models. The two silence models were one context-independent for word ends with longer pauses and one context-free for word ends without long pauses. The two HMM sets, the canonical version and the variant version, were trained separately using Baum-Welch reestimation. For the context-dependent models, different decision trees for clustering triphone states were built for the variant and canonically trained HMMs. The set of 41 monophone models (one context-independent and one context-free) gives a total of 62441 logical cross-word triphones to cover all possibilities. The training data contained 18814 different triphones for the variant setup and 17893 for the canonical setup giving a total of 56442 and 53673 states respectively. The decision tree based clustering was done using state tying and the number of states after clustering was approximately 10000. The number of distinct (physical) triphones was approximately 28000 after tying.

Training of canonical and variant models

When training acoustic models using baseform variants, the acoustic models are used to choose which baseform variant to use for training. This is done by retranscribing the training data with forced alignment using a variant lexicon. Then this transcription, believed to better represent the variation seen in the data, is used for further training. The most common scheme is to retranscribe the training data once using monophone models and then use this transcription for the rest of the training. We did some preliminary experiments with this type of variant trained models using 4 iterations at each mixture increase.

To make sure that the baseform variants affect the models we also tried a scheme where we retranscribed the data for every increase in the number of Gaussians in the observation probability mixture. For every level of mixture components we used 4 iterations of Baum-Welch reestimation using the transcription from the previous level. Then the reiterated models were used to retranscribe the data by choosing baseform variants. This transcription was then used to iterate 4 more times. The two different retranscription schemes were performed both for the monophone and triphone models.

For the canonically trained HMMs we used 4 iterations for every level of mixture components both for monophone and triphone models. To be sure that the number of iterations did not favour the variant trained models, we also trained canonical models using 8 iterations.

For the tests we used 10 Gaussians in the mixture for triphone models. For the monophone models, increasing the number of Gaussians from 16 to 32 did not give substantial improvement, so we decided to use 16. For the two silence models we used twice as many Gaussians as the other phone models as in [126].

Acoustic model adaptation

For acoustic model adaptation we used unsupervised adaptation and the Maximum Likelihood Linear Regression (MLLR) adaptation method [79]. The adaptation tests were performed using triphone models only, but for both the canonical and variant trained versions. We tried two adaptation schemes:

1. One-sentence adaptation where each sentence was used to find one global MLLR transform. The sentence was then re-recognized using this transformation. This is a reasonable adaptation scheme for telecommunication services where each speaker is active for a short amount of time and we have no information about speaker identity.

2. Incremental adaptation on 20 sentences from each speaker. In this way

each sentence from a speaker will update the transform that is used to recognize the next sentence. For this adaptation scheme we used regression classes in a tree structure so the number of transformations was chosen according to the amount and type of data available.

Lexical task adaptation

Using the adaptation set and the training set it is possible to derive speaking style dependent pronunciation probabilities from forced alignment. WSJ includes an adaptation set with the same speakers as in the test set for the non-native task. The adaptation set consists of 40 sentences per speaker and the speakers read the same sentences. This means that the adaptation set has a very limited vocabulary, only 349 different words, but all words are repeated at least 10 times (once by each speaker).

7.3 Results

Some of the results reported here were presented in [6].

For all the tables in this section the "Training" column shows the lexicon used in training of the acoustic models and the "Test" column shows the lexicon used in test. The last line shows a "mismatch" test where we used variants in acoustic model training but not in the test lexicon. The figures show relative improvement in WER to make a comparison between the speaking styles easier. The term "significant" is used when the McNemar test gives a p-value less than 0.01, although the errors can not be claimed to be independent when using a bigram language model, see section 2.5 for more details on statistical considerations.

Details on experiments performed to choose the number of iterations for every mixture update are given in appendix E. The chosen setting for the canonical models is 4 iterations of retraining for every mixture update. For the variant setup the choice is 8 iterations where a retranscription is performed after the 4th iteration and 4 more iterations are performed with the new transcription. All the results presented in the next section, both for monophone and triphone acoustic models, are derived using this setup.

7.3.1 Context-independent models

For the monophone models we observed small improvements by including lexical variants in the test, see table 7.3. The improvements relative to the canonical setup are shown in figure 7.1. Using the McNemar test the improvements using variants in test were only significant for `h1` and `h2`. The further

Training	*Test*	*h1*	*s9*	*s3*	*h2*
Canonical	Canonical	34.7	40.4	27.6	22.2
Canonical	Variant	32.9	39.3	27.2	21.1
Variant	Variant	32.0	38.2	26.7	20.1
Variant	Canonical	35.8	40.8	28.2	22.2

Table 7.3: WER in [%] for monophone acoustic models

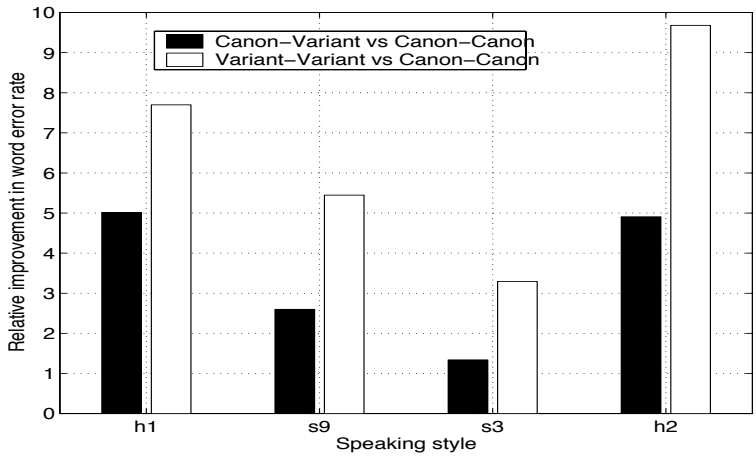

Figure 7.1: Relative WER improvement from canonical pronunciations in both training and test using monophones

improvement using variants both in training and test was not significant. For `s9` we had to use variants in both training and test to get significant improvement. For `s3` the improvement was small, there were no significant differences other than the deterioration of the mismatch test. The mismatch test (using variants in training, but not in test) gave significant deterioration compared with using variants in both training and test for all speaking styles.

7.3.2 Context-dependent models

The improvement seen using variants for the monophone situation was not as uniform for triphones, see table 7.4 for WER results and figure 7.2 for improvements relative to the canonical setup. We actually see a deterioration for the two speaking styles `s9` and `s3`. The deterioration when using variants

Training	*Test*	*h1*	*s9*	*s3*	*h2*
Canonical	Canonical	15.9	23.7	27.7	8.0
Canonical	Variant	15.2	23.5	28.4	8.4
Variant	Variant	15.4	24.4	28.9	7.4
Variant	Canonical	20.5	29.4	31.0	11.6

Table 7.4: WER in [%] for triphone acoustic models

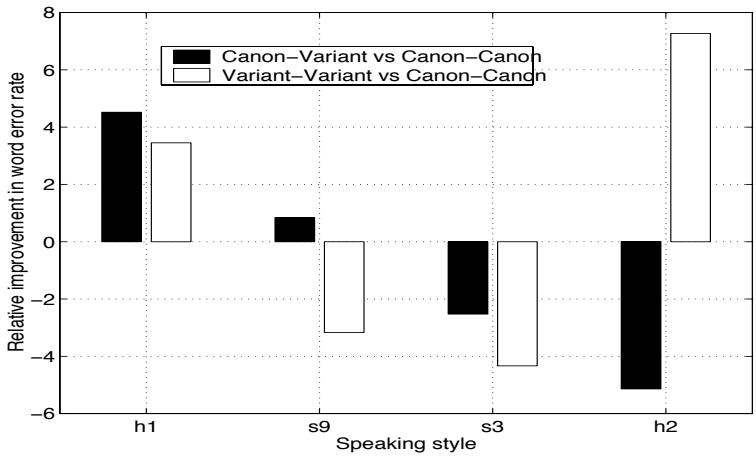

Figure 7.2: Relative WER improvement from canonical pronunciations in both training and test using triphones

in both training and test for `s3` is significant. For `h2` we see a significant improvement when using variants in both training and test. When using variants only in test we see a deterioration for `h2` that is not consistent with the other read native task `h1`. There was a significant difference between the mismatch test (using variants in training, but not in test) for all speech types compared with both the canonical setup and using variants only in test.

7.3.3 Comparison of context-independent and dependent models

The improvement from context-independent to context-dependent modelling gave the largest difference for `h2`, see figure 7.3. Again `s9` behaved more similar to `h1` than `s3` to `h2`. For `s3` there was no significant change for the canonical setup, and for the variant setup there was a significant deterioration.

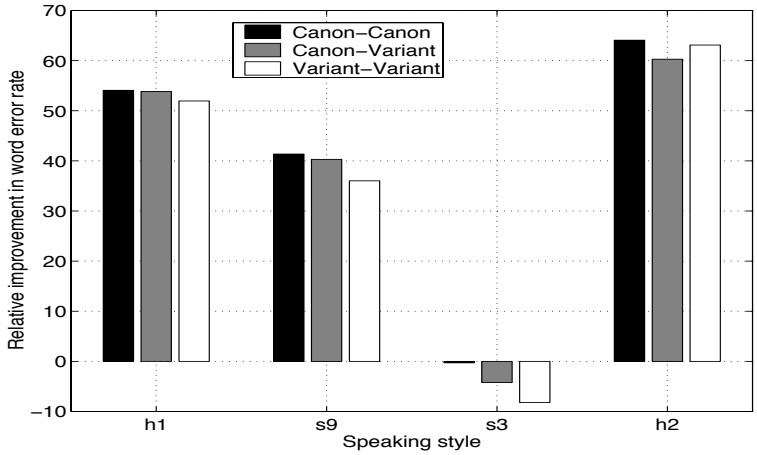

Figure 7.3: Relative improvement in WER from monophones to triphones

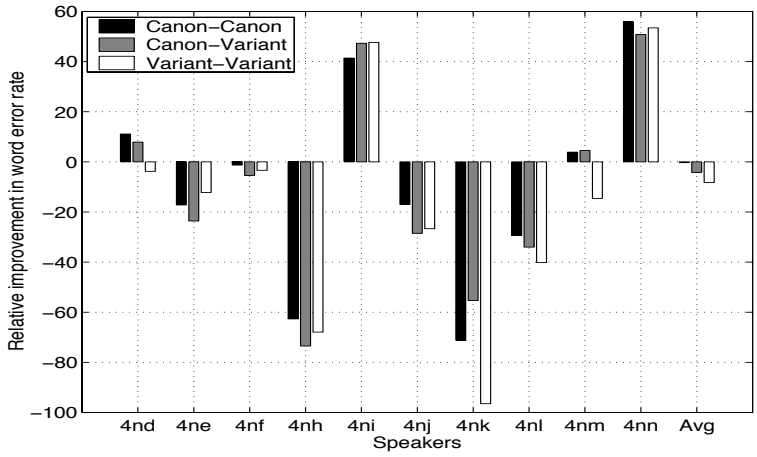

Figure 7.4: Relative improvement in WER from monophones to triphones per speaker for `s3`

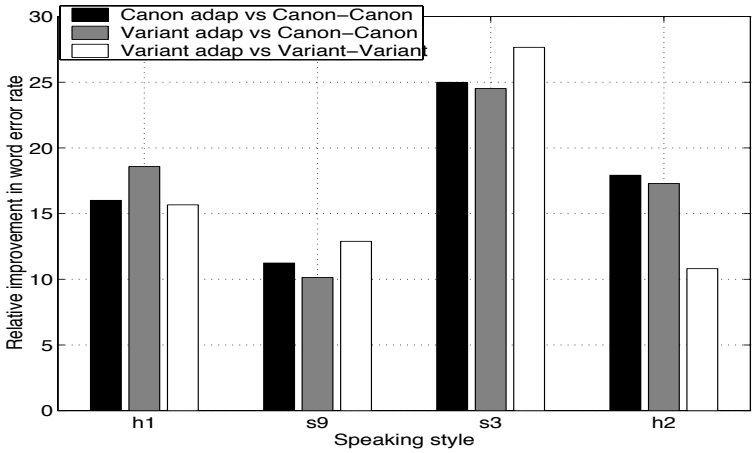

Figure 7.5: Relative improvement in WER using 20 sentence adaptation on triphone models

One reason for this is that the variant triphones performed worse than the canonical triphones. The increased modelling capability of the triphones does not help for non-natives on average. As seen in figure 7.4 the performance for the non-native speakers was very variable.

7.3.4 Acoustic model speaker adaptation

Using 20 sentences for MLLR speaker adaptation of the acoustic models gave as expected a significant performance gain, see table 7.5. Adaptation using variant models gave no or small improvement compared with the canonical models, see figure 7.5. The largest adaptation gain was seen for the non-native speakers, whereas the spontaneous dictation gave the smallest gain.

For the `h1` task we see the same increased performance by using variant models over canonical models with adaptation as we saw without adaptation. For the other tasks it was no surprise to see that adding variants did not give any increased performance over acoustic model adaptation only, as they gave no improvement for unadapted models. The smaller increase using variant adaptation over variant models seen for the task `h2` is due to the fact that using variants without adaptation gave a better performance. The resulting performance is equal, so we here see the effect of modelling the same variation either by general purpose variants in the lexicon or acoustic model speaker adaptation.

MLLR adaptation using one sentence gave no improvement, see table 7.5.

HMMs	*h1*	*s9*	*s3*	*h2*
Unadapted canon	15.9	23.7	27.7	8.0
1 sent. canon adap	16.2	23.3	27.3	8.0
20 sent. canon adap	13.4	21.0	20.8	6.6
Unadapted variant	15.4	24.4	28.9	7.4
20 sent. variant adap	13.0	21.3	20.9	6.6

Table 7.5: WER in [%] for triphone setup and adaptation

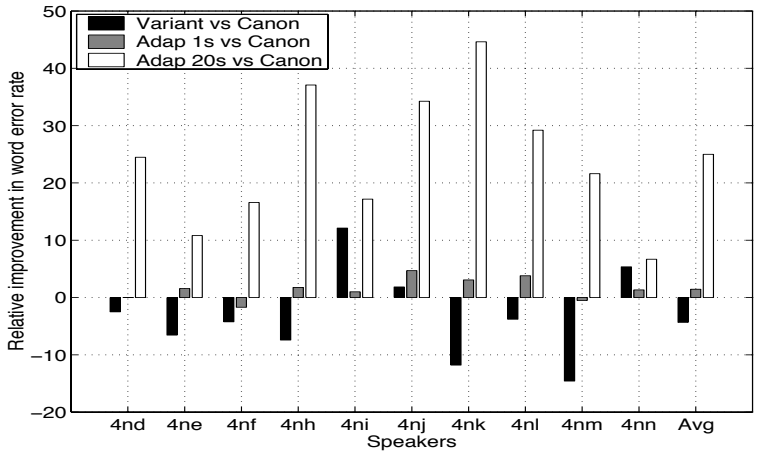

Figure 7.6: Relative improvement in WER using pronunciation variants and speaker adaptation for `s3`, results per speaker

No difference was observed between using one global transform approach and a regression tree where the number of transforms is given by type and amount of data. One sentence adaptation calls for more sophisticated adaptation methods than MLLR [107].

In figure 7.6 we compare the improvement per speaker in the non-native `s3` set for the 20 sentence and the 1 sentence acoustic model adaptation and the lexical modelling of variation by including pronunciation variants in training and test. The acoustic model adaptation gain was more uniformly distributed over the speakers compared with using baseform variants in the lexicon.

We saw the largest improvement by using variants in training and test for the read speech 5k vocabulary task `h2`. The performance increase was more uniform, but still less than the adaptation using 20 sentences, see figure 7.7.

Since the monophone acoustic models performed similar to triphones for

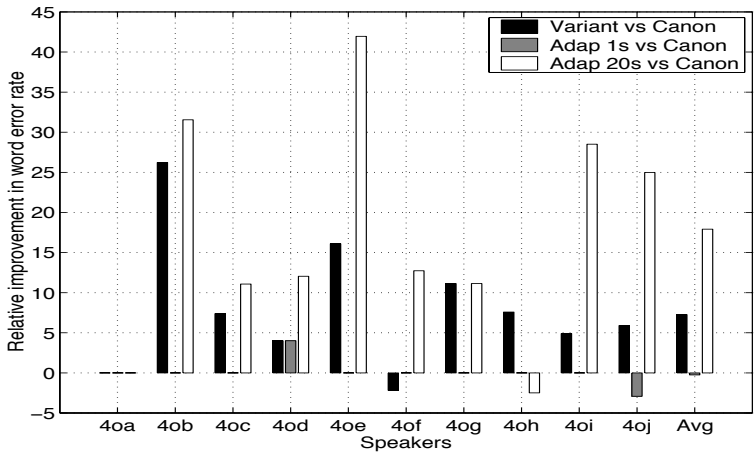

Figure 7.7: Relative improvement in WER using pronunciation variants and speaker adaptation for `h2`, results per speaker

non-native speech, we also tried adaptation on monophone models for this speaking style. For the canonical monophone models and 20 sentence adaption the result was 24.1% WER. This is better than without adaptation (27.6% WER) but worse than the triphone adaptation (20.8%). This shows that the increased modelling capacity of the triphones is utilized in speaker adaptation even if this was not the case for the speaker independent models.

7.3.5 Error analysis

Even if we got similar recognition rates using canonically and variant trained HMMs, this does not mean that the recognition results were identical. There are both errors corrected and errors introduced, see tables 7.6 and 7.7. The insertions are counted as errors according to the WER formula in equation (2.12). The total number of "correct" and "error" will therefore be larger than the total number of words, which is 3446 for `h1` and 7435 for `s3`. For `h1` 25.3% of the errors for the variant system were different from those in the canonical system. This was similar for all speaking styles; 23.9% for `s9`, 29.0% for `s3`, and 27.4% for `h2`. The differences between the canonical system and the system with canonically trained HMMs, but variant lexicon used in test, were not that large: 6.5% for `h1`, 9.8% for `s9`, 12.9% for `s3` and 9.3% for `h2`.

Even if the variants in these experiments do not help recognition performance, they do make a difference. Our results are similar to the ones shown in [123] and indicate that improving the recognition performance could be

		Canonical Training + Test	
		Correct	*Error*
Variant Training + Test	Correct	2888	153 (27.9%)
	Error	134 (25.3%)	396

Table 7.6: Error analysis on word level for `h1`

		Canonical Training + Test	
		Correct	*Error*
Variant Training + Test	Correct	5708	533 (25.9%)
	Error	623 (29.0%)	1524

Table 7.7: Error analysis on word level for `s3`

possible by careful selection of baseform variants.

7.3.6 Pronunciation probabilities

Using an adaptation set it is possible to derive speaking style dependent pronunciation probabilities from forced alignment and use these to select which variants to include in an adapted lexicon. This is a simple way of incorporating speaking style dependent lexical adaptation.

As the non-native speakers potentially would gain more from pronunciation modelling we chose this task for some preliminary experiments. In an adaptation set with the same speakers as in the test set all speakers read the same 10 sentences 40 times. The vocabulary size of this set is only 349, and 256 of the words were present in the 5k test vocabulary. Only 92 of these words had baseform variants in the CMU lexicon. This is a small number of words compared to the total number of words in the vocabulary (5000), but we assume that these are frequent words that are most important to model, since frequent words incorporate more pronunciation variation [31]. The error distribution of function words (which are frequent), like "a", "an", "and", "are", "as", in this 92-word list showed that these words were involved in many errors and confirms that they are important to model. The baseforms never selected were left out, and 71 words were left with variants. We did experiments both with and without pronunciation probabilities added, in both cases removing the baseforms not seen in the adaptation set.

The experiments did not show any improvement, even with different values of the pronunciation probability scaling factor. In fact, we saw a small deteri-

Speaking style	*HMMs*	*Acc. score*	*LM. score*
h1	monophones	97.5	2.6
s9	monophones	97.6	2.4
s3	monophones	97.85	2.15
h2	monophones	97.6	2.4
h1	triphones	96.9	3.1
s9	triphones	97.1	2.9
s3	triphones	97.4	2.6
h2	triphones	97.25	2.75

Table 7.8: Contribution from acoustic score and language model score in [%]

oration. The reason may be either that this approach calls for larger amounts of adaptation data than we had available, or that the variants present in the CMU lexicon were not representative for the non-native speakers. Collecting sufficient speaking style dependent baseforms "by hand" is infeasible. There is a need for methods for deriving and assessing baseform variants that are more automatic and more consistent with WER. Data-driven baseform variation modelling is one answer, but requires sufficient amounts of representative language resources.

7.3.7 Language model effect

To assess the impact of the language model on the different speaking styles we compared the contributions from the acoustic score and the language score in the recognized output, see table 7.8. The most striking result is that the distribution of the scores was very similar for the different speaking styles and that the acoustic score was by far the major factor in the overall score. We note, however, that the read native 20k vocabulary had the highest influence from the language model, whereas the read non-native 5k vocabulary had the lowest.

Another approach is to test without a language model, i.e. a word-loop. This was infeasible for the 20k vocabulary. For the two 5k tests the word-loop tests gave a very high error rate: 70.6% for s3 and 32.8% for h2 with the variant setup and triphone acoustic models. This means that the non-native speech gave about twice the word error rate of the native speech on the same task when the language model effect was removed. When using a bigram language model, the non-native speech had more than three times the error rate of the native speech. We should however be careful with the conclusions

Lexicon	*Model type*	*Language model*	*h1*	*h2*
CMU lexicon	Gender independent	bigram	15.4	7.4
Dragon/HTK	Gender dependent	bigram	14.5	6.9
Dragon/HTK	Gender dependent	trigram	12.7	5.0

Table 7.9: Comparison of WER in [%] between results in this chapter and experiments in [126] for SI-284 trained systems

from this experiment as the word error rates for the word-loop were very high. They indicate, however, that the language model effect is more beneficial for the native speech, which is also an intuitive result.

7.4 Comparisons with other systems

The `h2` and `h1` results for context-dependent models can be compared with the results reported in [126] shown in table 7.9. The results were used as the CUED-HTK contribution to the ARPA November 1993 WSJ evaluation and gave the second best result for the `h1` trigram test and the best result for the `h2` trigram and the `h1` bigram. One major difference is the lexicon used in [126] which is the Dragon WSJ pronunciation lexicon version 2.0 with locally generated additions and corrections. Another difference is that HTK version 1.5 was used, but this should not have any great impact on the recognition results.

The "CMU lexicon" line is the variant versions from table 7.4. Since variants were used in the CUED-HTK setup this gives the most appropriate comparison. For the SI-284 set no gender independent results were reported, which makes comparison with the CMU lexicon setup reported in this chapter difficult. For both the `h1` and the `h2` task the results from this chapter are comparable with the gender dependent results. We also note that a trigram language model gave a substantial increase in performance.

In chapter 8 we have used the Bell Labs automatic speech recognizer (BLASR). This system was only trained on a smaller training set, SI-84 (84 speakers). A comparison of this system with the CUED-HTK system is given in table 7.10. As we can see, the baseline is also in this case comparable with the CUED-HTK result. No trigram results for models trained on the SI-84 set were given in [126].

Lexicon	*Model type*	*Language model*	*h2*
BLASR	Gender independent	bigram	10.9
BLASR	Gender independent	trigram	7.8
Dragon/HTK	Gender independent	bigram	8.7

Table 7.10: Comparison of WER in [%] between results in chapter 8 and experiments in [126] for SI-84 trained systems

7.5 Discussion

For pronunciation variation depending on speaking style, the question of lexicon versus acoustic model adaptation has no clear-cut answer. The results presented here, using variants present in a general purpose lexicon, show a discouraging performance when lexical variants were included, only small gains. We observe that it seems more important *not* to include variants in the training when we are uncertain which variants will be used in the test. Models trained without variants were able to use a lexicon with variants with equal or better performance. The models trained on variants showed a significant decrease in performance when using a canonical lexicon in test. This is probably because these models are less diffuse and therefore more tailored to the lexicon they are trained on.

The variants in the standard purpose lexicon tested gave modest improvements. The largest improvements were seen for context-independent models where all speaking styles except non-native speech had a significant improvement. We observed the expected increase in performance when adding variants in the lexicon, and a further increase when also including the variants in training. For the context-dependent models the variants did only help for read speech, which is the speaking style of the acoustic model training set. The increased modelling capacity of context-dependent models could apparently handle the observed variation just as well as the variants in the CMU lexicon.

We observed, however, that the errors differed: About 20% of the errors were different when using variants compared with using only canonical pronunciations. This suggests that there is a potential in selecting variants. To learn more about the contribution of the variants in the lexicon a more thorough error analysis is needed.

The use of context-dependent models gave a large gain compared with context-independent models for all speaking styles except non-native speech. The lack of improvement for non-native speech is also observed in [119]. Context-dependent models trained on native speech are apparently not very

good for modelling non-native speech, and more sophisticated methods for this difficult task are necessary.

Acoustic model adaptation gave the largest gain for non-native speakers, while the spontaneous dictation had the least improvement. It is reasonable to assume that the acoustic model mismatch is highest for non-natives, and speaker adaptation therefore will be most beneficial in this case.

Even if the spontaneous dictation encounters less variation than conversational speech, we saw a large degradation compared with read speech. Even with speaker adapted models, the performance was much worse than read speech, and the gain for adaptation was also less than for read speech. Why the adaptation gain was lower is difficult to explain. One reason may be that as we have used unsupervised adaptation in the experiments, the artifacts of spontaneous speech may give automatic transcription with lower quality.

Language modelling by including pronunciation probabilities did not give better performance. The pronunciation probabilities were derived using forced alignment on an adaptation set. Since the errors with and without pronunciation variants differ, it is reasonable to assume that filtering out non-useful variants should be beneficial. For the experiments reported here, this was not the case. One reason may be that the number of variants was low. Looking at the distribution of the language model part of the total score compared with the acoustic model part, showed no big difference between the speaking styles. Experiments using word-loop grammars indicate that the language model helps more for native speakers than for non-native speakers even for read speech.

7.6 Summary of the main results

- Pronunciation variants in the general purpose CMU lexicon gave only modest improvements over using one canonical baseform.
- The improvement seen was largest when using context-independent acoustic models and for read native speech.
- Error analysis of the canonical system compared with the variant system revealed that about 25% of the errors differed, although the resulting word error rates of the two systems were similar.
- The improvements of context-independent acoustic models over context-dependent acoustic models were largest for native speech and substantial for spontaneous speech. For non-native speech there was no improvement.

- Speaker dependent acoustic model adaptation gave largest gain for non-native speakers. The improvement was larger using context-dependent acoustic models than context-independent acoustic models, although the baseline performances of the two models were similar.
- The improvement for spontaneous dictation when using speaker dependent acoustic model adaptation was significant, but smaller than for read speech.
- Non-native speech and spontaneous dictation performed much worse than read native speech even after speaker adaptation.

The poor gain when using variants may be interpreted as a demand for more care in generation and selection of pronunciation variant candidates. Data-driven variant generation and lexicon optimization using an objective criterion [51] is one such approach.

Two issues are important in pronunciation modelling: 1) candidate pronunciations, and 2) a way to assess these pronunciation variants. To assess pronunciation variants we need representative data. Methods based on pronunciation rules instead of directly on variants can generalize to pronunciations not present in the training data, and will make it possible to assess these unseen pronunciation variants. This is treated in chapters 8 and 9.

Chapter 8

Data-driven pronunciation modelling for non-native speech

8.1 Introduction

Automatic speech recognition of non-native speakers with different mother tongues is a difficult task due to the large variation between the speakers. Modelling speaker variation can be done in several ways, the most usual is making the acoustic models more detailed or adapting them to each speaker. For large vocabulary speech recognition a well designed language model will also help, as a possible mismatch between the speaker and the acoustic models may be overruled by the language model. In chapter 7 we saw that the ASR system performed substantially worse for non-native than for native speech. Speaker dependent acoustic model adaptation improved the system, but the performance was still far from the native speech recognition rate. Some of the variation in pronunciation caused by a non-native mother tongue may be better handled by careful design of the pronunciation dictionary, i.e. pronunciation modelling. In this chapter we use some of the methods presented in chapter 6 to perform lexicon adaptation. Using the lexicon to capture the speaker variation makes it possible to model several speakers simultaneously, thus using the same lexicon and the same acoustic models for all speakers. The experiments include both joint and speaker dependent lexicon adaptation.

Identity	*Native country*	*Gender*
4nd	Argentina	male
4ne	France	male
4nf	France	male
4nh	Israel/Argentina (Spanish)	female
4ni	Denmark	male
4nj	Israel	female
4nk	Japan	female
4nl	Germany/Spain (German)	female
4nm	Nicaragua	female
4nn	New Zealand	male

Table 8.1: Native country and gender for the speakers in the database.

Segment type	*Adaptation*	*Test*
Sentences	400	416
Words	5229	7435
Phones	22749	

Table 8.2: Distribution of adaptation and test set for non-native speech.

8.2 Experimental procedure

8.2.1 The database

We have applied our pronunciation modelling methods to the Wall Street Journal (WSJ) adaptation and test sets for non-native speakers of US English (`s3`). More information on WSJ is given in section 7.2.1. The `s3` part of the test consists of 10 speakers reading 40 sentences for adaptation and 40–43 sentences for testing, see table 8.1. The test set consists of 1401 different words. The 10 speakers read the same adaptation sentences, giving a total of 349 different words in the adaptation set, 256 of which were found in the test vocabulary (i.e. in the WSJ 5k language model). In the lexicon adaptation we needed transcriptions for all words and for the missing baseforms we used the lexicon generated for a 64k WSJ vocabulary. The size of the adaptation set is 5229 words, thus each word is repeated 15 times on average, see table 8.2.

Speaker identity	*WER*	
	bigram	*trigram*
4nd	36.4%	34.7%
4ne	37.9%	35.7%
4nf	38.6%	32.3%
4nh	43.9%	38.3%
4ni	14.9%	11.4%
4nj	29.2%	24.7%
4nk	32.9%	30.6%
4nl	25.0%	22.7%
4nm	37.6%	36.9%
4nn	28.1%	24.2%
Average	32.5%	29.2%

Table 8.3: BLASR baseline WER for the `s3` speakers, triphone acoustic models.

8.2.2 The BLASR reference recognizer

The vocabulary of the test sentences was the 5k closed WSJ vocabulary. The reference 5k lexicon was generated with the Bell Labs Text-to-Speech system [110]. This lexicon has one canonical baseform for each word. Including general purpose variants was not considered important since results in chapter 7 showed that the variants in the general purpose CMU lexicon gave no benefit for non-native speech. For the test set a closed trigram language model for the 5k vocabulary was used. A reference recognizer with 12 Mel frequency cepstral coefficients plus log-energy term and their first and second derivatives was trained on 84 native speakers (WSJ0 SI-84). Phonetic decision tree state tying was used to build cross-word triphone HMMs with an average of 11 Gaussian pdfs per state [92]. This recognizer gave the baseline result of 29.2% word error rate (WER), see table 8.3 for individual WER. A 95% confidence interval for the test on all 10 speakers is [28.2%, 30.3%], see section 2.5.

In table 8.3 we have also included the bigram baseline results for comparison with the results in chapter 7. The performance using cross-word triphones and canonical baseforms for the HTK trained system and CMU lexicon was 27.7% WER (table 7.4). This result was obtained using HMMs trained on SI-284, i.e. about 66 hours of training data compared with the about 14 hours from SI-84 used in this chapter. We also note that for this read non-native speech the more complex trigram language model was beneficial.

The acoustic models were not retrained, i.e. the same models were used for both segmenting the adaptation sentences by forced alignment, phone loop transcription, and testing. This is done deliberately, because the pronunciation modelling will tailor the lexicon to the present acoustic models. If for any reason the acoustic models are retrained, we may have to regenerate the rules.

8.2.3 Rule derivation

Alternative transcriptions

We have used word segments for the transcription and alignment used to derive rules. The reason is that we want to maintain timing information to compare transcriptions belonging to the same acoustic data. The drawback is that our experiments then are restricted to word internal rules. However, studies of non-native speech imply that non-native speakers have less coarticulation between words [81] and there should be less need for cross-word rules. The reference recognizer was used to segment the adaptation sentences in word segments by forced alignment. The segmentation was manually inspected for some samples and perceived to be quite accurate in spite of the mismatch between the acoustic models and the non-native speech.

The reference lexicon had only one baseform per word so the reference transcription could be made without using the recognizer. For the alternative transcription we have chosen to use a phone loop grammar as in [54]. This will give many transcription "errors", but will not be restricted by the reference transcription, except via the acoustic models. Some of the differences between the reference and alternative transcriptions, i.e. the "errors", are actually the variation we want to model. As the alignment was based on association strength, see section 6.3.2, this will help us filter out non-systematic errors and reveal the systematic differences that we believe are due to pronunciation variation.

The phone-loop based alternative transcription was performed with the same triphone acoustic models as used in recognition (testing) later. To control the phone loop performance we experimented with the insertion and duration penalties at the HMM level. We chose the settings that gave the lowest phone error rate (PER). This might not be the best measure as we assume there will be mismatch between the reference and alternative transcriptions due to pronunciation variation, but we found no better alternative as discussed in section 6.3.1. As we can see from table 8.4 we have more deletions than insertions. This is consistent with other results, for example the experiments referred in [18], where the rules introduced 1.86% insertions, 4.85% deletions and 10.13% substitutions. We also note that the performance is worse for

	PER	*Sub.*	*Del.*	*Ins.*
Non-native sentences	47.0	31.0	12.7	3.3
Non-native words	55.8	36.8	15.7	3.3
Non-native words adapted	39.4	23.5	7.1	8.8
Native words	45.5	18.9	13.5	13.1

Table 8.4: Phone error rate (PER) on the non-native adaptation set and a native set for the chosen parameter setting.

words than sentences. This may be due to the acoustic models being trained as cross-word triphones and that the transcription of isolated word segments therefore suffers from the lack of context in the words. Less confidence should be given to the changes in the border phones, and they are therefore not used as the focus of the rules reported.

Using speaker adapted models we should be able to capture more lexical pronunciation variations, since the allophonic mismatch between the recognizer and the non-native speech will be smaller and the quality of the alternative transcription better. For speaker dependent lexica this could be a useful approach, end re-adapting the models using the new transcriptions could also be beneficial. One pitfall is that in this way we may hide variation better modelled in the lexicon. The best way to avoid this is to modify the lexicon first, then adapt models according to the more suitable transcription and hopefully achieve more consistent models. Using speaker adapted models to adapt a speaker independent lexicon will give a mismatch between the models used for rule generation and for recognition using the new lexicon. No speaker adaptation was performed for the experiments reported in this chapter.

The phone loop transcription was performed with 5-best recognition to get more transcriptions. As we have a limited amount of data this is favourable. The phones that are different in the N-best transcriptions will be the ones with lower likelihood and we would like to put less emphasis on them. An example of this is the 5-best phone loop transcriptions for the word "numerous":[1]

```
[n ow m aa r eh z].
[n ow m aa r ih z]
```

[1]We have used the CMU phonetic alphabet for the transcriptions, see appendix A

[n ow m aa r z]
[n ow m er z]
[n ow m eh r z]
We can be more confident about the first part of this word than the last (the reference transcription is [n uw m er ax s]). The difference between two N-best transcriptions is typically a change in only one or a few phones. This is independent of the total number of phones in the segment so a set of N-best transcriptions of sentence segments will also differ in only a few phones. A sentence level N-best transcription will give little extra information compared with one (first-best) transcription since most word transcriptions probably will not change. This is also one reason to use word segments rather than sentences when generating alternative transcriptions.

Alignment

We experimented with several dynamic programming based schemes. The differences lie in the way the costs for phone-to-phone mappings are derived, as explained in section 6.3.2. The results for the different alignment methods are presented in chapter 10. The three costs investigated were: uniform cost, phonological cost, and the novel method based on statistical co-occurrence of phones, the association strength. The association strength based costs gave most promising results, see chapter 10, and all the results in this chapter were derived using this method.

The threshold in the association algorithm (as explained in section 6.3.2) was set according to the association strength for the phone-to-phone mapping with the highest number of occurrences. In this experiment this was the identity mapping of the phone [ax].

Context-dependent phone-to-phone mappings: Rules

From the alignment we derive context-dependent phone-to-phone mappings, i.e. rules. As we have more deletions than insertions (i.e. the reference transcription contains more phones than the alternative), see table 8.4, the cost in the dynamic programming is set higher for an insertion than for a deletion. According to our rule notation, see section 6.3.3, a typical rule is x1-A+x2 $\rightarrow$ B where (A, B) is the transformation with A as the focus and B as the output of the rule. In this notation x1-A+x2 is called the *rule condition*. A, x1, and x2 are all one phone, whereas B can be a single phone (substitution of A with B), several phones (insertion/substitution), or DELETED (deletion of A).

Since we have small amounts of data in our experiments, we have chosen to use only one preceding and one succeeding phone as context for the phone-

to-phone rules. This is discussed in section 5.4.3. In [94] the immediate phone neighbours are shown to be most important in pronunciation modelling. A rule condition then consists of 3 phones. The different 3-phone segments present in the adaptation set will tell us which variation we can model. The BLASR test lexicon contains 4376 different such segments, whereas the baseforms for the words in the adaptation set only contain 871 3-phone segments. This means that we are unable to model the majority of the possible 3-phone segments. The situation is not as bad as these figures indicate; the distribution of the 3-phone segments is far from uniform. 3467 of the words in the 5k word test lexicon contain at least one of the 3-phone segments present in the adaptation set.

Rule pruning

For the non-native task we tried several of the methods for assessing rules described in section 6.3.4. The most usual way to assess pronunciation rules is an estimation of rule probability. This estimate is based on counting occurrences. The equations are repeated here for the convenience of the reader:

$$\mathrm{RPR1} = \widehat{P}(\texttt{x1-A+x2} \rightarrow \texttt{B}) = \frac{\mathrm{count}(\texttt{x1-A+x2} \rightarrow \texttt{B})}{\mathrm{count}(\texttt{x1-A+x2})} \tag{8.1}$$

$$\mathrm{RPR2} = \widehat{P}(\texttt{x1-A+x2} \rightarrow \texttt{B}) = \frac{\mathrm{count}(\texttt{x1-A+x2} \rightarrow \texttt{x1-B+x2})}{\mathrm{count}(\texttt{x1-A+x2})} \tag{8.2}$$

The phone error rate comparing the reference and alternative transcriptions is about 50%. Not all this difference will be variation in pronunciation, but the alternative transcriptions will contain many transcription errors. This may favour RPR2 over RPR1. On the other hand RPR2 will give fewer occurrences that survive the filtering.

Our main results were achieved with a log likelihood based rule pruning as described in section 6.3.4. The equations for log likelihood rule pruning are repeated here:

$$\begin{aligned} \mathcal{LL}(x_i) &= \log\left[\frac{p(x_i|B_i^{\mathrm{alt}},\theta)}{p(x_i|B_i^{\mathrm{ref}},\theta)}\right] \\ &= \log[p(x_i|B_i^{\mathrm{alt}},\theta)] - \log[p(x_i|B_i^{\mathrm{ref}},\theta)] \end{aligned} \tag{8.3}$$

Here, B_i^{ref} is the reference baseform for the word belonging to x_i, and B_i^{alt} is the alternative baseform for the same word after modification according to the rule we want to assess. θ is the set of parameters of the recognizer used in computing the log likelihoods. For each rule, we combine the positive

contributions from the words affected to compute an improvement measure $\mathcal{LLH}$:

$$\mathcal{LLH}(\text{rule } k) = \sum_{W_j} \sum_{k(j) \in K(j)} \mathcal{LL}(x_{k(j)}) \tag{8.4}$$

Here, $x_{k(j)}$ is an acoustic segment for a word W_j affected by rule k. For each word W_j we sum over all acoustic segments $K(j)$ where $\mathcal{LL}(x_{k(j)}) > 0$. For each rule we sum over all affected words.

We have also done some experiments with log likelihood assessment of the baseform variants as described in section 6.2.

Combining pronunciation rules to derive pronunciation variants

We have chosen to use only one rule at a time when generating the new pronunciations to test only the most important variants. Applying several rules simultaneously without an additional selection algorithm, will give many pronunciation variants for long words and increase the confusability, see section 5.4.5. The rule hierarchy used in [18] and [129] is one way of treating this that also incorporates a method to choose among rules with the same rule condition. The use of a rule hierarchy was is not investigated in this dissertation.

For all pronunciation rule experiments the canonical reference baseform is kept, since the amount of data used in rule derivation is limited as discussed in section 5.4.5.

Confusability reduction

Many pronunciation variants in the lexicon may increase the confusability and thereby the error rate. Adding variants also slows down recognition, and similar rules may model the same pronunciation variation and thereby add superfluous complexity. Rule pruning using probability estimates limits the number of pronunciation variants and control the confusability implicitly.

Because rule conditions that occur often will get a relatively higher score in our improvement measure $\mathcal{LLH}$, equation (8.4), we may get a lot of confusable rules, i.e. rules with the same rule condition, but transformation into different phone(s). These rules may be modelling the same variation and add superfluous complexity and confusability. An example is the rules `sh-ax+n` $\rightarrow$ `ah` and `sh-ax+n` $\rightarrow$ `eh`, where the actual baseform often seems to be somewhere in-between `[ah]` and `[eh]`. We have applied a confusability reduction approach by restricting the rule set to consist of at most one rule per rule condition. We retained the rule with highest log likelihood improvement $\mathcal{LLH}$. (In the example this is `[ah]`.)

As a lower WER is the final goal, confusability reduction should be incorporated more explicitly in the rule pruning measure. This means that we need an assessment of the number of errors corrected compared to the new errors introduced, see section 6.4. One obvious confusability measure is therefore recognition results on the adaptation set. Both the error rate and the smoothed error count described in section 6.4 were investigated.

As we have quite different vocabularies in the adaptation and test sets for the WSJ non-native task we cannot use the same language model for the two tests. Without a language model the tests will not be completely comparable. We therefore tried a log likelihood measure that was not dependent on the language model, but still considering the sentence level. This was done by using forced alignment of the known word string and adding log likelihood increases over baseline. In all the experiments reported the reference baseform was kept and we will therefore never encounter a decrease in log likelihood.

8.3 Results

Some of the experiments reported in this section were also presented in [4] and [5]. All results shown are obtained on the test set. The baseline result on the test set was 29.2% WER on average, see table 8.3. As the different speakers have very different WER, the results in the figures are shown as improvement relative to the baseline WER.

8.3.1 Individual pronunciation rules

We first derived individual lexica, since the speakers have different mother tongues we wanted to investigate individual lexicon adaptation.

In figure 8.1 the results for different thresholds for RPR1 are shown. For the best result, RPR1 $> 30\%$, the average WER was 28.2%. The three speakers with deterioration had French (4ne and 4nf) and British (New Zealand) (4nn) origin. The speaker with no improvement (4ni) had the lowest baseline WER, 11.4% and was Danish. We saw an improvement for all the 5 female speakers. A lower threshold on the rule probability resulted in more rules. We observed that the improvements as well as the deteriorations increase when we use more rules. The average result was not affected by the increase of the number of rules, but the best result was obtained with a modest number of rules. For RPR1 $> 30\%$ the number of rules varied from 107 to 147 for each speaker.

Using equation (8.2) and threshold RPR2 $> 20\%$ gave an overall error rate of 28.5%. In figure 8.2 the results for RPR1 are compared with RPR2.

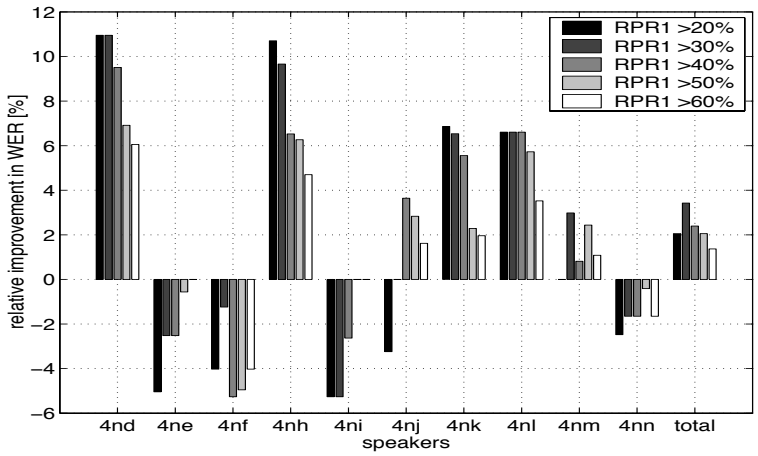

Figure 8.1: Relative improvement in WER for different RPR1.

As we can see the rules that were derived from both sorting criteria gave no deterioration for any speaker[2].

Sorting by rule condition occurrence in the test lexicon and retaining the top 30 rules for each speaker gave a WER of 28.6%. Sorting by word frequency computed from the language model training text gave similar results.

8.3.2 Individual maximum likelihood baseforms for words

For frequently occurring words it may be beneficial to model the baseform for the whole word. One of the arguments favouring rules over direct modelling is that the rules depend on smaller segments than words and will occur more often, giving more reliable estimates. For frequently occurring words, e.g. function words, this argument is not longer valid. The generalization argument favouring rules is still true, but we will only be able to model pronunciation variation for seen words. One possibility is to use both techniques to model different effects.

The few repetitions of each word per speaker means that a ML optimization on the word level as described in [51] and section 6.2, was infeasible for most words. In an initial experiment we used this technique for one of the speakers (4nd), with the baseline result of 34.7% WER, see table 8.3. The 95% confidence interval for this WER is [31.3%, 38.0%]%.

We found speaker dependent baseforms for the 23 words that occurred

[2]This was only tested without iterating the association algorithm.

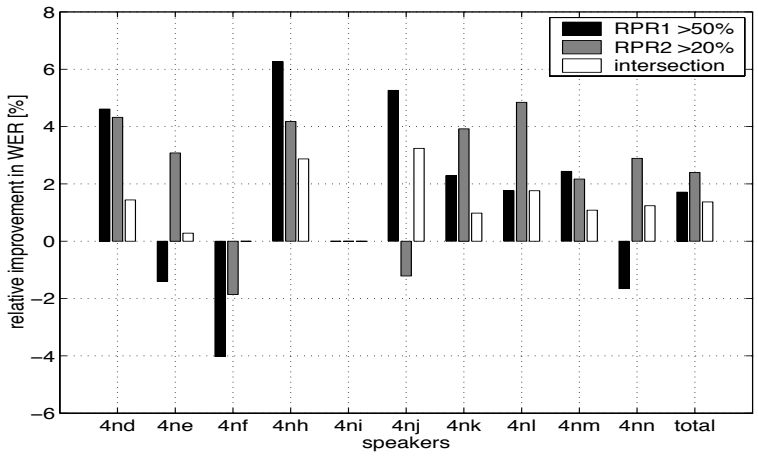

Figure 8.2: Relative improvement in WER for RPR1 and RPR2.

more than 3 times for this speaker. Since we used a 5-best loop, this means we had at least 5 different candidates for each acoustic segment. For 7 of the words the ML baseform was the same as in the reference lexicon. For the remaining 16 words we substituted the reference baseform with the best ML baseform. This gave a WER of 36.1%. We then added the best ML baseform keeping the reference baseform for these 16 words. This gave a WER of 31.9%. We also tried adding several ML baseforms. This gave only minor differences when we kept the reference baseform.

The results for all speakers were not encouraging: A WER of 28.9% when adding the best ML baseform for each speaker. In figure 8.3 we see, however, that for most speakers the ML baseforms gave a better performance. The average is highly affected by the deterioration seen for one speaker (4nj). In this figure we also observe that the distribution of errors and corrections was different for the speakers compared with the rule based lexicon modifications. This indicates that we model different variation with the two approaches and that a combination can be beneficial.

Combining rules and ML baseforms

The 23 frequent words for which we derive ML baseforms are short, and we might see a joint effect in that they are frequent enough to be reasonably modelled and that these short segments give a more reliable phone loop transcription. Since the ML baseforms are short, many of the corresponding words are not affected by our word internal rules that demand at least three phones

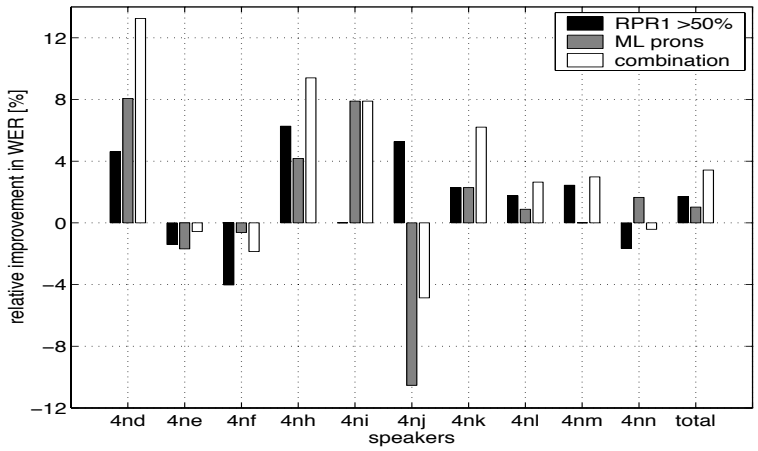

Figure 8.3: Relative improvement in WER for RPR1 and ML.

in the rule condition. We therefore tried to add the ML baseforms to one of the rule modified lexica to see if the two pronunciation modelling approaches together gave increased performance. For speaker 4nd the ML baseforms gave a WER of 31.9% compared with the baseline of 34.7%, whereas the rule based lexicon with estimated rule probability RPR1 >50% gave a WER of 33.1%. Combining the two lexica gave a WER of 30.1%.

We did not achieve the same improvement for the rest of the speakers, only 28.2% WER compared with the RPR1-result of 28.7% on average[3]. There was still an improvement for most speakers and the effects seem to be additive. Even for the speaker (4nj) where we observed a large deterioration using the ML baseforms, the combined effect was better since the rule based variants gave an improvement. The problem is how to combine the ML rules for the different speakers. Computing the true ML baseforms for all speakers combined will be computationally quite heavy. We did therefore not perform such experiments and do not know if the effect will be the same as the individual ML baseform results.

8.3.3 Common pronunciation rules for all speakers

We have examined three approaches for deriving rules in order to make a common lexicon for all speakers:

1. Derive association strength and alignment for each speaker individually

[3]This was only tested without iterating the association algorithm.

to derive individual rules. Then we merge these rules to a common rule set (merged rule probability, called MRPR)

2. Derive association strength and alignment for each speaker individually, but derive rules for all speakers simultaneously (combined rule probability, called CRPR)

3. Derive association strength, alignment, and rules for all speakers simultaneously (joint rule probability, called JRPR)

The variation between speakers is large since the speakers have different mother tongues, and it seems reasonable to find rules for each speaker separately and later merge them to make a common lexicon. We wanted to examine the effect of using the association algorithm both on the individual speakers and on all speakers simultaneously. All the experiments for common rule derivations use association strengths after 4 iterations.

To merge the individual rules according to approach 1, we added the counts from each speaker to find a joint rule probability. We used individual rules with RPR1 $> 50\%$ and retained the rules with estimated merged rule probability over a certain threshold. MRPR computed according to equation (8.1) is called MRPR1. For the lexicon with merged rule probability MRPR1 $> 50\%$, 1559 of the 4986 words in the reference lexicon were affected by the rules, and the average number of baseforms per word (BPW) was 1.44. In table 8.5 results for the different joint rule probability thresholds are shown. We see that the best merged lexicon gave the same improvement over baseline as the individual lexica. A lower threshold on individual RPR1 gave less improvement. Using the RPR2 scheme did not give the same improvement for the common lexica compared with individual lexica. The best performance in this case was 28.6% WER.

MRPR1	*No. rules*	*BPW*	*WER*	*Rel. improvement*
$> 30\%$	169	1.61	28.4%	2.7%
$> 40\%$	137	1.49	28.2%	3.4%
$> 50\%$	114	1.44	28.6%	2.1%

Table 8.5: Results for merged rules using different thresholds on the estimated merged rule probability MRPR1.

Simultaneously generating rules for all of the speakers from individual association strength derivation and alignment resulted in mostly low probability rules. As the rules are derived using more data, it is reasonable to used a

lower threshold for the rule probability than when deriving rules individually. Using a lower threshold gave an insignificantly increased performance over the merged rules, see table 8.6. Setting the threshold for the combined rule probability CRPR1 to $> 20\%$ the lexicon became quite large and the recognition was slow.

CRPR1	*No. rules*	*BPW*	*WER*	*Rel. improvement*
$> 20\%$	413	1.84	28.1%	3.8%
$> 30\%$	152	1.30	28.5%	2.4%

Table 8.6: Results for rules derived for all speakers simultaneously, but with individual association strength derivation, using different thresholds on the estimated joint rule probability CRPR1.

Approach 3, generating rules for all of the speakers simultaneously, and also deriving association strength for all speakers simultaneously, gave results similar to approach 2, see table 8.7. The results were slightly better than the individual lexica and the merged lexicon. As the association derivation relies on statistical methods, the individual association derivation may suffer from scarce data. This can explain why the association performed for all speakers simultaneously gave better result. Using the RPR2 scheme gave a WER of 29.0% when generating the rules simultaneously, i.e. hardly any improvement over baseline.

JRPR1	*No. rules*	*BPW*	*WER*	*Rel. improvement*
$> 20\%$	423	1.85	28.0%	4.1%
$> 30\%$	168	1.34	28.4%	2.7%

Table 8.7: Results for rules derived for all speakers simultaneously using different thresholds on the estimated joint rule probability JRPR1.

Relative improvements in WER per speaker for the best lexicon for all three schemes are shown in figure 8.4. We get more deterioration for one of the speakers using lexica based on common rule derivation than merging individual rules. One other advantage with the merged lexicon is that the performance was similar using fewer rules, 137 compared with 423, which is favourable for recognition speed. Testing on native speakers we get a deterioration from 7.8% to 8.3% WER for the merged lexicon shown in figure 8.4. For both lexica based on simultaneous modelling of all speakers, the native result was 8.5% WER. The merged lexicon is thus not only smaller but has less confusability

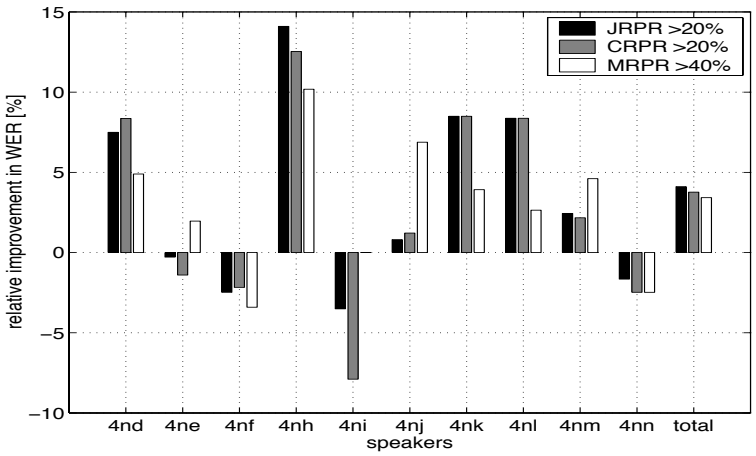

Figure 8.4: Relative improvement in WER for common lexica using different rule pruning schemes.

when measured on native speakers.

A modified rule probability scheme was tried using the top 30 rules for each speaker after sorting the rules by rule condition probability. These $40 \cdot 30 = 1200$ rules were then merged by adding the counts. Using a joint RPR1> 50% we got 82 rules, a BPW of 1.41, and WER 28.4%. We achieved equal performance with a smaller lexicon, showing that sorting by rule condition probability retained the most useful rules[4].

8.3.4 Retranscription

To perform retranscription we used the rules to form a grammar for each word in the adaptation set and performed a 5-best forced alignment. The border phones were not changed since we do not have border phone rules. In these experiments several rules could be used in the same baseform (i.e. a huge number of alternatives for long words). This approach was chosen because we did not want to place tight restrictions on the choice of rules to use in the retranscription in order to let the acoustic models still have large influence. All the rules with individual RPR1 $\geqslant 20$ as well as the simultaneously derived rules with RPR1 $\geqslant 20$ were used in the retranscription[5]. We also included rules

[4]This was only tested without iterating the association algorithm.

[5]Since we have a tighter restriction on which 3-phone segments that we are able to model using individual rule derivation, some rules can only be derived from simultaneous rule derivation.

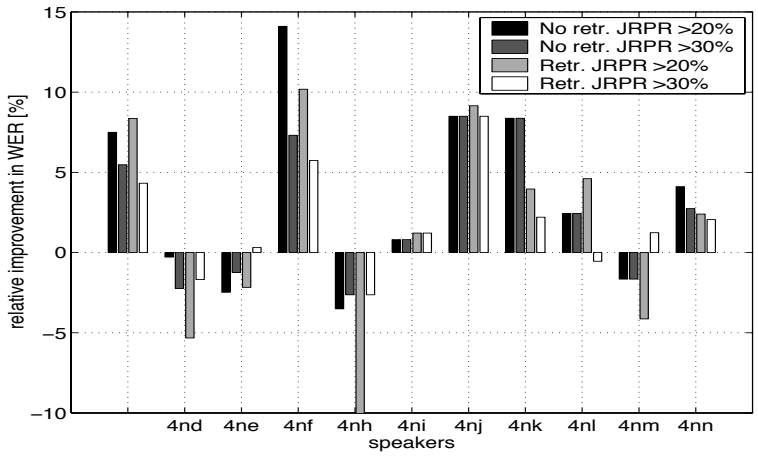

Figure 8.5: Relative improvement in WER with and without retranscription.

from alignment both with and without iterations in the association strength. This gave a total of 1883 rules used in the retranscription.

In figure 8.5 we compare results using lexica derived for all speakers simultaneously with and without retranscription. The number of rules for the two retranscription experiments were 411 for JRPR1 $>$ 20% and 237 for JRPR1 $>$ 30%. The corresponding BPWs were 1.93 and 1.39. As shown in the figure we got a deterioration compared with the non-retranscription experiments in this case. We did not observe major improvements for any setup compared to the non-retranscription experiments. Retranscription gives no new rules, the purpose of retranscription is rather to get better alternative transcriptions used for alignment and rule derivation. We have no good explanation why there is no increased performance in this case. One explanation may be that by using the association strength alignment we have already discarded random errors in the transcriptions, and that no additional information is gained by retranscription.

8.3.5 Log likelihood based rule pruning

In the first experiment using log likelihood we removed the rules that gave no improvement on the adaptation set, i.e. $\mathcal{LLH}(\text{rule } k) = 0$. The number of rules was reduced, but we observed no reduction in WER. Setting the threshold $\mathcal{LLH}(\text{rule } k) = 1$ in the same experiment resulted in even fewer rules and still unchanged WER.

We then selected a subset of rules first using rule probability pruning and

then sorted by log likelihood afterwards. This gave equivalent or better results than using only rule probability based selection.

Further, we used only the log likelihood improvement measure to find the best combined rules from the complete set of initial rules. Considering the performance of the log likelihood measure only we selected all the rules with individual RPR1 $\geqslant$ 20 as well as the simultaneously derived rules with RPR1 $\geqslant$ 20 (1883 rules in total, the same rules as used in the retranscription). Results from different selection of top rules sorted by the threshold $\mathcal{LLH}$ are shown in table 8.8. When including low probability rules like this, we get more rules to choose from and more confusable rules among the top rules sorted by $\mathcal{LLH}$ than when combining the rule probability and log likelihood pruning.

No. rules	*BPW*	*WER*	*Rel. improvement*
100	1.50	28.7%	1.7%
150	1.66	28.6%	2.1%
200	1.82	28.3%	3.1%

Table 8.8: Results for top rules sorted by log likelihood without any confusability reduction.

8.3.6 Simple log likelihood based confusability reduction

To reduce the confusability we restricted the rule set to contain at most one rule per rule condition selected by the log likelihood improvement measure $\mathcal{LLH}$. Low probability rules now perform equally well as higher probability rules, showing that for rule pruning the log likelihood improvement measure can be used without the added restriction of rule probability. Results for confusability reduction are shown in figure 8.6. The best result was obtained using 150 rules giving a WER of 27.3%. The confusability reduced rule set of 50 rules corresponds approximately to 100 non-reduced rules and the rule set of 100 corresponds to approximately 250 non-reduced rules[6]. In figure 8.7 the results for the confusability reduced lexica are shown as a function of baseforms per word. The complexity of the recognizer (i.e. recognition time and memory needed) will depend strongly on BPW.

Equivalent or better results were achieved using fewer rules compared to the rule probability approach. In table 8.9 we have compared different lexica tested on the usual non-native speakers as well as on *native* speakers (WSJ

[6]Correspond = Inspecting which "unique" rules that are still present when removing the confusable ones.

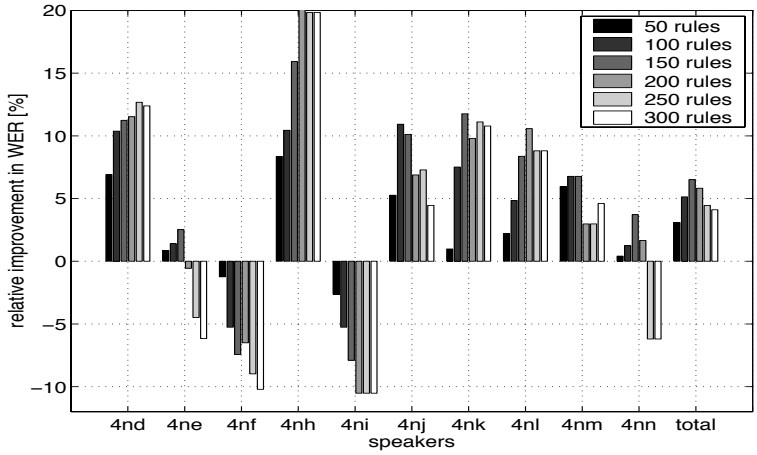

Figure 8.6: Relative improvement in WER for different number of rules using log likelihood rule pruning and confusability reduction

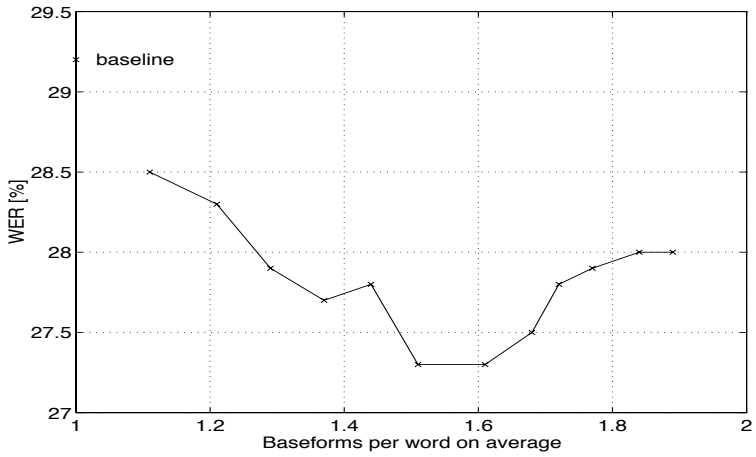

Figure 8.7: WER for log likelihood lexica as a function of BPW

h2). The baseline for the native test was 7.8% WER, see section 7.4. As we can see there is a large gap in the performance between the native and the non-native speakers. This is due to the diversity among the speakers, which affect both the acoustic models and the lexicon. The log likelihood based lexica gave slightly less deterioration than rule probability based lexica on the native speech.

Rule pruning method	*No. rules*	*BPW*	*Non-native WER*	*Native WER*
JRPR> 20%	423	1.85	28.0%	8.5%
JRPR> 30%	168	1.34	28.4%	8.0%
MRPR> 40%	137	1.49	28.2%	8.3%
LLH	50	1.21	28.3%	7.9%
LLH	100	1.37	27.7%	8.0%
LLH	150	1.51	27.3%	8.1%
LLH	200	1.68	27.5%	8.3%

Table 8.9: Results for comparison of rule pruning methods. JRPR = joint rule probability and MRPR = merged rule probability

8.3.7 Other confusability reduction experiments

A recognition based measure to assess complete lexica was difficult to use because the adaptation set contained several words not present in the test set and therefore not in the language model. For recognition tests on the adaptation set we therefore used a word-loop. The word insertion penalty was adjusted to get the best baseline WER on the adaptation set. In this way we could test continuous speech and not only isolated words. Recognition on the adaptation data was not consistent with tests on the test set, probably because of the lack of language model. To avoid the need for a language model, we used forced alignment in the next experiment. These results were not consistent with test on the test set either.

To assess each rule from the log likelihood pruned rule list, we performed experiments based on error counting. The results were counterintuitive, i.e. tests on adaptation and test set were not consistent. We looked closer at a subset of rules that occurred in words that were corrected in the test set. Since we are working with continuous speech the interpretation of the results must be done with care. The differences will not affect one word isolated. Some of the rules investigated resulted in baseforms that gave higher error rates on

the adaptation set because they introduced more errors than they corrected. Our test and adaptation set contain quite different vocabularies and this may be one of the reasons, since an error counting based measure will not take into account the unseen words. More study is warranted for a deeper understanding.

Four lexica were chosen for further evaluation using the confusability measures from section 6.1, see table 8.10. A "smoothed error" was computed using equation (6.19) with $\eta = 2$. This gives a comparison of the log likelihood of the correct word with a smoothed average instead of the worst competitor as in ordinary WER computations. An "estimated loss" was computed using a sigmoid as in equation (6.20) with $\xi = 2$. Preliminary testing to adjust the values of η and ξ, gave no large differences in relative loss.

The results for the chosen confusability measures are given in table 8.11. Even if the MRPR-lexicon had similar WER on the test set compared with the "LLH-50"-lexicon, it gave a much lower relative score on the smoothed error count and the estimated loss. For the log likelihood based lexica the comparison of the confusability measures gave too high score (relative improvement) compared with the "LLH-50"-lexicon. Comparison between the LLH-lexica gave results consistent with WER in the test set. Apparently the confusability measures gave too high punishment on the many rules in the MRPR scheme compared with the true performance on the test set.

Lexicon	*No. rules*	*BPW*	*WER*	
			Absolute	*Relative*
Reference	-	1.00	29.2%	-
MRPR	137	1.49	28.2%	3.4%
LLH	50	1.21	28.3%	3.2%
LLH	150	1.51	27.3%	6.5%
LLH	250	1.77	27.9%	4.6%

Table 8.10: Results on test set for the lexica to be compared by confusability measures

8.4 Discussion

For individual lexicon adaptation adding variants from a rule probability pruning scheme gave a modest improvement, on average a WER of 28.2% compared with the baseline result of 29.2%. The improvement was very variable for the different speakers, but we observed an improvement for most

Lexicon	*WER*		*Smoothed error*		*Estimated loss*	
	Absolute	*Relative*	*Absolute*	*Relative*	*Absolute*	*Relative*
Reference	27.1%	-	7.5%	-	8.3%	-
MRPR	26.9%	0.9%	7.4%	1.3%	8.2%	0.8%
LLH	26.4%	2.6%	7.2%	4.8%	7.9%	4.1%
LLH	26.0%	4.1%	7.2%	4.8%	7.9%	4.4%
LLH	26.1%	3.7%	7.3%	3.3%	8.0%	3.1%

Table 8.11: Results on adaptation set for confusability measures

speakers.

Individual maximum likelihood pronunciation variants were derived to the 23 words that occurred sufficiently often in the adaptation set. The effect on the performance varied even more between the speakers than the rule based pronunciation modelling. Combining the two approaches gave better results than using each method separately.

We observed that common lexica generated by merging individual rules performed similar to generating rules for all speakers at the same time, with a WER of 28.2% compared with the baseline result of 29.2%. The merged lexicon scheme gave fewer rules and thus fewer baseform per word on average and faster recognition than generating rules for all speakers simultaneously. Testing on native speakers revealed a lower confusability in the merged lexicon.

It was somewhat surprising that the joint pronunciation modelling of the non-native speakers gave results similar to individual pronunciation modelling. Other experiments have shown that individual lexica outperform one common lexicon even for non-native speakers with the same mother tongue [41]. In our experiments the increased confusability using rules from several speakers in the same lexicon was probably outweighed by the higher confidence in the rule selection as we use more data. Even if the speakers have different mother tongues, similar variation may be present, and this may be the reason why we get improvements by modelling all speakers in the same lexicon. Speaker dependent pronunciation modelling using only 40 utterances was reported in [54]. For the joint speaker pronunciation modelling in this experiment about 500 utterances were sufficient when the new baseforms were merged with the reference baseforms. All were US English speakers and thus a more homogeneous set, and not surprisingly the joint modelling of all speakers gave a better result.

We have compared a new log likelihood based rule pruning measure with more traditional rule probability based measures. Using acoustic log likelihood

as a rule pruning measure we achieved equivalent or better performance with fewer rules. We also achieved less deterioration for native speakers using the same lexicon, i.e. less confusability. Lexica with fewer rules are favourable, as more rules give more baseforms in the lexicon and slows down recognition. Thus log likelihood based rule pruning is a better way of combining individual rules to generate a joint lexicon. Adding a confusability reduction, we achieved the same result with even fewer rules. We achieved the best result using a 150 rule lexicon which gave a WER of 27.3% compared with the baseline WER of 29.2%.

Controlling the confusability when adding baseform variants to the ASR lexicon is important. In addition to the confusability reduction at rule level, we wanted to test confusability measures to compare entire lexica. The measures gave results inconsistent with tests on the test set. One reason may be that the vocabulary of the adaptation set was very limited and different from the test set. The confusability measure based on performance on the word level without incorporating a language model, may also be a reason for the inconsistency.

8.5 Summary of the main results

In this chapter we have presented several completely data-driven approaches to modelling both individual and common lexica for a group of non-native speakers with different mother tongues.

- For individual lexicon adaptation we observed a combined effect of using maximum likelihood pronunciation variant modelling and pronunciation rule modelling.
- For common lexica a scheme merging the rule probabilities derived for each speaker performed better than a joint rule probability derivation. The common lexica performed better on average than individually derived lexica.
- Acoustic log likelihood based rule pruning performed better than rule probability based pruning.
- Confusability measures for comparing lexica is a difficult task when the adaptation and test set differ strongly in vocabulary.

Chapter 9

Data-driven pronunciation modelling for spontaneous speech

9.1 Introduction

Spontaneous speech is a difficult task for ASR systems, but must be handled to achieve more natural user interfaces based on speech technology. In chapter 7 we saw that the performance for spontaneous dictation was substantially worse than for read speech, even if dictation is a rather restricted form of spontaneous speech compared with conversational speech. Speaker dependent acoustic model adaptation gave some improvement, but relatively less than for read speech. In this chapter we therefore investigate some of the data-driven pronunciation modelling techniques presented in chapter 6 for the spontaneous dictation task.

9.2 Experimental procedure

9.2.1 The database

The pronunciation modelling methods were applied to the spontaneous dictation part `s9` of the Wall Street Journal (WSJ). More information on the WSJ database is given in section 7.2.1. We have used the spontaneous dictation training set for lexicon adaptation. This set consists of 4000 utterances spoken by 20 journalists, see table 9.1. All speakers of both the adaptation and test sets were native US English speakers, and there was no overlap in speakers between the adaptation and test set. 15 of the adaptation utterances had

	Adaptation	*Test*
Speakers	20	10
Utterances	3483	200
Words	71732	4633

Table 9.1: Distribution of adaptation and test set for spontaneous dictation.

no transcription and were removed, while 502 other utterances were removed because they contained words not present in the CMU lexicon. The remaining 3483 adaptation utterances contained about 300 000 phones.

9.2.2 The HTK reference recognizer

The vocabulary of the test set for `s9` was the 20k open WSJ vocabulary with verbal punctuation (vp). We have trained a SI-284 baseline recognizer using fairly standard methods [126], but using the CMU lexicon. The baseline is described in chapter 7. We chose to use the canonical CMU lexicon and canonically trained triphone models, table 7.4, since the CMU variants gave no improvement and it was detrimental to include more variants in training than in test. For all the experiments reported we added variants derived during the pronunciation modelling so the reference canonical baseform was always retained. The reference recognizer gave the baseline result of 23.7% WER, results per speaker are shown in table 9.2. The reference recognizer was also tested on a subset of the adaptation set. 400 utterances were extracted randomly from the adaptation set and gave a result of 19.5% WER.

We used the same acoustic models in test and to find the alternative transcriptions, the same approach as for the experiments on non-native speech in chapter 8.

9.2.3 Rule derivation

Alternative transcriptions

The rule derivation was performed on the sentence level using 1-best only. One reason for this is that the size of the adaptation set available for this task was sufficient to avoid the need for N-best transcriptions, and we did not need the shorter segments to get variation between the N-best transcriptions.

Manual inspection of the transcription revealed very long silences between the words for some utterances. In the reference transcription we therefore used two kinds of word boundaries dependent on the silence model (context-

Speaker identity	*WER*
4p0	19.9%
4p1	26.8%
4p2	25.4%
4p3	14.0%
4p4	42.0%
4p5	26.3%
4p6	15.9%
4p7	24.0%
4p8	35.6%
4p9	23.4%
Average	23.7%

Table 9.2: Baseline WER for the `s9` speakers, triphone acoustic models and bigram language model.

	PER	*Sub.*	*Del.*	*Ins.*
Phone loop	27.9	16.2	8.3	3.4
Phone bigram	27.3	16.0	7.1	4.2

Table 9.3: Phone error rate (PER) on the adaptation set for the chosen parameter setting.

independent or context-free) chosen by the recognizer. The pause duration can then be used to decide whether we should model a coarticulation effect on the rule border. The pauses where the context-independent silence model was chosen should be ineligible for cross-word rules whereas the pauses were the context-free short pause model was chosen should be eligible for cross-word effects.

The alternative transcription was made by a phone loop transcription, as in chapter 8, but also using a phone bigram trained on the reference lexicon. This last approach will give a more restricted transcription and hopefully less transcription errors. Since the spontaneous dictation task had native speakers we assumed that the variation would follow the native phonotactical rules found in the lexicon. The insertion penalty was adjusted to give the lowest phone error rate (PER), see table 9.3.

We note that the PER is much lower than for the non-native task although

the baseline recognition WER result is similar. One reason is that we in the present experiment use sentence transcriptions. Preliminary tests using segmented words for the spontaneous task gave in the order of 40% PER. This was also one reason to chose sentence level transcriptions for the spontaneous dictation. We also note that the phone bigram and phone loop transcriptions obtained similar PER. About 5% of the phones differed between the two transcription types, half of them substitutions. One of the reasons may be that a phone bigram trained on the reference lexicon not will cover cross-word combinations.

Alignment

For the alignment we have used the association algorithm, as in chapter 8, but also the time synchronous alignment introduced in section 6.3.2.

The first association algorithm was based on a word segmentation. We modified the association algorithm to base the first mapping costs on a uniform alignment. This is an extension of the iterated version of the association strength where we based the iterated association strength on the previous association alignment. In this modified version we also included the possibility of learning mappings from phones to deletions and from insertions to deletions. Manual inspection of preliminary experiments using this alignment showed no improvement and we therefore continued using phone-independent costs for deletions and insertions.

Many of the sentences contain long pauses and restarts. There are two reasons why this is challenging for the alignment: 1) This kind of speech is difficult to label confidently and the reference transcription is more likely to contain errors compared with the acoustic content of the utterance. 2) The alternative transcriptions will often introduce spurious phones in long pauses. If we encounter a misalignment due to transcription errors, an alignment based on dynamic programming may propagate the error and no longer align labels belonging to the same acoustic segments. This problem is discussed in more detail in chapter 10.

Rule derivation

From the association strength alignment and time synchronous alignment we derived context-dependent phone-to-phone mappings as in chapter 8, i.e. rules with a rule condition length of 3 and a rule focus length of 1. From the time synchronous alignment we were also able to derive rules with longer focus. The context for these expanded rules was chosen to be one “stable” phone to each side. “Stable” means that the phone was identical in both the reference

and alternative transcription, and that the time borders were the same (one frame deviation was allowed). The minimum rule condition length was set to 3, and the maximum rule condition length was set to 7. There were no restrictions on the number of word borders as long as they were eligible for cross-word rules.

Rule pruning

We performed comparative tests using the rule probability pruning, using both the association based alignment and the time synchronous alignment, as well as the pruning based on acoustic log likelihood. Details on the pruning methods are given in section 6.3.4.

The time synchronous rule derivation also include a log likelihood measure for the rules. We used this measure to perform an initial pruning before assessing the rules using the log likelihood. For this assessment we used a word segmented version of the adaptation set.

Combining pronunciation rules to derive pronunciation variants

For the experiments we chose to use only word internal rules to validate the rules before proceeding to cross-word rules. Many of the rules that appeared across word borders contained phone combinations not present within words and were therefore discarded. Some rules may occur both within words and at word borders. For these rules we assume that the coarticulation effects will be similar across word border and within words, and we found it reasonable to use the variation observed across words also as word internal rules.

As for the pronunciation variant derivation in chapter 8 we used only one rule at the time to assess the most important variants.

9.3 Results

The association based alignment of the phone loop transcription and the reference transcription were used to derive context-dependent phone-to-phone mappings. These rules were then pruned using rule probability (a similar approach as for non-native speakers in chapter 8). The results on the test set are shown in table 9.4. The results were almost identical to the baseline. An error analysis revealed that the errors were approximately the same for the two systems. Even for the largest dictionary, RPR$>$ 20%, only about 6% of the errors differed.

Using the time synchronous alignment to derive the same type of rules, i.e. context-dependent phone-to-phone mappings, gave similar rules. In table

RPR thres.	*No. rules*	*BPW*	*WER*
RPR > 60%	67	1.03	23.7%
RPR > 40%	221	1.13	23.5%
RPR > 20%	710	1.70	23.5%

Table 9.4: Results in WER for rule probability pruning on association based, phone loop transcription alignment

9.5 the top ten rules for the rule probability pruning of association based alignment are shown with corresponding rank, both for the association based alignment and the time synchronous alignment. The rank was computed for all rules, but only rules that can be used as word internal rules for the 20k vocabulary are shown. The rule probability pruned rules from the time

	RPR-rank		
Rule	*Assoc loop*	*Time loop*	*Time phbig*
`hh-ah+w` → `ow`	1	1	1
`s-k+t` → `DELETED`	2	23	16
`ng-g+l` → `DELETED`	3	13	28
`ch-ah+w` → `DELETED`	4	174	159
`n-d+z` → `DELETED`	6	4	6
`f-t+s` → `DELETED`	9	11	8
`n-d+sh` → `DELETED`	11	14	29
`k-t+s` → `DELETED`	12	16	23
`f-t+w` → `DELETED`	13	195	178
`m-aa+t` → `ah`	16	53	11

Table 9.5: Top rules for rule probability pruning based on association strength and time synchronous alignment on a phone loop transcription and time synchronous alignment on a phone bigram transcription

synchronous alignment gave also results similar to the baseline, see table 9.6.

The phone bigram alternative transcription did not deviate much from the phone loop transcription. We performed a time synchronous alignment also for the phone bigram transcription, and in table 9.5 we see that the ranks were similar to the ranks from the phone loop transcription. As expected the results when testing on the test set were similar to the phone loop approach.

There were 706 rules from the time alignment with RPR $\geqslant$ 20%. These rules were sorted by the acoustic log likelihood measure. 106 of the rules were

RPR thres.	*No. rules*	*BPW*	*WER*
RPR > 60%	69	1.03	23.7%
RPR > 40%	228	1.13	23.6%
RPR > 20%	677	1.63	23.5%

Table 9.6: Results in WER for rule probability pruning on time synchronous alignment, phone loop transcription

only applicable across word borders and were therefore discarded. 233 rules had a log likelihood measure of zero, i.e. $\mathcal{LLH}(\text{rule } k) = 0$, and these rules were also discarded. Further we applied the confusability reduction by only retaining the best rule for each rule condition. This left 358 rules for testing. The sorting by log likelihood was quite different from rule probability sorting, see table 9.7 for the top ten log likelihood pruned rules. We note that the deletion rules still are most frequent. A scatter-plot of the two rule pruning

Rule	*LLH-rank*
`f-ao+r` → `DELETED`	1
`t-ah+d` → `ih`	2
`t-ah+n` → `DELETED`	3
`n-t+iy` → `DELETED`	4
`dh-ih+s` → `ah`	5
`w-ih+l` → `DELETED`	6
`y-ao+r` → `DELETED`	7
`n-t+ah` → `DELETED`	8
`dh-ae+n` → `ah`	9
`hh-ae+d` → `eh`	10

Table 9.7: Top rules for acoustic log likelihood pruning of rule based time synchronous alignment, phone loop transcription

measures shown in figure 9.1 also shows the low correlation between the top rules for the two measures. The results in table 9.8 showed again neither improvement nor deterioration compared with the baseline result. In this case the error analysis showed that only 4% of the errors differed from the baseline.

Using rules derived directly from the time synchronous alignment allowed longer rule focus. Sorting by the acoustic log likelihood measure the majority of the high likelihood rules had 3-phone rule conditions as the rules we derived

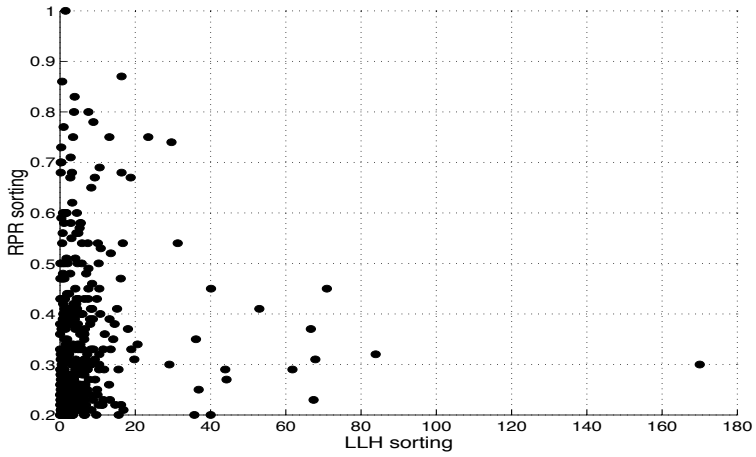

Figure 9.1: Scatter plot for the two rule pruning measures: rule probability and acoustic log likelihood.

No. rules	*BPW*	*WER*
100	1.27	23.8%
200	1.43	23.7%
300	1.54	23.6%
358	1.58	23.6%

Table 9.8: Results in WER for time synchronous alignment and log likelihood rule pruning, phone loop transcription

from the alignment. Since the log likelihood measure rewards frequently occurring segments this will favour short rules. Sorting by a relative log likelihood measure gave different top rules, but still a majority of 3-phone rule conditions. Preliminary tests using these rules gave no improvement.

No confusability measures were evaluated on this task as the lexica tested gave similar results and the error analysis showed only small differences in the recognized strings compared with the baseline system.

9.4 Discussion

The improvements seen using data-driven pronunciation rule modelling for the non-native task in chapter 8 did not occur for the spontaneous dictation task. Including variants based on the derived rules gave no improvement compared

with the baseline WER and only minor differences in the type of errors. In chapter 7 the variants from the CMU lexicon gave no improvement, but the errors differed. The rules we observed from the alignment of a reference and an alternative transcription were mostly deletion rules and allophonic variation. For this native task the variation was apparently already sufficiently modelled by the acoustic models. Other results on the same set using CART based pronunciation modelling showed only minor improvements [93].

For this task we have an adaptation set with the same vocabulary as the test set available in the WSJ corpus. A direct pronunciation modelling approach could therefore be beneficial for the "seen" words, possibly using the rules to generate candidate variants. Of the 7868 words in the adaptation set also in the 20k vocabulary, 1972 different words occurred at least 5 times.

In chapter 7 we saw that speaker dependent acoustic model adaptation gave a significant improvement, but the improvement was less than for the read speech case. Recognition of spontaneous speech is still a problem.

One of the major differences in spontaneous speech compared with read speech is that it contains many pauses, both silent and filled, restarts and other disfluencies. These types of speech segments as well as partial words may be better handled by specific disfluency modelling. The rate-of-speech varies both between speakers and within utterances, so a special rate-of-speech modelling may also be beneficial. Last, but not least, the language model of read speech is not appropriate for spontaneous speech with its many restarts and hesitations. An adaptation of the language model to spontaneous speech is one way of increasing the performance for this task.

9.5 Summary of the main results

- The data-driven pronunciation rule modelling gave neither improvement nor deterioration on the native spontaneous dictation task.
- The variation captured by the rules consisted of deletions and allophonic variation apparently sufficiently modelled by the acoustic models.
- The different rule pruning measures resulted in different rules, and no correlation was seen for the top rules. We observed, however, no difference in recognition performance.
- More sophisticated pronunciation modelling techniques may help more than the methods presented here. For the disfluencies and changes in rate-of-speech that are characteristic for spontaneous speech, pronunciation modelling may not be the right answer.

Chapter 10

Comparison of alignment methods

10.1 Introduction

In pronunciation rule derivation the most common method is to start with an alignment of two transcriptions, one symbolic and one surface form, where the latter is assumed to better represent the variation we want to model. Alignment of two transcriptions may also be used for other tasks like phone confusion matrix derivation. Some substitutions are more probable than others, and a uniform substitution cost for phone-to-phone mappings will give misalignments compared with an alignment by a phonetician. Using phonetically based substation costs is a way to achieve an automatic alignment more similar to a hand-made one, this is addressed in e.g. [19]. For pronunciation modelling purposes the alignment errors made by using a uniform cost scheme may not be severe as these errors probably will be random. We will, however, discard data that could give useful information if the alignment was better.

In the previous chapters we have identified a number of problems with the usual dynamic programming approach using either uniformly or phonetically based substitution costs:

1. A phone loop transcription will often contain transcription errors that violate phonotactical rules, and a phonetically based substitution cost may give misalignments in these cases.

2. The phone-to-phone mappings in pronunciation variation are not symmetric. For example, a `[t]` or `[d]` in some contexts get flapped, but the flapped version does not become a `[t]` or `[d]` in the same context. This is also observed in [72] for consonants (without pronunciation variation).

3. Disfluencies may give errors both in reference and alternative transcriptions that lead to misalignments for larger segments. A dynamic programming scheme may propagate these errors.

10.2 Individual rules for non-native speakers

It is difficult to assess alignments directly. We have used indirect testing by using different alignment methods to derive pronunciation rules and pronunciation variants. The resulting modified lexica were then tested.

To compare the association strength with other alignment methods, we performed experiments using different substitution cost schemes. We performed alignment using only uniform costs by assigning high association (low cost) to identity mappings only. All non-identity mappings were assigned the same higher cost. We also used a simple phonological grouping described in e.g. [18], with 4 phone groups; vowels, sonorants, plosives and fricatives.

10.2.1 Experimental procedure

The experiments were performed using the non-native setup described in section 8.2.

Sorting the phone-to-phone mappings by association strength, we observe that many of the strongest mappings are intuitive. The top three mappings for one of the speakers (4nd) are:
`s` → `z`,
`z` → `s`, and
`p` → `b`.
In this case it seems that the place of articulation is a more important similarity measure than the manner of articulation. For another speaker a high association strength is achieved for the mapping: `l` → `ow`
Although this may seem like an error, a closer look at the alignments revealed that `l` after `ow` often was deleted, making this a useful mapping. An example alignment of the reference transcription `[ao l s ow]` and the alternative transcription `[ow z ow]` for the word "also" is:
`(ao → ow)(l → ow)(s → z)(ow → ow)`.
In this alignment we observe deletion of `l` and two substitutions; `(ao → ow)` and `(s → z)`. Without any similarity measure it would be equally probable to get the alignment
`(ao → ow)(l → z)(s → ow)(ow → ow)`,
giving a deletion of `s` and the substitutions `(ao → ow)` and `(l → z)`.

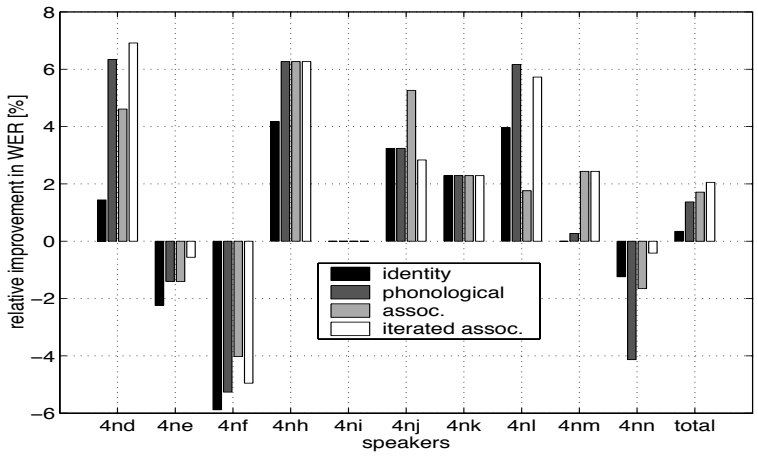

Figure 10.1: Relative improvement in WER for the different alignment schemes for non-native speakers, RPR1 > 50%.

10.2.2 Results

For each speaker we performed alignment using identity, phonological, and association based costs and made individual rules from these alignments. Sorting by estimated rule probability RPR1 > 50% according to equation (8.1) gave an overall error rate of 28.7% for the association based lexica. After four iterations of association strength computation, the resulting lexica gave an overall error rate of 28.6% WER. In figure 10.1 the results for the different alignment schemes are shown for RPR1 > 50%. The improvements (deterioration for some speakers) varied, but the association based alignment performed best on average. From figure 10.1 we also notice that the pronunciation rules that perform best for each speaker are based on one of the association based alignments, except for speaker 4nl.

Using equation (8.2) and threshold RPR2 > 20% gave an overall error rate of 28.5%. Using the RPR2 threshold the identity mapping alignment based rules were similar to the association strength alignment.

10.3 Alignment of spontaneous dictation

Inspection of the lexicon adaptation set for spontaneous dictation revealed many pauses and other disfluencies. One example of a restart is given in figure 10.2. The restart was not present in the reference transcription in this case, so the phone loop transcription corresponds better to the acoustic signal.

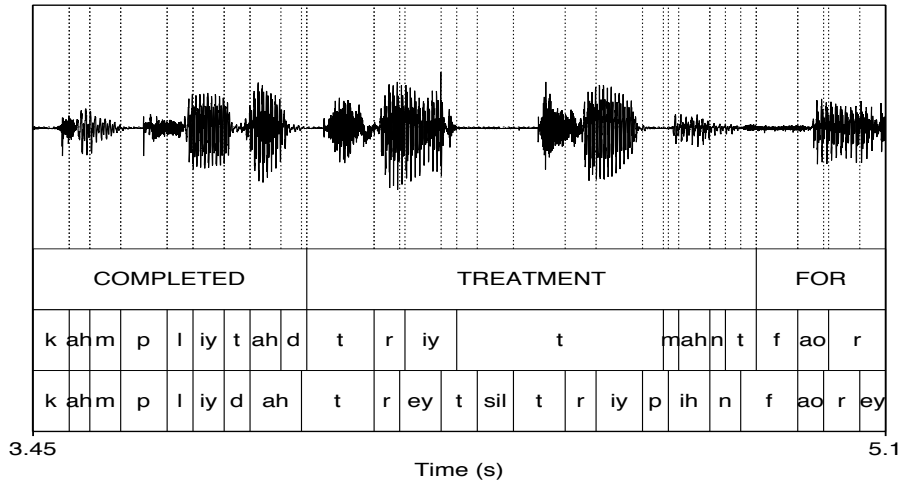

Figure 10.2: Spontaneous dictation with a restart. Word transcription is shown at the top, then forced alignment of reference transcription, and at the bottom phone loop transcription.

Dynamic programming using uniform phone-to-phone mapping costs gave the alignment:

```
k ah m p l iy      t      ah   d  t r iy t m ah n t f ao r
k ah m p l iy d ah t r ey t   sil t r iy   p ih n   f ao r ey
```

We see that the restart in "treatment" is misaligned as the end of the previous word "completed" hiding the variation `[iy t ah d $ t]` → `[iy d ah $ t]` (`$` marks the word border).

A phonetically based cost gave the alignment:

```
k ah m p l iy      t   ah d     t r iy t m ah n t f ao r
k ah m p l iy d ah t r ey t sil t r iy p   ih n   f ao r ey
```

We see that we have an alignment more phonetically appealing, but the misalignment of the end of the word before the restart is still present.

Using association strength based cost gave the alignment:

```
k ah m p l iy t      ah   d     t r iy t m ah n t f ao r
k ah m p l iy d ah t r ey t sil t r iy p   ih n   f ao r ey
```

The alignment of `[d]` in the phone loop to the `[t]` in "completed" is now better because this mapping has a high association strength, but the context for this mapping is not correct.

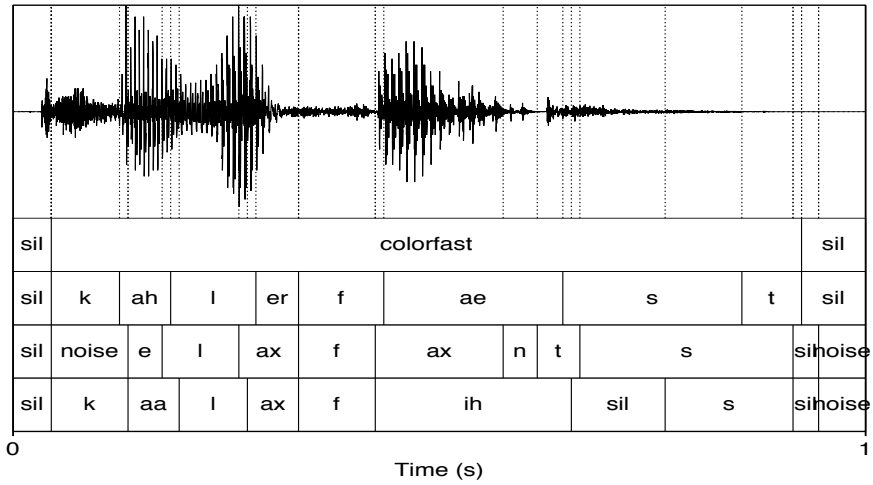

Figure 10.3: Alignment of correct word "colorfast", forced alignment of correct word, transcription of recognized word "elephants", and phone loop transcription.

Using the time synchronous alignment introduced in section 6.3.2 we get the following alignment:

```
k ah m p l iy t ah d t r iy t              m ah n t f ao r
k ah m p l iy d ah   t r ey t sil t r iy p   ih n   f ao r ey
```

We see that for the time synchronous alignment the word "completed" was aligned according to the acoustic segments and we observe the deletion of the `[d]` as well as the transformation from `[t]` to `[d]`.

10.4 Alignment of error transcriptions

One way to learn acoustic confusability for later error prediction, is to align the correct word and the errors of the recognizer output [30]. This sometimes leads to alignment of very different words that is a challenge for the usual dynamic programming algorithms. One example of this is shown in figure 10.3. The phone loop transcription is also included in the figure as this is an alternative way of learning the acoustic confusability. We observe that the segments with the same phone in the correct and recognized word also contain the same phone in the phone loop transcription.

The phonetically based alignment was in this case:

```
k ah l er f ae s t
  e  l ax f ax n t s
```

whereas a more appropriate alignment using time information could be:

```
k ah l er f ae     s t
  e  l ax f ax n t s
```

10.5 Discussion

In section 6.3.2 we introduced a new metric in the alignment based on statistical co-occurrence called *association strength*, which has been evaluated in this chapter. Using the association strength method we can easily find phone relations for individuals or groups of speakers and in this way make the alignments used in the rule generation more consistent.

Lexica modified by rules made from alignments using the association strength measure gave an improvement over baseline that was not present when using uniform phone-to-phone mapping costs in the alignment. This indicates that the proposed method is able to automatically capture the relations in phone variation due to pronunciation variation from comparison of reference and alternative phone loop based alternative transcriptions. The improvement over rules based on phonological alignment was not as high as expected; for a more restricted rule selection scheme even identity mapping gave the same result. It is still an interesting approach and it reveals some phone-to-phone mapping properties. These properties may also be useful in other parts of the pronunciation modelling task or for entirely different applications.

We have shown examples of problems that arise when aligning transcriptions with disfluencies and with correct and recognized words. A solution using time synchronous alignment is proposed.

10.6 Summary of the main results

- The association based cost scheme for dynamic programming was shown to give better alignments for rule derivation. The association based cost scheme is one way of automatically learning the mappings from the data dependent on speaker or other partitionings. The resulting costs will normally by non-symmetric.
- A time synchronous alignment scheme was presented for examples of difficult alignments.

Chapter 11

Concluding summary

In this dissertation pronunciation variation modelling for different speaking styles has been investigated. We have interpreted the term "speaking styles" widely, we have included not only read and spontaneous speech, but also non-native speech. A goal for the research on ASR is to make applications based on speech technology more user-friendly. To achieve this it is important that the system can accept pronunciations that are perceived as normal by the user. Spontaneous and non-native speech that is "normal" in the sense that humans recognize it without problems, causes problems for current ASR systems. We believe that a better modelling of pronunciation variation will give a more robust system for different speaking styles.

Pronunciation variation can be modelled in different parts of the ASR system: the acoustic models, the lexicon, or the language model. This dissertation is focused on lexical modelling, but the pronunciation modelling technique presented utilizes assessment metrics incorporating both acoustic models and implicitly the language model. A joint optimization can prevent adding variation in one part of the system that is already sufficiently modelled in another part of the system, or even worse; adding contradicting variations. We therefore strive for a unified optimization using objective criteria for all parts of the ASR system, including the lexicon.

A complete data-driven approach to pronunciation rule derivation was presented in this dissertation. Using rules we can generalize from the variation seen in the adaptation data to words not present in these data. The method presented shows how to use the acoustic log likelihood as a metric to assess the rules.

11.1 Variation modelling for different speaking styles

We have shown in chapter 7 that augmenting the recognizer lexicon with pronunciation variants found in a general purpose lexicon gave small performance gains, and most for read native speech. Error analysis showed that the system using a single canonical pronunciation generated different errors than the one using pronunciation variants, although the word error rate was similar. For non-native speech we observed no improvement for context-dependent acoustic models compared to context-independent models. This speaking style had the largest gain using speaker dependent acoustic model adaptation, but the performance was still far from the results for native speech. For spontaneous speech we observed less improvement by speaker adaptation than for read speech.

11.2 Data-driven pronunciation rule derivation and assessment

Baseform variant generation by using data-driven rule derivation can be described in five steps (repeated from section 5.4):

1. Automatically generate alternative transcriptions
2. Align the reference and alternative transcriptions
3. Derive rules from the alignment
4. Assess and prune the rules
5. Generate baseform variants from the rules, assess the variants, prune or assign weights, and modify the lexicon

In this dissertation data-driven approaches were presented for all steps, but the main contributions are on steps 2, 4 and 5. For step 2 we have introduced association strength as a way to derive phone substitution costs from the data. For step 4 we have introduced a metric based on acoustic log likelihood and for step 5 we have presented a framework for a confusability metric based on decision theory.

In chapter 8 the methods presented were evaluated on non-native speech. The results show that the acoustic log likelihood pruning gave improved performance compared with the more traditional rule probability pruning. We observed a better performance when modelling the non-native speakers jointly than individually. This was a surprising result, as the speakers had quite

different language backgrounds, but may be because the amount of data used was small. Even if the joint set of non-native speech was more diverse, the larger amount of data was beneficial to get a more reliable rule selection for the data-driven methods investigated. The confusability metric gave inconsistent results for this task, showing the vulnerability for the method when the vocabularies of the adaptation and test sets differ strongly.

For spontaneous dictation the results in chapter 9 showed no benefit by using the same pronunciation modelling techniques as used for non-native speech, neither for rule probability pruning nor acoustic log likelihood pruning. One reason may be that the rather simple rules investigated in this dissertation only modelled variation that already was sufficiently modelled by the acoustic models. The reason may also be that pronunciation modelling is not the main answer to better modelling of this speaking style.

11.3 Alignment

In chapter 10 some shortages of current alignment methods were identified. To compare dynamic programming alignment methods with different substitution cost schemes, we have compared WER after rule based pronunciation variation modelling using the different alignments. The non-native task was chosen for this experiment and the alternative transcription was made using a phone loop. The statistically based association strength was shown to produce equal or better performing rules than uniform or phonetically based substitution costs.

The usual dynamic programming approach has difficulties: 1) for transcription errors and disfluencies (usual in spontaneous speech) and 2) when aligning different words (for confusability measures). We have shown examples of this where a time synchronous alignment may be beneficial to ensure that we compare the same acoustic segments.

11.4 Conclusions

Current ASR systems perform substantially worse for both non-native and spontaneous speech than for read speech. In this dissertation we have therefore investigated a data-driven approach to rule based lexicon adaptation for these two speaking styles.

For a spontaneous dictation task the proposed method did not give any improvement, nor did more traditional rule probability based pronunciation modelling. The recognized strings were similar to the baseline result; the variation modelled by the rules did neither correct, nor introduce errors. Baseform

variants from a general purpose lexicon did not give any improvement either, but here the errors differed compared with using one canonical baseform entry.

For the non-native task we observed the same effect as for spontaneous speech when adding general purpose variants: The errors differed, but the performance did not. The proposed rule-based lexicon adaptation gave significant improvements for this task, and we observed larger gain for the new acoustic log likelihood metric compared with a rule probability metric.

The results indicate that we should choose different ways to model pronunciation variation for these two speaking styles. However, the rules investigated in this dissertation are rather simple and this may be one reason why they were better able to model the larger shifts in pronunciation present in non-native speech. Further research combining more sophisticated rule derivation with the proposed acoustic log likelihood pruning should be tried before we reject the hypothesis that pronunciation variation modelling also will give a better performance for spontaneous speech.

One of the main differences between read and spontaneous speech is the grammar used as well as disfluencies like restarts and long pauses. The language model may therefore be the best choice for more research to achieve better performance for this speaking style.

Even if speaker adaptation was shown to give large improvements for non-native speech, the resulting performance was worse than for native speech on the same task. To achieve results more comparable to native speech, a combination of lexical and acoustic adaptation may be beneficial. It is then crucial to use a metric for the pronunciation rule and variant pruning that incorporates the variation accounted for by the acoustic models. The proposed metric in this dissertation is a step towards this goal.

11.5 Some directions for further work

The rules investigated in this dissertation are rather simple. We have only looked at phone identity as context for the transformation. We will then only be able to model the phone sequences seen in the adaptation data. CART based rule modelling is one way of generalizing the context automatically from the data, and a combination of CART based rule derivation and acoustic log likelihood assessment would therefore be interesting for further studies. With a better generalization (or more data) more sophisticated rules could be derived using more information, e.g. stress and syllable information or for non-natives especially, orthographic information. Coarticulaton effects across word borders should also be included in the variation modelling.

The step from rules to variants needs more attention. We have in this

dissertation assessed only the most important variants by the restriction of applying only one rule to each baseform. A rule hierarchy based on an acoustic log likelihood metric must be formulated to extend the method presented to using several rules in one baseform.

The unigram language model was implicitly incorporated in the rule pruning metric. This was done by assessing the total effect instead of the relative effect of the rules. We then incorporate the word probability found in the adaptation set. Higher order language models should be incorporated, and more explicitly by using the probabilities from the language model defined for the task. The probabilities given will then be the same as used in the recognition phase. The task language model is usually trained on much larger amounts of text data than in the adaptation data and will give more reliable estimates for the probabilities. One problem with assessing the total effect is that we then needed the extra restriction of using only one rule for each rule condition. A clustering procedure where we ensure that each acoustic segment only contributes once may be beneficial.

Controlling the confusability is important. A joint optimization using acoustic log likelihood in the pronunciation rule pruning will to some extent help to reduce the confusability by preventing addition of superfluous variation. For a proper confusability metric the combined effect of the baseforms should be assessed. We have in this dissertation presented a framework for discriminative pronunciation assessment with confusability measures based on decision theory. Assessing this framework on data with better agreement between the adaptation and test data than the non-native task should be investigated. When a suitable metric is found it can be used in data-mining approaches to find the optimal set of baseforms.

One difference between the non-native and spontaneous tasks we have investigated is that for the non-native speakers we derived pronunciation rules from an adaptation set with the same speakers as in the test set. Deriving speaker dependent lexica is an interesting task for future research where the main challenge is that we normally will have small amounts of data for each speaker.

Dialects and native accented speech are not investigated in this dissertation, but we may assume that these speaking styles will behave more similar to non-native speech than spontaneous speech. An investigation of the proposed method, and refinements of it for dialect pronunciation modelling, should be investigated.

Appendix A

Phonetic alphabets

Phone symbol		Example	Transcription	Phone class
CMU	SAMPA			
ih	I	it	ih t	checked vowel
eh	E	ed	eh d	checked vowel
ae	{	at	ae t	checked vowel
aa	A	odd	aa d	checked vowel
ah	V	hut	hh ah t	checked vowel
uh	U	hood	hh uh d	checked vowel
iy	i	eat	iy t	free vowel
ey	eI	ate	ey t	free vowel
ay	aI	hide	hh ay d	free vowel
oy	OI	toy	t oy	free vowel
uw	u	two	t uw	free vowel
ow	oU	oat	ow t	free vowel
aw	aU	cow	k aw	free vowel
ao	O	ought	ao t	free vowel
er	3`	hurt	hh er t	free vowel

Table A.1: US English vowels in the CMU and SAMPA phonetic alphabets

The International Phonetic Alphabet (IPA) of the International Phonetic Association [57] is widely used by linguists, but for ASR purposes more computer friendly alternatives are used. In this dissertation we have mainly used the phone symbols given by the CMU phonetic alphabet [15]. This is a phonetic alphabet for transcription of US English and consists of 39 symbols, not counting variants for lexical stress. The Speech Assessment Methods Pho-

netic Alphabet (SAMPA) is a well defined standard for a computer readable phonetic alphabet, [97]. Both phone sets are shown in tables A.1 and A.2 with the phone classification used by SAMPA.

The CMU phone set is similar to the ARPABET which is widely used in the ASR community. Slightly different versions of ARPABET exist. The version defined in [73] and [90] consists of more phones than the CMU set and some symbols differ. For example is `[nx]` used instead of `[ng]` and `[axr]` instead of `[er]`.

The Bell Labs recognizer (BLASR) uses the two additional phones shown in table A.3:

- Where the BLASR lexicon uses the phone here described as `[ax]`, the CMU lexicon uses `[ah]` or `[ih]`. This phone is often called "schwa".
- Where the BLASR lexicon uses the phone here described as `[aaa]`, the CMU lexicon uses `[aa]`.

If pronounced differently the `[aa]` will be rounded while the `[aaa]` will be unrounded. These two phones may be pronounced similarly in some regions in the US (both unrounded) while they will be different in other regions (e.g. in British English).

Phone symbol CMU	Phone symbol SAMPA	Example	Transcription	Phone class
p	p	pee	p iy	plosive
b	b	be	b iy	plosive
t	t	tea	t iy	plosive
d	d	dee	d iy	plosive
k	k	key	k iy	plosive
g	g	green	g r iy n	plosive
ch	tS	cheese	ch iy z	affricate
jh	dZ	gee	jh iy	affricate
f	f	fee	f iy	fricative
v	v	vee	v iy	fricative
th	T	theta	th ey t ah	fricative
dh	D	thee	dh iy	fricative
s	s	sea	s iy	fricative
z	z	zee	z iy	fricative
sh	S	she	sh iy	fricative
zh	Z	seizure	s iy zh er	fricative
hh	h	he	hh iy	fricative
m	m	me	m iy	sonorant
n	n	knee	n iy	sonorant
ng	N	ping	p ih ng	sonorant
r	r	read	r iy d	sonorant
l	l	lee	l iy	sonorant
w	w	we	w iy	sonorant
y	j	yield	y iy l d	sonorant

Table A.2: US English consonants in the CMU and SAMPA phonetic alphabets

Phone symbol used	Phone symbol SAMPA	Example	Transcription	Phone class
ax	@	another	ax n ah dh er	short central vowel
aaa	A:	stars	s t aaa r z	free vowel

Table A.3: Extra vowels used in BLASR

Appendix B

CMU lexicon specification

In the experiments in chapters 7 and 9 we have used the cmudict.0.6 from August 8, 1998, [15]. The README-file is given below. In chapter 8 we have used the BLASR lexicon from Bell Labs. A comparison of the CMU and BLASR lexica is difficult as they use different phone sets as described in appendix A.

The CMU lexicon README file:

```
Date: 11-8-95

Files: README (this file), cmudict.0.1.Z (compressed), cmulex.0.1.Z,
cmudict.0.2.Z (compressed), cmudict.0.3.Z (compressed), cmudict.0.4.Z,
cmulex.0.3.Z, cmulex.0.4.Z, phoneset.0.1, phoneset.0.3, phoneset.0.4.

This directory contains pronunciation dictionaries (cmudict.0.1.Z is
the first one we put out, cmudict.0.4.Z is the latest and most
up-to-date) containing approximately 100k words and their
transcriptions; lists of the words are in cmulex.0.[134].Z. We use
these dictionaries at Carnegie Mellon in our speech understanding
systems.

The phone set for cmudict.0.4 contains 39 phones, a list of which can be
found in phoneset.0.4.

Lexical stress is indicated by means of a numeral [012] attached to a vowel:
  0 = no stress
  1 = primary stress
  2 = secondary stress

Alternate transcriptions are identified with a numeral in parentheses as
part of the lexical entry.
```

```
We generated this dictionary using the following independent sources:
- a 20k+ general English dictionary, built by hand at Carnegie Mellon
  (extensively proofed and used).
- a 200k+ UCLA-proofed version of the shoup dictionary.
- a 32k subset of the Dragon dictionary.
- a 53k+ dictionary of proper names, synthesiser-generated, unproofed.
- a 200k dictionary generated with Orator, unproofed.
- a 200k dictionary generated with Mitalk, unproofed.

All entries that occur solely in copyrighted sources, like the Dragon
dictionary, are not currently included in this dictionary. If you have
words and transcriptions that you would like included in this unrestricted
resource, please send them to Robert L. Weide (weide@cs.cmu.edu) and we
will consider them for an upcoming version.

All of the above sources were preprocessed and the transcriptions in the
current cmudict.0.1 were selected from the transcriptions in the sources or
a combination thereof. We have removed some potentially unreliable
transcriptions from this dictionary, including those based on only one
source, and will reintroduce them once we have verified the transcriptions.

CMU does not guarantee the accuracy of this dictionary, nor its suitablity
for any specific purpose. In fact, we expect a number of errors, omissions
and inconsistencies to remain in the current result. We intend to
continually update the dictionary as we make progress in correcting them.
We will make subsequent versions available via anonymous ftp, and those
who would like notification when updated versions are available should
send email to weide@cs.cmu.edu.

We welcome input from users: send e-mail to Robert L. Weide
(weide@cs.cmu.edu) if you have comments and suggestions on the content
of the dictionary.

The Carnegie Mellon Pronouncing Dictionary [cmudict.0.4 and all previous
versions] is Copyright 1993, 1994, and 1995 by Carnegie Mellon University.
Use of this dictionary for any research or commercial purpose is completely
unrestricted.  If you make use of or redistribute this material, we would
appreciate acknowlegement of its origin.

If you add words to or correct words in this dictionary, we would like
the additions and corrections sent to us (weide@cs) for consideration
in a subsequent version. All final entries will be approved by Robert L.
Weide, editor of the dictionary.
```

Appendix C

Confidence intervals

The confidence intervals used in this dissertation are computed according to methods reported in [47] and [56]. We assume that the errors are independently distributed according to a binomial distribution where n is the number of samples (in our case words) and p is the word error rate (WER) of the baseline system. The probability of the number of errors n_e can then be expressed as:

$$P(n_e = x) = b(x; n, p) = \binom{n}{x} \cdot p^x \cdot (1-p)^{(n-x)}. \tag{C.1}$$

We want to test the hypothesis H_0 that a new recognizer has the same number of errors; $n_e = np$ (where np is the mean of the binomial distribution). We reject the H_0 hypothesis if the probability that the number of errors n_e is outside a confidence interval $(c_1,\ c_2)$, is larger than a value given by the confidence coefficient. The low and high borders of the confidence interval for a given confidence coefficient can be found by:

$$P(c_1 < n_e < c_2) = 1 - \alpha \tag{C.2}$$

Furthermore, the binomial distribution can be approximated with a normal (Gaussian) distribution with mean np and variance $np(1-p)$. We then get an expression for the confidence interval given the total number of samples n and the error rate p of the baseline system.

The tables below give the 95% confidence intervals ($\alpha = 0.05$) computed for the number of samples in the test sets used in this dissertation for a selection of word error rates.

WER	*Low*	*High*	*Interval*
5	4.30	5.79	1.49
10	9.01	11.04	2.03
15	13.81	16.22	2.41
20	18.68	21.38	2.70
25	23.55	26.47	2.92
30	28.46	31.54	3.08
35	33.41	36.62	3.21
40	38.35	41.65	3.30
45	43.31	46.66	3.35
50	48.32	51.68	3.36

Table C.1: Confidence intervals for 3446 words (`h1`)

WER	*Low*	*High*	*Interval*
5	4.39	5.66	1.27
10	9.15	10.90	1.75
15	13.97	16.05	2.08
20	18.85	21.17	2.32
25	23.76	26.27	2.51
30	28.67	31.33	2.66
35	33.62	36.38	2.76
40	38.58	41.42	2.84
45	43.54	46.43	2.89
50	48.54	51.44	2.90

Table C.2: Confidence intervals for 4633 words (`s9`)

WER	*Low*	*High*	*Interval*
5	4.51	5.52	1.01
10	9.33	10.70	1.37
15	14.20	15.83	1.63
20	19.10	20.93	1.83
25	24.01	25.99	1.98
30	28.96	31.05	2.09
35	33.91	36.10	2.19
40	38.89	41.13	2.24
45	43.86	46.13	2.27
50	48.85	51.14	2.29

Table C.3: Confidence intervals for 7435 words (`s3`)

WER	*Low*	*High*	*Interval*
5	4.33	5.74	1.41
10	9.06	10.98	1.92
15	13.89	16.17	2.28
20	18.73	21.29	2.56
25	23.64	26.40	2.76
30	28.54	31.46	2.92
35	33.49	36.53	3.04
40	38.43	41.55	3.12
45	43.42	46.59	3.17
50	48.40	51.58	3.18

Table C.4: Confidence intervals for 3849 words (`h2`)

Appendix D

Decision theory

This appendix contains some general decision theory and provides definitions for the equations used in section 6.1. The appendix is based on [22], [96], and [64]. The notation chiefly follows Katagiri et al. in [64].

The basic problem in decision theory is that we observe an object and want to decide which class the observation belongs to. The object will usually be some kind of feature vector representing speech, images, or other signals. We use training data to optimize the classifier that we use for the decisions.

A classification system consists of:

- The classifier $C(\cdot)$
- M classes $C_j, j = 1, 2, \ldots, M$
- A design sample set of feature vectors $\{\mathbf{x}\}$

Classification can be described as a partitioning of the signal space in disjoint regions. We want to use the training data to find the partitioning that gives us minimum error rate. Therefore we need an optimizing criterion emphasizing borders rather than typical examples.

D.1 Bayes classifier

Optimization regarding the *minimum error classification* (MCE) criterion will give us a *Bayes classifier* which is defined as choosing the class that maximizes the *a posteriori* probability, [22].

$$C(\mathbf{x}) = C_i \iff i = \underset{j}{\operatorname{argmax}}\, P(C_j \mid \mathbf{x}) \tag{D.1}$$

The a posteriori probability can be expressed utilizing the conditional probability densities and the *a priori* probability:

$$\begin{aligned} P(C_j \mid \mathbf{x}) &= \frac{p(\mathbf{x}, C_j)}{p(\mathbf{x})} \\ &= \frac{p(\mathbf{x} \mid C_j)P(C_j)}{\sum_k p(\mathbf{x} \mid C_k)P(C_k)} \\ &= \frac{p_j(\mathbf{x})P(C_j)}{\sum_k p_k(\mathbf{x})P(C_k)} \end{aligned} \tag{D.2}$$

The last line is just a change of notation showing the conditional probability densities as class densities.

Using only the numerator gives a *maximum likelihood* (ML) classifier:

$$C(\mathbf{x}) = C_i \iff i = \operatorname*{argmax}_j p(\mathbf{x} \mid C_j)P(C_j) = \operatorname*{argmax}_j p_j(\mathbf{x})P(C_j) \tag{D.3}$$

The class density is estimated using the acoustic models and the a priori probability is estimated using the language model. In the maximum likelihood (ML) framework both these two components are estimated, but only for each class individually. Skipping the denominator removes the dependency on the other classes.

Using the Bayes classifier we must estimate the posterior probability. Usually this is done using a model for the posterior probability, i.e. a *parametric* estimation. We then estimate the parameters in the density chosen instead of the entire density. The design issue is which type of probability densities to use since we rely on a reasonable assumption for the model. The functional form of the conditional probability density is in practice rarely known, but a Gaussian distribution is often used since the central limit theorem states that the limiting distribution as the number of sampled increases is Gaussian. The estimation will often be done using a plug-in classifier and we estimate the posterior probability by estimating the class density and the class probability:

$$\widehat{P}(C_j \mid \mathbf{x}) = \frac{p_j(\mathbf{x}; \widehat{\theta})\widehat{P}(C_j)}{\sum_k p_k(\mathbf{x}; \widehat{\theta})\widehat{P}(C_k)} \tag{D.4}$$

The class probability is often estimated by counting occurrences:

$$\widehat{P}(C_j) = \frac{\text{count}(\mathbf{x} \in C_j)}{\sum_k \text{count}(\mathbf{x} \in C_k)} \tag{D.5}$$

The parameters θ are the parameters needed to describe the class densities. Using a plug-in classifier we have two levels of approximations, first we must find an estimator $\widehat{\theta}$ for θ and secondly assume that $p_k(\mathbf{x}; \widehat{\theta})$ is a good estimator for $p_k(\mathbf{x}; \theta)$.

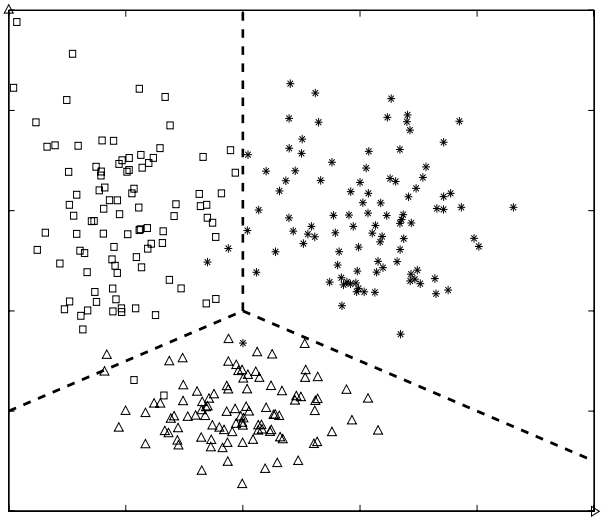

Figure D.1: Two dimensional signal space with three classes divided in disjoint regions.

D.2 Discriminant functions

Instead of estimating the posterior probability the classifier can be represented by a *discriminant function* for each class C_j:

$$g_j(\mathbf{x}; \Lambda) \tag{D.6}$$

The classifier will then consist of the parameter set Λ and a decision rule given by the discriminant functions. The decision rule is to choose the class with the discriminant function that gives the highest value:

$$C(\mathbf{x}) = C_i \iff i = \underset{j}{\operatorname{argmax}}\, g_j(\mathbf{x}; \Lambda) \tag{D.7}$$

The discriminant functions will divide the signal space in disjoint regions as shown in figure D.1.

The relation between the Bayes decision rule and the discriminant function representation of the classifier is given by the following "Bayes classifier" discriminant function:

$$g_j(\mathbf{x}; \Lambda) = P(C_j \mid \mathbf{x}; \Lambda) \tag{D.8}$$

The discriminant function does not need to be a probability function, the parameters Λ of the classifier can be trained to reduce classification error which

will be a more direct approach than to model parameters of a class distribution. Estimating the discriminant function directly is called a *non-parametric* estimation. The discriminant function approach makes it possible to achieve minimum classification error without having (or assuming) the true parametric form of the probability function. The distinction between parametric and non-parametric is often less clear-cut than the names indicate, quoting [96] page 26:

> "The real distinction is between families of probabilities which are quite constrained by having only few parameters, and those which are so flexible that they can approximate (almost) any posterior probabilities."

D.3 Optimization

To train the classifier we need an optimizing function, and we want to find the connection between this function and the parameters in the classifier that we can adjust. For minimum error classification the optimizing function is related to an estimate of the average error rate, or in decision theory notation: expected loss.

We need a *loss function* giving us the cost of a misclassification. A zero-one loss function will count all errors as equally important and by that avoid the difficulty in setting class-by-class loss, see figure D.2:

$$l_i(\mathbf{x};\Lambda) = \begin{cases} 0, & C(\mathbf{x}) = i \\ 1, & \text{otherwise} \end{cases} \tag{D.9}$$

The decision process of the classifier in equation (D.7) is to compare competing classes. Using the discriminant function from equation (D.6) this process can be emulated using a distance we will call a *misclassification measure*:

$$d_i(\mathbf{x};\Lambda) = g_j(\mathbf{x};\Lambda) - g_i(\mathbf{x};\Lambda) \tag{D.10}$$

A positive value from this measure means that C_j is preferred over C_i and we have a classification error. ($C_i = C_j$ will give the value zero, i.e. no loss). C_j is the class with the largest discriminant value among those classes other than C_i. The zero-one loss function can then be reformulated using the misclassification measure:

$$l_i(\mathbf{x};\Lambda) = \begin{cases} 0, & d_i(\mathbf{x};\Lambda) \leq 0 \\ 1, & 0 < d_i(\mathbf{x};\Lambda) \end{cases} \tag{D.11}$$

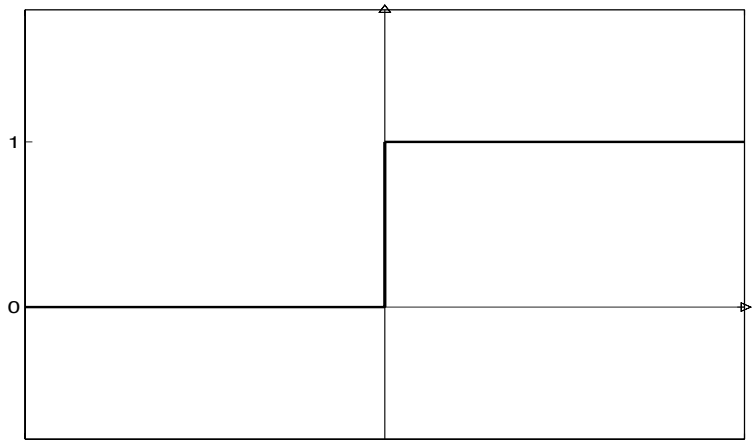

Figure D.2: Zero-one loss function.

We may also use introduce a doubt decision in the loss function. This may help to handle unseen events. We can use the misclassification measure to incorporate the "amount" of misclassification in a doubt region using a linear function as shown in figure D.3, [64]:

$$l_i(\mathbf{x};\Lambda) = \begin{cases} 0, & d_i(\mathbf{x};\Lambda) \leq 0 \\ \frac{1}{c2} \cdot d_i(\mathbf{x};\Lambda), & 0 < d_i(\mathbf{x};\Lambda) \leq c2 \\ 1, & c2 < d_i(\mathbf{x};\Lambda) \end{cases} \tag{D.12}$$

This makes the classifier depend on how well the value of the discriminant functions and the misclassification measures correspond to the "real" misclassification and how comparable the values for different errors are.

To find the total expected loss we must first define the misclassification probability, $\text{pmc}(C_i)$, for class C_i (the class is truly C_i):

$$\text{pmc}(C_i) = P\{C(\mathbf{x}) \neq C_i \mid C_i\} \tag{D.13}$$

For a Bayes classifier we count errors and use the zero-one loss function. The *expected loss*, also called *risk function*, will then become, [96]:

$$\begin{aligned} R(C(\cdot), C_i) &= \sum_{k=1}^{K} P\{C(\mathbf{x}) = C_k \mid C_i\} \\ &\approx P\{C(\mathbf{x}) \neq C_i \mid C_i\} \\ &= \text{pmc}(C_i) \end{aligned} \tag{D.14}$$

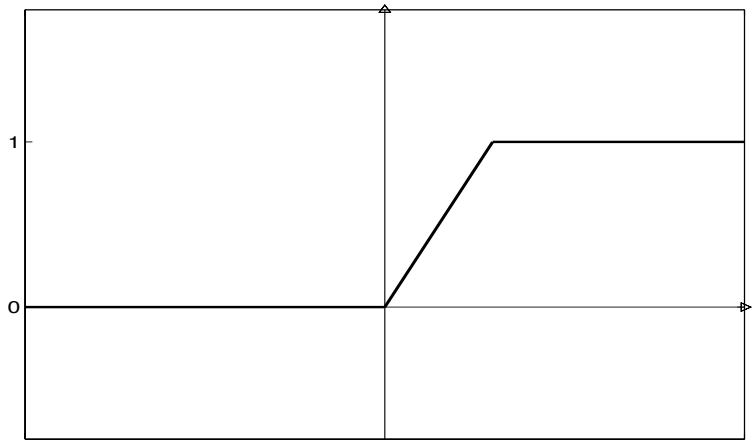

Figure D.3: Loss function with linear doubt region.

The approximation will be equality if we have non-overlapping errors, i.e. that for each sample one or no competing classes will have better score than the correct one. If several competing classes have better scores than the correct one for a sample of C_k, the first line will give a higher loss than the true one. The expression in line one will be an *upper bound* on the expected loss.

To get the *total expected loss* or *total risk* we take the expectation over all classes:

$$\begin{aligned} R(C(\cdot)) &= \mathrm{E}[R(C(\cdot), C_i)] \\ &= \sum_{k=1}^{K} P(C_k)\mathrm{pmc}(C_k) \end{aligned} \tag{D.15}$$

The probability of misclassification (pmc) must be estimated using the parameters of the classifier. We will then have a function giving us the connection between minimum error rate and the parameters of the classifier. We can then optimize the classifier regarding the parameters using this function.

It is, however, important that our classifier training method generalizes well. Learning the training data by heart will often not help much in classifying unseen samples. A loss function that represents the amount of error will help. Usually this is done by using a function of the misclassification measure. This will make the system more dependent on any assumptions in the design of the classifier.

Choosing between a parametric and a non-parametric estimation will be a trade-off between our belief in a distribution assumption and the amount

of relevant data. If we have sufficient training data we can estimate the distributions (or discriminative functions) directly. On the other hand, the model assumptions in parametric modelling may help us in generalizing when we have less data.

Appendix E

Detailed experimental results for different speaking styles

E.1 Choice of retranscription scheme for variant training

As explained in section 7.2.3 we have several ways of training variant models. The usual scheme is to use the variants in a forced alignment once in an early phase of the training, i.e. using monophone models with only one component in the observation density. We have also tried a scheme where we retranscribe with the current models at every mixture update to get a better quality transcription for further training.

We have used two ways of assessing the effect of the retranscription schemes used in training:

1. The likelihood of the training data
2. The WER of the test data

We can examine how well the acoustic models explain the training data for various amounts of training and setups by looking at the average log likelihood per frame. Retranscribing for every mixture update gave higher log likelihood of the training data after each iteration compared with retranscribing once both for monophones, see figure E.1, and for triphones, see figure E.2. The difference was however minor and we saw the same increase for canonical and variant models.

The log likelihood after each iteration of training for the variant and canonical models was also compared. There was hardly any difference between using transcriptions based on a canonical lexicon and transcriptions

with variants. For the canonical models the increased log likelihood after 8 iterations compared with 4 iterations is only due to the larger number of iterations used in the training. This suggests that the increased log-likelihood for the variant models trained using 8 iteration also is due to the larger number of iterations rather than the effect of the retranscription. Since the difference between canonical and variant models was minor, the retranscription effect may be small because the variant effect is small.

The triphones, figure E.1, gave much higher log likelihood on the training set than the monophones, figure E.2, as expected.

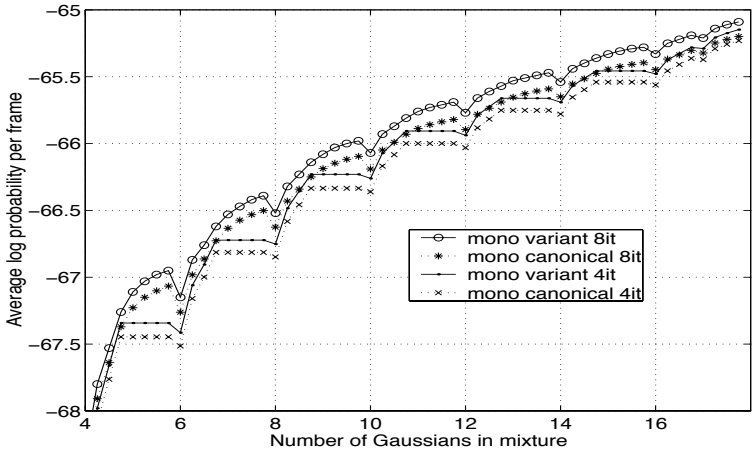

Figure E.1: Log probability per frame, monophone acoustic models.

The recognition results on the test set showed a small increase in performance for most conditions when retranscribing for every mixture update, see tables E.1, E.2, E.3, and E.4.

For the variant HMM experiments presented in the main results in chapter 7 we chose to use models with retranscription for every mixture update, i.e. the 8 iteration models. For the canonical models we used the 4 iteration version since we saw no improvement and a small deterioration for some triphone setups using 8 iterations.

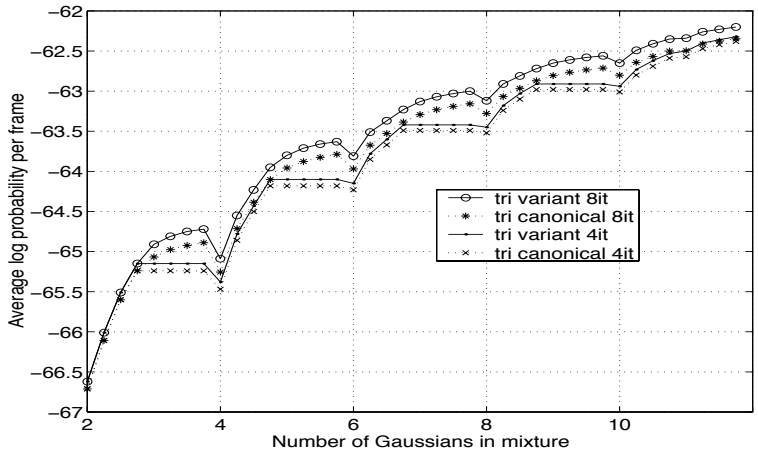

Figure E.2: Log probability per frame, triphone acoustic models.

Training	*Test*	*h1*	*s9*	*s3*	*h2*
Canonical	Canonical	34.7	40.4	27.6	22.2
Canonical	Variant	32.9	39.3	27.2	21.1
Variant	Variant	33.2	40.3	28.4	21.5
Variant	Canonical	36.2	42.8	29.7	23.7

Table E.1: WER in [%] for monophone acoustic models using 4 iterations (only retranscribing once)

Training	*Test*	*h1*	*s9*	*s3*	*h2*
Canonical	Canonical	34.2	40.3	27.4	21.6
Canonical	Variant	32.5	39.2	27.5	21.2
Variant	Variant	32.0	38.2	26.7	20.1
Variant	Canonical	35.8	40.8	28.2	22.2

Table E.2: WER in [%] for monophone acoustic models using 8 iterations (retranscribing for every mixup)

Training	*Test*	*h1*	*s9*	*s3*	*h2*
Canonical	Canonical	15.9	23.7	27.7	8.0
Canonical	Variant	15.2	23.5	28.4	8.4
Variant	Variant	15.4	24.0	28.5	7.7
Variant	Canonical	20.1	29.1	30.3	11.5

Table E.3: WER in [%] for triphone acoustic models 4 iterations (only retranscribing once)

Training	*Test*	*h1*	*s9*	*s3*	*h2*
Canonical	Canonical	16.3	24.3	27.6	8.2
Canonical	Variant	15.8	23.6	28.4	8.5
Variant	Variant	15.4	24.4	28.9	7.4
Variant	Canonical	20.5	29.4	31.0	11.6

Table E.4: WER in [%] for triphone acoustic models using 8 iterations (retranscribing for every mixup)

Bibliography

[1] M. Adda-Decker, "Towards multilingual interoperability in automatic speech recognition," *Speech Communication*, vol. 35, pp. 5–20, 2001.

[2] J. B. Allan, "How do humans process and recognize speech?," *IEEE Transactions on speech and audio processing*, vol. 2, pp. 567–577, October 1994.

[3] I. Amdal, T. Holter, and T. Svendsen, "Maximum likelihood pronunciation modelling of Norwegian natural numbers for automatic speech recognition," in *Proc. Norwegian Signal Processing Symposium (NORSIG)*, (Asker, Norway), pp. 145–150, 1999.

[4] I. Amdal, F. Korkmazskiy, and A. C. Surendran, "Data-driven pronunciation modelling for non-native speakers using association strength between phones," in *Proc. ISCA ITRW ASR2000*, (Paris, France), pp. 85–90, 2000.

[5] I. Amdal, F. Korkmazskiy, and A. C. Surendran, "Joint pronunciation modelling of non-native speakers using data-driven methods," in *Proc. ICSLP-2000*, (Beijing, China), pp. III:622–625, 2000.

[6] I. Amdal and T. Svendsen, "Evaluation of pronunciation variants in the ASR lexicon for different speaking styles," in *Proc. LREC-2002*, (Las Palmas de Gran Canaria, Spain), pp. 1290–1295, 2002.

[7] B. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *Journal of the Acoustical Society of America*, vol. 55(6), pp. 1304–1312, 1974.

[8] L. R. Bahl, R. Bakis, J. Bellegarda, P. F. Brown, D. Buhrstein, S. K. Das, P. V. de Souza, P. S. Gopalakrishnan, F. Jelinek, D. Kanevsky, R. L. Mercer, A. J. Nadas, D. Nahamoo, and M. A. Picheny, "Large vocabulary natural language continuous speech recognition," in *Proc. ICASSP-89*, (Glasgow, Scotland), pp. 465–467, 1989.

[9] L. R. Bahl, S. Das, P. V. de Souza, M. Epstein, R. L. Mercer, B. Merialdo, D. Nahamoo, M. A. Picheny, and J. Powell, "Automatic phonetic baseform determination," in *Proc. ICASSP-91*, (Toronto, Canada), pp. 173–176, 1991.

[10] L. R. Bahl, P. V. de Souza, P. S. Gopalakrishnan, D. Nahamoo, and M. A. Picheny, "Decision trees for phonological rules in continuous speech," in *Proc. ICASSP-91*, (Toronto, Canada), pp. 185–188, 1991.

[11] J. K. Baker, "The DRAGON system approach – An overview," *IEEE Transactions on acoustics, speech and signal processing*, vol. ASSP-23, pp. 24–29, February 1975.

[12] B. Balentine and D. P. Morgan, *How to build a speech recognition application*. Enterprise Integration Group, 1999.

[13] W. J. Byrne, M. Finke, S. P. Khudanpur, J. McDouough, H. Nock, M. D. Riley, M. Saraclar, C. Wooters, and G. Zavaliagkos, "Pronunciation modelling using a hand-labelled corpus for conversational speech recognition," in *Proc. ICASSP-98*, (Seattle (WA), USA), pp. 313–316, 1998.

[14] W. Chou, C.-H. Lee, and B.-H. Juang, "Minimum error rate training based on N-best string models," in *Proc. ICASSP-93*, (Minneapolis (MN), USA), pp. II:652–655, 1993.

[15] *CMU Pronunciation Dictionary*. [online], 1998. [cited 2002-03-01]. URL: `http://www.speech.cs.cmu.edu/cgi-bin/cmudict/`.

[16] R. V. Cox, C. A. Kamm, L. R. Rabiner, J. Schroeter, and J. G. Wilpon, "Speech and language processing for next-millennium communication services," *Proceedings of the IEEE*, vol. 88(8), pp. 1314–1337, 2000.

[17] N. Cremelie and J.-P. Martens, "Automatic rule-based generation of word pronunciation networks," in *Proc. EUROSPEECH-97*, (Rhodes, Greece), pp. 2459–2462, 1997.

[18] N. Cremelie and J.-P. Martens, "In search of better pronunciation models for speech recognition," *Speech Communication*, vol. 29, pp. 115–136, 1999.

[19] C. Cucchiarini, "Assessing transcription agreement: methodological aspects," *Clinical linguistics & phonetics*, vol. 10 (2), pp. 131–155, 1996.

[20] V. Diakoloukas, V. V. Digalakis, L. G. Neumeyer, and J. Kaja, "Development of dialect-specific speech recognizers using adaptation methods," in *Proc. ICASSP-97*, (Munich, Germany), pp. 1455–1458, 1997.

[21] V. V. Digalakis and L. G. Neumeyer, "Speaker adaptation using combined transformation and Bayesian methods," *IEEE Transactions on speech and audio processing*, vol. 4, pp. 294–300, July 1996.

[22] R. O. Duda and P. E. Hart, *Pattern classification and scene analysis.* Wiley-interscience, 1973.

[23] *The European Language resources Distribution Agency (ELDA).* [online description], 2002. [cited 2002-08-01]. URL: `http://www.elda.fr/`.

[24] R. T. Endresen, *Fonetikk og fonologi: Ei elementær innføring.* Universitetsforlaget, (Oslo, Norway), 1991.

[25] M. Finke and I. Rogina, "Wide context acoustic modeling in read vs. spontaneous speech," in *Proc. ICASSP-97*, (Munich, Germany), pp. 1743–1746, 1997.

[26] M. Finke and A. Waibel, "Speaking mode dependent pronunciation modelling in large vocabulary conversational speech recognition," in *Proc. EUROSPEECH-97*, (Rhodes, Greece), pp. 2379–2382, 1997.

[27] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, (Santa Barbara (CA), USA), pp. 347–354, 1997.

[28] L. D. Fisher and G. van Belle, *Biostatistics*, ch. Hypothesis testing for binomial variables, pp. 182–183. Wiley, 1993.

[29] E. Fosler-Lussier, "Multi-level decision trees for static and dynamic pronunciation models," in *Proc. EUROSPEECH-99*, (Budapest, Hungary), pp. 463–466, 1999.

[30] E. Fosler-Lussier, I. Amdal, and H.-K. J. Kuo, "On the road to improved lexical confusability metrics," in *Proc. ISCA ITRW Pronunciation Modeling and Lexicon Adaptation (PMLA)*, (Estes Park (CO), USA), pp. 53–58, 2002.

[31] E. Fosler-Lussier and N. Morgan, "Effects of speaking rate and word frequency on pronunciations in conversational speech," *Speech Communication*, vol. 29, pp. 137–158, 1999.

[32] E. Fosler-Lussier and G. Williams, "Not just what, but also when: Guided automatic modeling of Broadcast News," in *Proc. DARPA Broadcast News Workshop*, (Herndon (VA), USA), pp. 171–174, 1999.

[33] J. E. Fosler-Lussier, *Dynamic pronunciation models for automatic speech recognition*. PhD thesis, University of California, Berkeley, 1999.

[34] B. Gajić, *Feature extracion for automatic speech recognition in noisy acoustic environments*. PhD thesis, NTNU (Norwegian University of Science and Technology), 2002.

[35] A. Ganapathiraju, J. Hamaker, J. Picone, M. Ordowski, and G. R. Doddington, "Syllable-based large vocabulary continuous speech recognition," *IEEE Transactions on speech and audio processing*, vol. 9, pp. 358–366, May 2001.

[36] J.-L. Gauvain and C.-H. Lee, "Bayesian learning for hidden Markov model with Gaussian mixture state observation densities," *Speech Communication*, vol. 11, pp. 205–213, 1992.

[37] J.-L. Gauvain and C.-H. Lee, "Maximum *a posteriori* estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on speech and audio processing*, vol. 2, pp. 291–298, April 1994.

[38] E. P. Giachin, A. E. Rosenberg, and C.-H. Lee, "Word juncture modeling using phonological rules for HMM-based continuous speech recognition," *Computer Speech and Language*, vol. 5, pp. 155–168, 1991.

[39] L. Gillick and S. J. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *Proc. ICASSP-89*, (Glasgow, Scotland), pp. 532–535, 1989.

[40] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "Switchboard: Telephone speech corpus for research and development," in *Proc. ICASSP-92*, (San Francisco (CA), USA), pp. I:517–520, 1992.

[41] S. Goronzy, R. Kompe, and S. Rapp, "Generating non-native pronunciation variants for lexicon adaptation," in *Proc. ISCA ITRW Adaptation methods for speech recognition*, (Sophia-Antipolis, France), pp. 143–146, 2001.

[42] S. Greenberg, "Recognition in a new key – towards a science of spoken language," in *Proc. ICASSP-98*, (Seattle (WA), USA), pp. 1041–1044, 1998.

[43] S. Greenberg, "Speaking in shorthand – A syllable-centric perspective for understanding pronunciation variation," *Speech Communication*, vol. 29, pp. 159–176, 1999.

[44] S. Greenberg, J. Hollenback, and D. Ellis, "Insights into spoken language gleaned from phonetic transcription of the Switchboard corpus," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. Addendum 24–27, 1996.

[45] T. Hain and P. C. Woodland, "Dynamic HMM selection for continuous speech recognition," in *Proc. EUROSPEECH-99*, (Budapest, Hungary), pp. 1327–1330, 1999.

[46] T. Hain and P. C. Woodland, "Modelling sub-phone insertions and deletions in continuous speech recognition," in *Proc. ICSLP-2000*, (Beijing, China), pp. IV:172–175, 2000.

[47] E. Harborg, *Hidden Markov models applied to automatic speech recognition*. PhD thesis, NTH (Norwegian Institute of Technology), 1990.

[48] P. A. Heeman, D. Cole, and A. Cronk, "The U.S. SpeechDat-Car data collection," in *Proc. EUROSPEECH-2001*, (Aalborg, Denmark), pp. 2031–2034, 2001.

[49] T. Holter, *Maximum likelihood modelling of pronunciation in automatic speech recognition*. PhD thesis, NTNU (Norwegian University of Science and Technology), 1997.

[50] T. Holter and T. Svendsen, "Incorporating linguistic knowledge and automatic baseform generation in acoustic subword unit based speech recognition," in *Proc. EUROSPEECH-97*, (Rhodes, Greece), pp. 1159–1162, 1997.

[51] T. Holter and T. Svendsen, "Maximum likelihood modelling of pronunciation variation," *Speech Communication*, vol. 29, pp. 177–191, 1999.

[52] X. Huang, A. Acero, and H.-W. Hon, *Spoken language processing: A guide to theory, algorithm, and system development*. Prentice Hall, 2001.

[53] J. J. Humphries and P. C. Woodland, "The use of accent-specific pronunciation dictionaries in acoustic model training," in *Proc. ICASSP-98*, (Seattle (WA), USA), pp. 317–320, 1998.

[54] J. J. Humphries and P. C. Woodland, "Using accent-specific pronunciation modelling for improved large vocabulary continuous speech recognition," in *Proc. EUROSPEECH-97*, (Rhodes, Greece), pp. 2367–2370, 1997.

[55] J. J. Humphries, P. C. Woodland, and D. Pearce, "Using accent-specific pronunciation modelling for robust speech recognition," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. 2324–2327, 1996.

[56] A. Høyland, *Statistisk metodelære*. Tapir forlag, (Trondheim, Norway), 1986.

[57] *The International Phonetic Alphabet*. [online], 1996. [cited 2002-03-01]. URL: `http://www2.arts.gla.ac.uk/IPA/ipachart.html`.

[58] F. Jelinek, "Workshops on large vocabulary conversational speech recognition at Johns Hopkins," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. Addendum 32–33, 1996.

[59] F. Jelinek, "Continuous speech recognition by statistical methods," *Proceedings of the IEEE*, vol. 64, pp. 532–556, 1976.

[60] F. T. Johansen, *Global discriminative modelling for automatic speech recognition*. PhD thesis, NTH (Norwegian Institute of Technology), 1996.

[61] B.-H. Juang and S. Katagiri, "Discriminative learning for minimum error classification," *IEEE Transactions on Signal Processing*, vol. 40, pp. 3043–3054, 1992.

[62] D. Jurafsky, W. Ward, Z. Jianping, K. Herold, Y. Xiuyang, and Z. Sen, "What kind of pronunciation variation is hard for triphones to model?," in *Proc. ICASSP-2001*, (Salt Lake City (UT), USA), pp. 577–580, 2001.

[63] D. Jurafsky and J. H. Martin, *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall, 2000.

[64] S. Katagiri, B.-H. Juang, and C.-H. Lee, "Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method," *Proceedings of the IEEE*, vol. 86, pp. 2345–2373, 1998.

[65] J. M. Kessens, H. Strik, and C. Cucchiarini, "A bottom-up method for obtaining information about pronunciation variation," in *Proc. ICSLP-2000*, (Beijing, China), pp. I:274–277, 2000.

[66] J. M. Kessens, M. Wester, and H. Strik, "Improving the performance of a Dutch CSR by modeling within-word and cross-word pronunciation variation," *Speech Communication*, vol. 29, pp. 193–207, 1999.

[67] F. Korkmazskiy and B.-H. Juang, "Discriminative training of the pronunciation networks," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, (Santa Barbara (CA), USA), pp. 137–144, 1997.

[68] F. Korkmazskiy and C.-H. Lee, "Generating alternative pronunciations from a dictionary," in *Proc. EUROSPEECH-99*, (Budapest, Hungary), pp. 491–494, 1999.

[69] H.-K. J. Kuo, E. Fosler-Lussier, H. Jiang, and C.-H. Lee, "Discriminative training of language models for speech recognition," in *Proc. ICASSP-2002*, (Orlando (FL), USA), pp. 325–328, 2002.

[70] K. Kvale, *Segmentation and labelling of speech.* PhD thesis, NTH (Norwegian Institute of Technology), 1993.

[71] P. Ladefoged, *A course in phonetics.* Harcourt Brace College Publishers, 1993.

[72] J. Laver, *Principles of phonetics.* Cambridge University Press, 1994.

[73] W. A. Lea, *Trends in speech recognition.* Prentice Hall, 1980.

[74] C.-H. Lee and Q. Huo, "On adaptive decision rules and decision parameter adaptation for automatic speech recognition," *Proceedings of the IEEE*, vol. 88, pp. 1241–1269, 2000.

[75] C.-H. Lee, C.-H. Lin, and B.-H. Juang, "A study on speaker adaptation of the parameters of continous density hidden Markov models," *IEEE Transactions on signal processing*, vol. 39, pp. 806–814, April 1991.

[76] C.-H. Lee, F. K. Soong, and K. K. Paliwal, *Automatic speech and speaker recognition: Advanced topics.* Kluwer, 1996.

[77] L. Lee and R. Rose, "A frequency warping approach to speaker normalization," *IEEE Transactions on speech and audio processing*, vol. 6, pp. 49–60, January 1998.

[78] L. Lee and R. C. Rose, "Speaker normalization using efficient frequency warping procedures," in *Proc. ICASSP-96*, (Atlanta (GA), USA), pp. 353–356, 1996.

[79] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language*, vol. 9, pp. 171–185, 1995.

[80] L. Mayfield Tomokiyo, "Lexical and acoustic modeling of non-native speech in LVCSR," in *Proc. ICSLP-2000*, (Beijing, China), pp. IV:346–349, 2000.

[81] L. Mayfield Tomokiyo, "Linguistic properties of non-native speech," in *Proc. ICASSP-2000*, (Istanbul, Turkey), pp. 1335–1338, 2000.

[82] D. McAllaster, L. Gillick, F. Scattone, and M. Newman, "Fabricating conversational speech data with acoustic models: A program to examine model-data mismatch," in *Proc. ICSLP-98*, (Sydney, Australia), pp. 1847–1850, 1998.

[83] R. Muhr, R. Höldrich, and E. Wächter-Kollpacher, "The pronouncing dictionary of Austrian German and the other major varieties of German – A phonetic resources database on the pronunciation of German," in *Proc. LREC-2002*, (Las Palmas de Gran Canaria, Spain), pp. 1284–1289, 2002.

[84] C. H. Nakatani and J. Hirschberg, "A corpus-based study of repair cues in spontaneous speech," *Journal of the Acoustical Society of America*, vol. 95(3), pp. 1603–1616, March 1994.

[85] *NIST Spoken Language Technology Evaluations*. [online], 2002. [cited 2002-07-01]. URL: `http://www.nist.gov/speech/tests/`.

[86] D. O'Shaughnessy, "Recognition of hesitations in spontaneous speech," in *Proc. ICASSP-92*, (San Francisco (CA), USA), pp. I:521–523, 1992.

[87] S. Oviatt, "Predicting spoken disfluencies during human-computer interaction," *Computer Speech and Language*, vol. 9, pp. 19–35, 1995.

[88] H. Printz and P. Olsen, "Theory and practice of acoustic confusability," *Computer Speech and Language*, vol. 16, pp. 131–164, 2002.

[89] *CALLHOME American English Lexicon (PRONLEX)*. [online description], 1995. [cited 2002-03-01]. URL: `http://morph.ldc.upenn.edu/Catalog/LDC97L20.html`.

[90] L. Rabiner and B.-H. Juang, *Fundamentals of speech recognition.* Prentice Hall, 1993.

[91] L. R. Rabiner and R. W. Schafer, *Digital processing of speech signals.* Prentice Hall, 1978.

[92] W. Reichl and W. Chou, "Decision tree state tying based on segmental clustering for acoustic modeling," in *Proc. ICASSP-98*, (Seattle (WA), USA), pp. 801–804, 1998.

[93] B. Resch, "Data driven pronunciation modeling for large vocabulary spontaneous speech recognition," Master's thesis, Graz University of Technology, 2002.

[94] M. Riley, W. Byrne, M. Finke, S. Khudanpur, A. Ljolje, J. McDonough, H. Nock, M. Saraclar, C. Wooters, and G. Zavaliagkos, "Stochastic pronunciation modelling from hand-labelled phonetic corpora," *Speech Communication*, vol. 29, pp. 209–224, 1999.

[95] M. D. Riley and A. Ljolje, *Automatic speech and speaker recognition: Advanced topics*, ch. Automatic generation of detailed pronunciation lexicons, pp. 285–301. Kluwer, 1996.

[96] B. D. Ripley, *Pattern recognition and neural networks.* Cambridge University Press, 1996.

[97] *SAMPA computer readable phonetic alphabet.* [online], 2000. [cited 2002-03-01]. URL: `http://www.phon.ucl.ac.uk/home/sampa/home.htm`.

[98] H. Schramm and X. L. Aubert, "Efficient integration of multiple pronunciations in a large vocabulary decoder," in *Proc. ICASSP-2000*, (Istanbul, Turkey), pp. 1659–1662, 2000.

[99] H. Schramm and P. Beyerlein, "Towards discriminative lexicon optimization," in *Proc. EUROSPEECH-2001*, (Aalborg, Denmark), pp. 1457–1460, 2001.

[100] T. Schultz and I. Rogina, "Acoustic and language modeling of human and nonhuman noises for human-to-human spontaneous speech recognition," in *Proc. ICASSP-95*, (Detroit (MI), USA), pp. 293–296, 1995.

[101] K. Shinoda and C.-H. Lee, "Structural MAP speaker adaptation using hierarchical priors," in *Proc. IEEE Workshop on Automatic*

Speech Recognition and Understanding, (Santa Barbara (CA), USA), pp. 381–388, 1997.

[102] B. Shneiderman, *Designing the user interface: Strategies for effective Human-Computer Interaction*. Addison-Wesley, 1998.

[103] E. Shriberg, "Disfluencies in Switchboard," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. Addendum 11–12, 1996.

[104] E. Shriberg and A. Stolcke, "Word predictability after hesitations: A corpus-based study," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. 1868–1871, 1996.

[105] R. Singh, B. Raj, and R. M. Stern, "Automatic generation of phone sets and lexical transcriptions," in *Proc. ICASSP-2000*, (Istanbul, Turkey), pp. 1691–1694, 2000.

[106] R. Singh, B. Raj, and R. M. Stern, "Structured redefinition of sound units by merging and splitting for improved speech recognition," in *Proc. ICSLP-2000*, (Beijing, China), pp. III:151–154, 2000.

[107] O. Siohan, T. A. Myrvoll, and C.-H. Lee, "Structural maximum *a posteriori* linear regression for fast HMM adaptation," *Computer Speech and Language*, vol. 16, pp. 5–24, 2002.

[108] M. Siu and M. Ostendorf, "Modeling disfluencies in conversational speech," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. 386–389, 1996.

[109] T. Sloboda and A. Waibel, "Dictionary learning for spontaneous speech recognition," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. 2328–2331, 1996.

[110] R. W. Sproat and J. P. Olive, "Text-to-speech synthesis," *AT&T Technical Journal*, vol. 74, pp. 35–44, 1995.

[111] A. Stolcke and E. Shriberg, "Automatic linguistic segmentation of conversational speech," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. 1005–1008, 1996.

[112] A. Stolcke and E. Shriberg, "Statistical language modeling for speech disfluencies," in *Proc. ICASSP-96*, (Atlanta (GA), USA), pp. 405–408, 1996.

[113] H. Strik, "Pronunciation adaptation at the lexical level," in *Proc. ISCA ITRW Adaptation methods for speech recognition*, (Sophia-Antipolis, France), pp. 123–130, 2001.

[114] H. Strik and C. Cucchiarini, "Modeling pronunciation variation for ASR: overview and comparison of methods," in *Proc. ESCA Workshop on Modeling Pronunciation Variation for Automatic Speech Recognition*, (Rolduc, the Netherlands), pp. 137–144, 1998.

[115] H. Strik, C. Cucchiarini, and J. M. Kessens, "Comparing the performance of two CSRs: How to determine the significance level of the difference," in *Proc. EUROSPEECH-2001*, (Aalborg, Denmark), pp. 2091–2094, 2001.

[116] T. Svendsen, K. K. Paliwal, E. Harborg, and P. O. Husøy, "An improved sub-word based speech recognizer," in *Proc. ICASSP-89*, (Glasgow, Scotland), pp. 108–111, 1989.

[117] G. Tajchman, E. Fosler, and D. Jurafsky, "Building multiple pronunciation models for novel words using exploratory computational phonology," in *Proc. EUROSPEECH-95*, (Madrid, Spain), pp. 2247–2250, 1995.

[118] D. Torre, L. Villarrubia, J. M. Elvira, and L. Hernandez-Gomez, "Automatic alternative transcription generation and vocabulary selection for flexible word recognizers," in *Proc. ICASSP-97*, (Munich, Germany), pp. 1463–1466, 1997.

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

[120] N. D. Warakagoda, *Nonlinear dynamical systems for automatic speech recognition*. PhD thesis, NTNU (Norwegian University of Science and Technology), 2001.

[121] M. Weintraub, K. Taussig, K. Hunicke-Smith, and A. Snodgrass, "Effect on speaking style on LVCSR performance," in *Proc. ICSLP-96*, (Philadelphia (PA), USA), pp. Addendum 16–19, 1996.

[122] M. Wester and E. Fosler-Lussier, "A comparison of data-derived and knowledge-based modeling of pronunciation variation," in *Proc. ICSLP-2000*, (Beijing, China), pp. I:270–273, 2000.

[123] M. Wester, J. M. Kessens, and H. Strik, "Pronunciation variation in ASR: Which variation to model?," in *Proc. ICSLP-2000*, (Beijing, China), pp. IV:488–491, 2000.

[124] M. Wolff, M. Eichner, and R. Hoffmann, "Automatic learning and optimization of pronunciation dictionaries," in *Proc. ISCA ITRW Adaptation methods for speech recognition*, (Sophia-Antipolis, France), pp. 159–162, 2001.

[125] P. C. Woodland, "Speaker adaptation for continuous density HMMs: A review," in *Proc. ISCA ITRW Adaptation methods for speech recognition*, (Sophia-Antipolis, France), pp. 11–19, 2001.

[126] P. C. Woodland, J. J. Odell, V. Valtchev, and S. J. Young, "Large vocabulary continuous speech recognition using HTK," in *Proc. ICASSP-94*, (Adelaide, Australia), pp. II:125–128, 1994.

[127] *Wall Street Journal speech database (WSJ).* [online description], 1993. [cited 2002-03-01]. URL: `http://morph.ldc.upenn.edu/Catalog/LDC94S13A.html`.

[128] S.-L. Wu, M. L. Shire, S. Greenberg, and N. Morgan, "Integrating syllable boundary information into speech recognition," in *Proc. ICASSP-97*, (Munich, Germany), pp. 987–990, 1997.

[129] Q. Yang and J.-P. Martens, "Data-driven lexical modeling of pronunciation variations for ASR," in *Proc. ICSLP-2000*, (Beijing, China), pp. I:417–420, 2000.

[130] S. Young, "A review of large-vocabulary continuous-speech recognition," *IEEE Signal Processing Magazine*, pp. 45–57, September 1996.

[131] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *HTK Version 3.0.* [online description], 2000. [cited 2002-03-01]. URL: `http://htk.eng.cam.ac.uk/`.

[132] V. Zue, "Conversational interfaces: Advances and challenges," in *Proc. EUROSPEECH-97*, (Rhodes, Greece), pp. KN9–KN18, 1997.