Tampere University of Technology
Department of Information Technology

Matti Vihola

Dissimilarity Measures for Hidden Markov Models and Their Application in Multilingual Speech Recognition

Master of Science Thesis

Subject approved by the Department Council
on November 14th 2001

Examiners:	Prof. Jukka Saarinen
	Prof. Jaakko Astola
	Dr. Tech. Petri Salmela
	Lic. Tech. Janne Suontausta

Preface

This thesis is a part of the User-Oriented Information Technology (USIX) technology program directed by the Finnish National Technology Agency (Tekes). The work has been carried out at the Institute of Signal Processing, Department of Information Technology, Tampere University of Technology; in collaboration with Nokia Research Center and Department of Phonetics, University of Turku.

I wish to thank Dr. Tech. Petri Salmela for advice and persistent guidance throughout the writing process. I am also grateful for Professor Jukka Saarinen, Professor Jaakko Astola and Lic. Tech. Janne Suontausta for advice and contructive criticism.

I express my gratitude to all the members of the multilingual speech recognition project both at the Institute of Signal Processing and at the Institute of Digital and Computer Systems. Furthermore, thanks goes to all the colleague researchers in the Audio Research Group. It has been a pleasure working with you all!

Tampere, 6th May 2002

Matti Vihola

Opiskelijankatu 4 E 289
33720 Tampere
Tel. (+358)400-908508

Contents

Tiivistelmä

TAMPEREEN TEKNILLINEN KORKEAKOULU
Tietotekniikan osasto
Signaalinkäsittelyn laitos
VIHOLA, MATTI: Erilaisuusmitat piilo-Markov -malleille ja niiden käyttö monikielisessä puheentunnistuksessa.
Diplomityö, 51 s.; liite, 3 s.
Tarkastajat: Prof. Jukka Saarinen, Prof. Jaakko Astola, TkT Petri Salmela, TkL Janne Suontausta (Nokia Research Center).
Rahoittajat: Teknologian kehittämiskeskus (Tekes), Nokia Research Center.
Toukokuu 2002
Avainsanat: Erilaisuusmitta, etäisyysmitta, Kullback-Leibler -divergenssi, monikielisyys, piilo-Markov -malli, puheentunnistus, ryhmittely, sekaannusmatriisi.

Automaattista puheentunnistusta voidaan jo käyttää joihinkin kuluttajasovelluksiin. Intensiivistä tutkimusta tarvitaan kuitenkin vielä esimerkiksi monikielituen kehittämisessä. Tämä työ käsittelee monikielistä akustista mallinnusta, erityisesti monikielisten akustisten mallien määräämistä. Eräs menetelmä monikielisen puheentunnistusjärjestelmän toteuttamiseksi pohjautuu valmiiksi koulutettuun joukkoon kieliriippuvia tunnistusjärjestelmiä. Näiden järjestelmien akustisten mallien välinen erilaisuus mitataan käyttäen tiettyä erilaisuusmittaa, ja sitä hyväksikäyttäen ryhmitellään akustiset mallit. Tarkoituksena on saada aikaiseksi pieni, mutta kattava joukko akustisia malleja, jotka jaetaan eri kielten kesken.

Tässä työssä käydään läpi erilaisuusmittoja piilo-Markov -malleille (HMM). Mittoihin, joita voidaan käyttää akustisten mallien ryhmittelyyn, kiinnitetään erityistä huomiota. Kaikki työssä esitetyt mitat pohjautuvat joko sekaannusmatriisiestimaattiin tai Kullback-Leibler (KL) -divergenssin estimaattiin tai sen approksimaatioon. Työssä luodaan katsaus edellä mainittuihin mittoihin, ja esitetään muunnettuja menetelmiä estimaattien tarkentamiseksi. Työssä esitellään lisäksi kaksi approksimaatiota KL -divergenssistä, jotka voidaan esittää suljetussa muodossa. Näiden approksimaatioiden laskennalliset kustannukset ovat erittäin alhaiset, mutta niitä voidaan soveltaa ainoastaan HMM:iin, joilla on multinormaalit havaintojakaumat, sekä tietynlainen mallirakenne.

Koejärjestelyssä erilaisuusmittoja tarkasteltiin monikielisten akustisten mallien määrittämisessä. Kokeissa monikieliset tunnistimet rakennettiin viidelle kielelle, jotka olivat englanti, espanja, italia, saksa ja suomi. Monikielinen puheentunnistusjärjestelmä, jossa on 64 akustista mallia, opetettiin jokaisen erilaisuusmitan tuottaman akustisten mallien ryhmittelyn perusteella. Näitä tunnistusjärjestelmiä verrattiin sekä keskenään, että monikielisiin järjestelmiin, joissa akustisten mallien ryhmittely ja määrittely oli suoritettu foneettisen tiedon pohjalta. Monikielisten tunnistusjärjestelmien tunnistustarkkuus arvioitiin puhujariippumattomassa irrallisten sanojen tunnistuksessa. Kieliriippuvilla järjestelmillä keskimääräinen tunnistusprosentti sanoille oli noin 89%. Monikielisten järjestelmien tunnistusprosentti vaihteli välillä 82–84%. Erot monikielisten järjestelmien välillä olivat siten pieniä. Monikielisiä järjestelmiä kokeiltiin myös kahden uuden kielen, ranskan ja ruotsin, tunnistamisessa. Näiden kielten kieliriippuville järjestelmille keskimääräinen tunnistusprosentti oli noin 84%. Vastaava prosentti monikielisille järjestelmille vaihteli välillä 60–64%.

Abstract

TAMPERE UNIVERSITY OF TECHNOLOGY
Department of Information Technology
Insitute of Signal Processing

VIHOLA, MATTI: Dissimilarity Measures for Hidden Markov Models and Their Application in Multilingual Speech Recognition.
Master of Science thesis, 51 pages; Appendix, 3 pages.
Examiners: Prof. Jukka Saarinen, Prof. Jaakko Astola, Dr. Tech. Petri Salmela, Lic. Tech. Janne Suontausta (Nokia Research Center).
Funding: Finnish National Technology Agency (Tekes), Nokia Research Center.
May 2002
Keywords: Clustering, confusion matrix, dissimilarity measure, distance measure, hidden Markov model, Kullback-Leibler divergence, multilinguality, speech recognition.

Although automatic speech recognition (ASR) technology is mature enough for some consumer products, intensive research is still needed to obtain e.g. the support for multiple languages. This thesis contributes to multilingual (ML) acoustic modeling, especially to the techniques that can be used for defining a set of ML acoustic models for ML ASR system. One starting point for the development of such a system is first to train a set of language dependent (LD) ASR systems. Based on some measure of dissimilarity between the obtained LD acoustic models, the objective is to reduce the acoustic model set into a compact set of models that are shared across the languages.

In this thesis, the dissimilarity measures for hidden Markov models (HMMs), especially those that are applicable for the clustering of acoustic HMMs, are covered. All the measures are based either on a confusion matrix estimate, or on an estimate or an approximation of the Kullback-Leibler (KL) divergence. This thesis reviews a number of such methods, and also proposes some modifications to get more accurate KL divergence and confusion matrix estimates. In addition, two closed-form approximations of the KL divergence are proposed. These closed-form approximations have very low computational cost, but they are restricted for HMMs with Gaussian emission densities and employ assumptions of the HMM topology.

The dissimilarity measures were experimented in the definition of the set of ML phone models. One ML ASR system having 64 phone models was trained for each phone cluster definition obtained from the corresponding dissimilarity measure. The systems were trained for five languages: English, Finnish, German, Italian and Spanish. The obtained ASR systems were compared both against each other, and to the alternative ML ASR systems having phone clusters determined solely by expert knowledge. The performances of these multilingual recognizers were evaluated in the task of speaker independent isolated word recognition. The average word recognition rate (WRR) of the baseline LD recognition systems was approximately 89%, while the average WRRs of the ML systems varied between 82–84%. Small differences were observed in the recognition accuracies of the different ML recognition systems. The ML systems were tested also with two new languages, French and Swedish. In the experiments, the average WRR was 84% for the baseline LD systems, while the average WRR of the ML ASR system dropped to 60–64%.

List of Acronyms

ASR	Automatic speech recognition
BW	Baum-Welch
DCT	Discrete cosine transform
DFT	Discrete Fourier transform
EM	Expectation-maximization
GMM	Gaussian mixture model
HMM	Hidden Markov model
HTK	Hidden Markov Model Toolkit
IPA	International Phonetic Association
KL	Kullback-Leibler
LD	Language dependent
LID	Language identification
MAP	Maximum *a posteriori*
MC	Monte-Carlo
MFCC	Mel-frequency cepstral coefficient
ML	Multilingual
MLE	Maximum likelihood estimate
MLLR	Maximum likelihood linear regression
PDF	Probability density function
SAMPA	Speech Assessment Methods Phonetic Alphabet
SD	Speaker dependent
SI	Speaker independent
SIL	Silence model
SP	Short pause model
WRR	Word recognition rate

List of Symbols

General notations

$\mathbf{A}, \mathbf{B}, \mathbf{C} \ldots$	Matrices
$\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \ldots$	Vectors
$a, b, c \ldots$	Scalars
$\mathcal{A}, \mathcal{B}, \mathcal{C} \ldots$	Sets
a_{ij}	Element of matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	Inverse of matrix $\mathbf{A}$
$\|\mathbf{A}\|$	Determinant of matrix $\mathbf{A}$
$\boldsymbol{b}^T$	Transpose of vector $\boldsymbol{b}$
$\operatorname{card} \mathcal{A}$	Cardinality, i.e. number of elements in the set $\mathcal{A}$
$\dim \boldsymbol{b}$	Dimension of vector $\boldsymbol{b}$
$E\left\{f(\mathbf{O}^\lambda)\right\}$	Expectation of function f of a random variable $\mathbf{O}^\lambda$ with distribution λ
$f_{\mathcal{D}}(\cdot;\theta)$	The PDF of distribution $\mathcal{D}$ with parameters θ
$t(\boldsymbol{x}, \boldsymbol{y})$	Tanimoto similarity ratio of vectors $\boldsymbol{x}$ and $\boldsymbol{y}$
$\operatorname{tr} \mathbf{A}$	Trace of matrix $\mathbf{A}$
$x[n]$	n:th sample of a discrete-time signal x
$\boldsymbol{\mu}$	Mean vector of a random distribution
$\boldsymbol{\Sigma}$	Covariance matrix of a random distribution
$\{x : \operatorname{cond}(x)\}$	A set of all such elements x that fulfill the condition $\operatorname{cond}(\cdot)$
$[c, d]^T$	A vector consisting of elements c and d
(a, b)	Ordered set containing the elements a and b

Mel-frequency cepstral coefficients

C_i	The ith cepstral coefficient of a frame
D	Dimension of an observation feature vector $\boldsymbol{o}$
E	The energy estimate of a frame
$\mathbf{O}$	Observation vector sequence
$\boldsymbol{o}_t$	Observation vector occurring at time instant t
ΔC	First time derivative coefficient corresponding C
$\Delta^2 C$	Second time derivative coefficient corresponding C

Hidden Markov models

$\mathbf{A}$	The transition matrix of a HMM
$b_j(\boldsymbol{o})$	The likelihood of emission distribution of state j of a HMM

$P(\mathbf{O})$	Likelihood of observation sequence $\mathbf{O}$
$P(\lambda)$	*A priori* probability of model λ
$P(\mathbf{O} \mid \lambda)$	Conditional likelihood of $\mathbf{O}$ given λ
$P^*(\mathbf{O} \mid \lambda)$	The likelihood of the most likely state sequence of $\mathbf{O}$ given λ
$\boldsymbol{q}$	State sequence vector
q_t	State at time instant t
$\boldsymbol{q}^*$	The most likely state sequence
$\mathcal{Q}^{(T)}$	Set of state sequences of length T
$Q(\lambda, \widehat{\lambda})$	Baum's auxiliary function
w_{jk}	The weight of the kth mixture component of a GMM of the state j of a HMM
$\alpha_t(j)$	Forward variable of state j at time instant t
$\beta_t(i)$	Backward variable of state i at time instant t
$\gamma_t(j,k)$	The *a posteriori* probability of an observation at time instant t originating from the kth mixture component of the state j of a HMM
$\delta_t(i)$	The accumulated likelihood of the most likely state sequence up to state i at time instant t
$\delta^*_m(\kappa)$	The accumulated likelihood of the most likely state sequence of model κ
Θ	Parameter set of all emission densities of a HMM
θ_{jk}	Parameters of a mixture density component k of the state j of a HMM
κ	The parameter set of a HMM
λ	The parameter set of a HMM
$\lambda \oplus \kappa$	The concatenated HMM consisting of λ and κ
$\xi_t(i,j)$	The probability of being in state i at time t and state j at time $t+1$
$\boldsymbol{\pi}$	Initial state distribution of a HMM
$\psi_t(i)$	The previous state of the partial most likely state sequence up to state i at time instant t

Dissimilarity measures

$\mathbf{C}$	Confusion matrix
$\widehat{\mathbf{C}}_X$	The estimate of the confusion matrix according to the estimation method X
$\mathbf{D}$	Dissimilarity matrix
$I(\lambda : \kappa)$	The directed KL divergence from distribution λ to κ
$\widehat{I}_X(\lambda : \kappa)$	The estimate of the directed KL divergence from λ to κ according to the estimation method X
$J(\lambda, \kappa)$	The KL divergence between distributions λ and κ
$\mathrm{LLR}(\mathbf{O} \mid \lambda, \kappa)$	Logarithmic likelihood ratio function of $\mathbf{O}$ given the models λ and κ
$\mathbf{S}$	Similarity matrix
$\mathrm{SC}(\mathbf{O} \mid \lambda)$	Likelihood score function of $\mathbf{O}$ given the model λ

Chapter 1

Introduction

An infant mimics speech sounds at a very young age, and learns to speak without explicit teaching quickly. A recent reasearch has proved an interesing result: even a newborn child (1-7 days old) can distinguish between different vowel sounds, while being fast asleep [Cheour et al. 2002]. If speech is such a built-in communication method for humans, why cannot machines be operated by voice commands?

Automatic speech recognition (ASR), i.e. recognition of human speech with a machine, has been researched since the 1950s [Gold and Morgan 2000]. Such a natural thing for humans as speech recognition has showed to be a very challenging task for machines. Despite the difficulties, the persistent research in the area ever since the 1950s has yielded marks of progress. Today, ASR technology has been applied to consumer products, e.g. name dialers in mobile handsets, telephone number queries and train timetable retreival systems. Some very fundamental issues have risen in employing ASR for the purposes of such applications. These include e.g. robustness, espcially for noise and changing acoustic conditions, and support for multiple languages. The speech interface is natural and convenient to use only when it is robust, and can be operated with the native language of the user. Therefore, a comprehensive support for languages is one of the key characteristics that the speech-driven applications need to fulfill before a true widespread acceptance can be achieved.

The spoken languages apparently share common acoustic features. This is evident as the source of the speech signal is always the human speech production system, regardless of the language. Different sounds that are produced by this system while speaking are known as phonemes. Phonemes are the atomic units of speech, or language, meaning that by changing a phoneme in a word, the meaning of the word can change. For example, by changing the first phoneme in the word "pet", we get the word "set"[1]. The phonemes can be classified by their articulatory characteristics, e.g. to resonants and obstruents (whether the vocal tract is blocked or not), consonants and vowels, and so on[2]. Speech recognition systems are often built using the phonemes as modeling units. One phoneme, or allophone[3], is represented with one acoustic model, most often a hidden Markov model (HMM).

This thesis concerns finding the phonemes, or allophones, that are similar to each other over a set of languages. This information has been employed previously in the definition

[1] Interestingly, in English, the change of the written form of a word doesn't necessarily affect the pronunciation, i.e. the phonemic representation. For example, the phonemic content of the word "sea", /siː/, is identical to the word "see". Conversely, the phonemic content of the written word "read" depends on the tense, either /riːd/ or /red/.

[2] Such categorization of phonemes is shown in Figure A.2 in Appendix A.

[3] Allophone is a phoneme in a certain context. For example, the two allophones of the phoneme /l/ in the words "feeling" and "alarm" sound rather different.

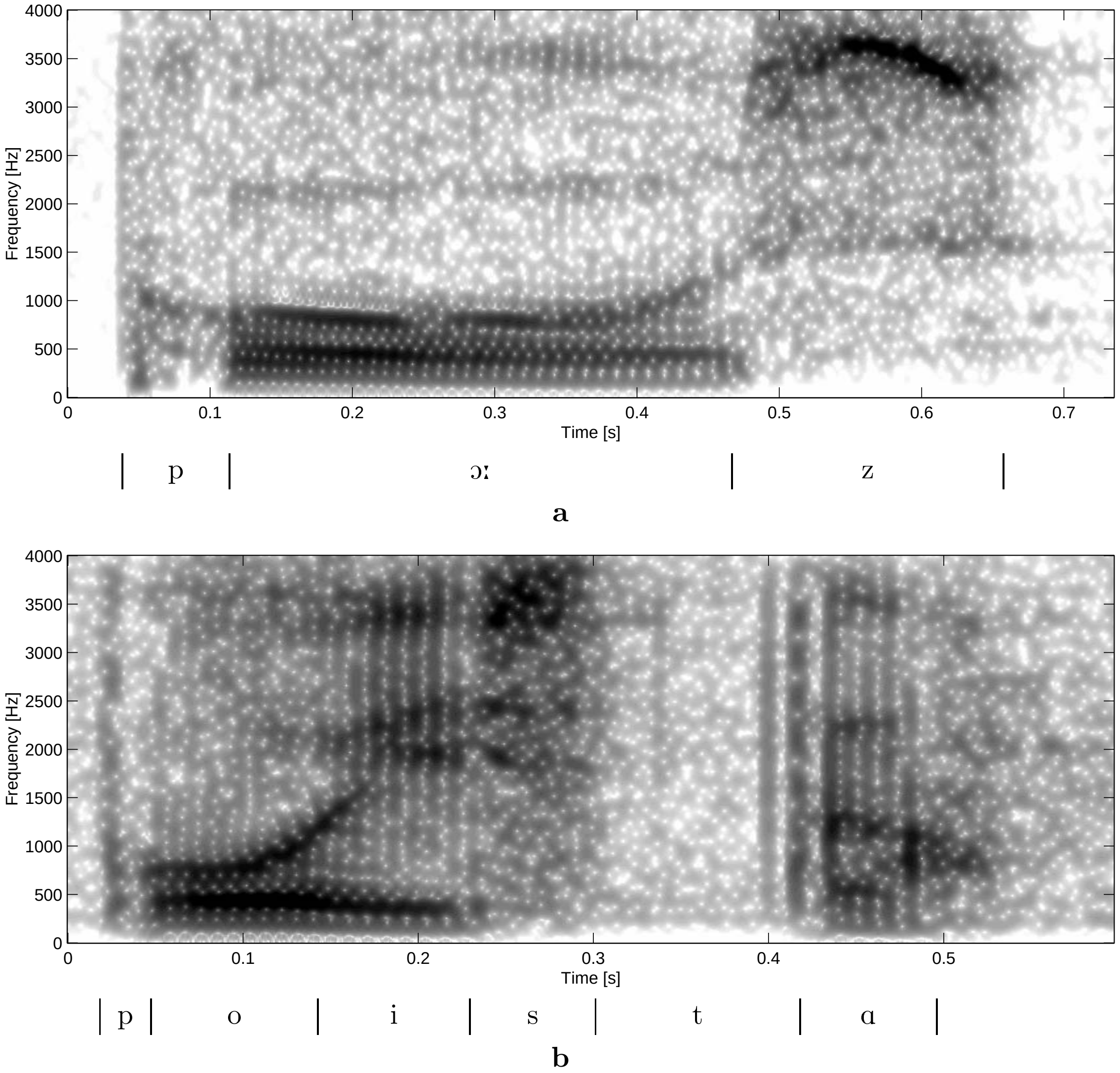

Figure 1.1: (a) Spectrogram of the English word "pause" /pɔːz/ and (b) the Finnish word "poista" /poistɑ/.

of a set of multilingual (ML) phone HMMs [Köhler 2000]. This means in practice that the phonemes of a set of languages are grouped, or clustered, such that the number of clusters is smaller than the original number of language dependent phonemes. The Figure 1.1 shows spectrograms of two uttered words, English "pause" and Finnish "poista" ("remove" in English). In this example, the English /p/ and Finnish /p/ could be represented with a common ML phone[4] model /p/. In addition, two pairs of similar phonemes can be found: English /ɔː/ and Finnish /o/; and English /z/ and Finnish /s/. The number of the ML phone models could be reduced from nine to six in this example, if the above tying of the phonemes were applied.

The search for these similar phonemes has been previously performed by comparing the acoustic models, HMMs, that are used in the speech recognition system [Andersen et al. 1994, Köhler 2001]. The proximity of the HMMs has been measured by employing some

[4] In this thesis, "phone" refers to a ML acoustic model that represents a set of LD phoneme models.

dissimilarity measure, which measures how discriminating two HMMs are. This thesis reviews the measures introduced for this task. In addition, four modified measures are proposed to give an increased accuracy for the previously proposed measures, and two measures having low computational cost are introduced. The behavior of the measures is compared, when they are applied in the task of phoneme model clustering.

This thesis consists of the following chapters. In Chapter 2, the methods used in modern speech recognition are discussed. This discussion covers the methods often used for the preprocessing of the speech signal and statistical pattern recognition, which are Mel-frequency cepstral coefficients (MFCCs) and HMMs, respectively. In Chapter 3, an overview of the issues covered by the research in the field of multilingual speech recognition is given. The main topic of the thesis, the dissimilarity measures for HMMs, is discussed in Chapter 4. The previously proposed methods are reviewed, and some novel techniques are proposed. Chapter 5 covers the experiments before the concluding remarks given in Chapter 6.

Chapter 2

Automatic Speech Recognition

Machine recognition of human speech, often referred to as automatic speech recognition (ASR), has been investigated from the early 1950s [Gold and Morgan 2000]. The research in the area has been rather intense since the late 1970s. During 1980s, statistical modeling, namely the hidden Markov models (HMMs), started to replace the earlier template-matching-based methods in ASR [Rabiner 1993]. Today, the most successful ASR systems are based on these statistical models, of course flavored with lots of adjustments and improvements.

In this chapter, a brief overview of the modern ASR systems is given. First of all, the categorization of the speech recognition tasks is discussed in Section 2.1. After that, Section 2.2 outlines the structure of a typical ASR system. This system can be roughly divided into two main units which are the feature extraction unit, and the pattern classifier unit. These two units are often also referred as the front-end and the back-end, respectively. The derivation of the typical Mel-frequency cepstral coefficient (MFCC) features produced by the front-end unit is covered in Section 2.3. After that, the techniques involved in the pattern classifier unit are reviewed briefly in Section 2.4. These include the concept of HMMs and some essential training and decoding algorithms. Finally, the speaker adaptation techniques, especially maximum likelihood linear regression (MLLR), are discussed in Section 2.5.

2.1 Classification of Speech Recognition Tasks

Speech recognition systems are nowadays implemented in a very application dependent manner. This is mainly due to limited resources and the many obstacles still faced in the field of speech recognition. The obvious consequence of application oriented implementations is that they may not work well in some other application domain. In the following, some characteristics are explained that discriminate the different ASR tasks [Adda-Decker 2001, Kiss 2001, Laurila 2000, Viikki 1999, Waibel et al. 2000].

Vocabulary size: Today, the systems with small vocabulary can distinguish some tens of words. The medium vocabulary size ranges from hundred to thousand words and the large vocabulary systems range up to 100000 words [Viikki 1999]. The number of words in the vocabulary affects both the recognition accuracy and the speed of the recognition process. In addition, the choice of the acoustic units depends on the size of the vocabulary. The whole word models can be used as acoustic units in a small vocabulary ASR task, while these units are replaced with subword models as the vocabulary becomes larger. The typical subword units represent phonemes, allophones or syllables. Furthermore, the type of the vocabulary can be fixed, such as in a digit recognition, or dynamic, such as in a name dialer in a mobile handset.

Continuous, connected or isolated word recognition: In an isolated word recognition task, the recognition system forms a hypothesis of the uttered word or phrase as a single entity. Connected word recognition can be considered a step forward, since the user is allowed to speak several words at a time, but a short pause has to be left between the words. However, the word boundaries are hard to find in natural speech due to coarticulation, which makes the task of continuous speech recognition challenging. Coarticulation of words means that they are articulated consecutively after each other with no inter-word pause. A language model is not necessarily needed in isolated word recognition, but in continous ASR, language modeling is an important issue.

Speaker dependent or independent systems: The speaker independency of an ASR system is a desired feature in general. A speaker independent (SI) system can cope with different speakers, speaking styles and even dialects. However, the speaker dependent (SD) ASR systems outperform the SI systems in recognition accuracy because the acoustic models of the SD system have to cope with smaller variability of acoustic features in speech [Viikki 1999]. The direct training of SD recognition system is unfeasible in most cases, as hours of speech material may be needed from the target speaker[1]. Speaker dependent speech recognition system can be obtained by adapting SI recognition system either continuously, i.e. performing the adaptation online, or with relatively small adaptation data set. Such adaptation techniques are discussed more in detail in Section 2.5.1.

Language dependent or multilingual systems: Typically the ASR systems are limited to only a single language. The support for multiple languages in ASR systems has emerged as the first widely spread applications have been introduced [Waibel et al. 2000]. The multilinguality of an ASR system can be viewed somehow analogous to speaker independency. The multilingual speech recognition is discussed in Chapter 3 in more detail.

Environmental robustness: Many speech recognition applications work well in laboratory, but they have difficulties in realistic environments. Their applicability can collapse very quickly instead of graceful deterioration when the signal is corrupted e.g. with background noise or microphone distortion. In general, the ASR systems work well in conditions similar to the conditions of the acoustic material used during training. The real world ASR applications demand the recognition system to be robust for various acoustic and noise conditions. The techniques that are used in noise robust speech recognition can be grouped into the following categories: noise robust feature vectors, techniques compensating noise from feature vectors and model parameter adaptation and compensation techniques [Furui 1995, Junqua 2000].

The field of the speech recognition research can be stated extensive. Isolated word recognition task with a small vocabulary is a rather straightforward pattern classification task. However, considering the recognition of continuous conversational speech with large vocabulary, the semantics and pragmatics involved in the discussion must be included to achieve accurate recognition performance. The scope of this thesis covers pattern classification and acoustic modeling of speech. Language modeling and understanding are not discussed.

[1] Practically, the SD system is feasible to build for a small vocabulary ASR task.

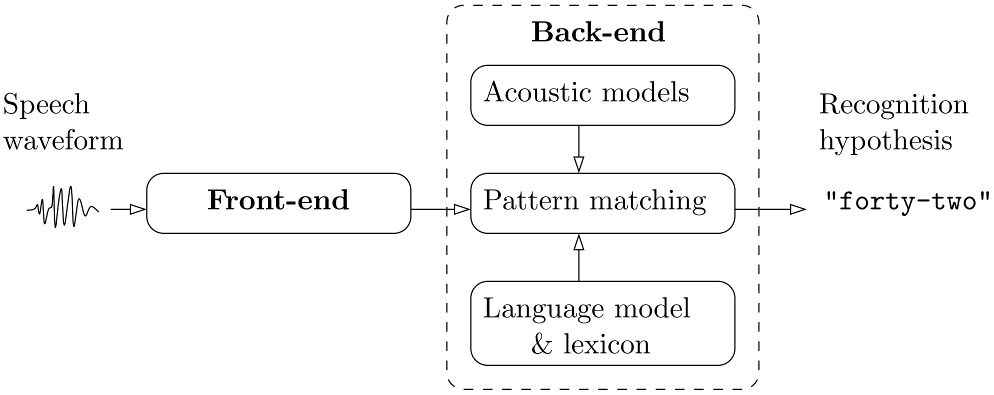

Figure 2.1: Block diagram of a typical speech recognition system.

2.2 Typical Structure of a Modern Speech Recognition System

Most of the modern speech recognition systems can be divided into front-end and back-end processing units [Rabiner 1993]. They are referred also according to their function as feature extraction and pattern classifier units. The structure of a typical speech recognition system is depicted in Figure 2.1. The front-end and back-end processing units are usually very independent of each other, and can thus be implemented separately. The front-end unit can be viewed as “ears” of the speech recognition system. In the human ear, the changes in the air pressure are converted into a stream of properly coded neural impulses [Rossing 1990]. These impulses contain all the information from the acoustic signal that is received by ears. The information is processed further in the auditory cortex. Similarly, the front-end signal processing unit of a speech recognition system converts the input speech waveform into a stream of feature vectors. These vectors contain the information that is relevant to the speech recognition process. Usually, each feature vector describes the spectral content of the speech signal at a particular time instant. The pattern classifier unit forms a recognition hypothesis based on the sequence of feature vectors $\mathbf{O} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T)$ produced by the front-end unit. The items in the vocabulary are compared against the acoustic evidence $\mathbf{O}$, and the best matching item is chosen as the recognition hypothesis.

2.3 Feature Extraction Unit

Mel-frequency cepstral coefficients (MFCCs) are the most commonly used acoustic features in speech recognition. They are computationally efficient and found good in practice [Karjalainen 1999]. A block diagram of the derivation of MFCCs is depicted in Figure 2.2. The functionality of each of these blocks is reviewed below.

First of all, the digitized[2] input speech waveform $x[n]$ is fed into a digital finite impulse response (FIR) filter of the form

$$y[n] = x[n] - \eta x[n-1] \tag{2.1}$$

where $0 < \eta < 1$ is a constant. The value of η is usually chosen between 0.90 and 1.00 in ASR applications. This pre-emphasis filter is high-pass type, and is used to flatten the

[2] The sampling frequency of the signal is usually 8kHz or 16kHz in speech recognition applications.

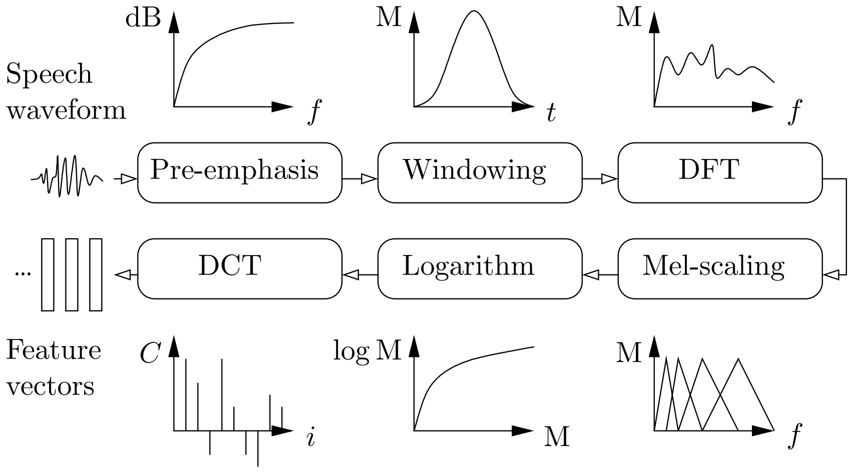

Figure 2.2: Block diagram of MFCC feature extraction. The graphs beside of DFT and DCT blocks represent the form of the signal at the current phase. The other graphs visualize the transforms carried out to the signal in the corresponding block. The symbols f, t and M correspond to frequency, time, and magnitude, respectively. The figure at the bottom left shows the resulting MFCCs C_i for a particular time instant.

input speech spectrum and discard the low-frequency components before the frequency analysis [Deller et al. 2000]. These low-frequency components refers to the frequency band $0 < f < 90$Hz not containing speech information.

The next block in Figure 2.2, i.e. windowing, has essentially two functions. Firstly, the input waveform is blocked into fixed length frames of length around 25ms. These frames usually overlap, and a typical interval between the successive frames is 10ms. After frame blocking, a window function and the following signal processing steps are applied to each frame. The window function weights the samples of the frame such that the effect of discontinuities is minimized and the spectral estimate is smoothed [Stoica and Moses 1997]. The typical weighting scheme used is the Hamming window function

$$z[n] = w[n]y[n] = \left[0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right)\right] y[n] \tag{2.2}$$

where N is the length of the window. Next block in Figure 2.2 performs a discrete Fourier transform (DFT) to the windowed piece of a signal derived from the previous block. This results in a short-time energy spectrum estimate in the particular time instant. To reduce the amount of data, only the energies of Mel-scaled frequency bands are preserved. The Mel-scale is one of the psychoacoustical frequency mappings that are supposed to match the nonlinear frequency selectivity of the basilar membrane in human ear. This mapping from Hz into Mel-scale can be given as [Davis and Mermelstein 1980]

$$f_{\mathrm{Mel}} = 2595\log_{10}\left(1 + \frac{f}{700}\right) \tag{2.3}$$

where f_{Mel} is frequency in Mel-scale and f frequency in Hz. In ASR applications, the number of the Mel-scaled frequency bands is usually around 20. The energies of these bands are calculated from the magnitude spectrum using triangular window functions such as shown in Figure 2.2. The human ear works nonlinear not only with respect to frequencies, but also with the signal power. The perceived power of the audio signal, i.e. loudness, is roughly logarithmic compared to the power of the signal [Rossing 1990]. This is why a logarithm of each of the bandwise energies is taken.

In order to obtain MFCCs, the last block in Figure 2.2 performs the discrete cosine transform (DCT) to the logarithmic energy estimates of the Mel-scaled frequency bands [Deller et al. 2000]. Typically only the first 13 DCT coefficients are preserved to form the basis of a feature vector. The DCT reduces correlation between the elements of the feature vector, which is useful in the back-end processing[3]. Usually the zeroth MFCC, C_0, is replaced with an energy estimate E of the frame. This energy estimate is considered less noisy than C_0. An inter-frame context-dependency is also often supplemented to the feature vector by calculating the first and second time derivative coefficients of the consecutive MFCC-based feature vectors [Furui 1986]. Finally, the resulting feature vector has the form

$$\boldsymbol{o} = \left[E, C_1, \cdots C_L, \Delta E, \Delta C_1, \cdots \Delta C_L, \Delta^2 E, \Delta^2 C_1, \cdots \Delta^2 C_L\right]^T \tag{2.4}$$

where E is the energy and C_i the ith cepstral coefficient of the frame. In addition, the number of cepstral coefficients in the feature vector is denoted with L and the first and second derivative coefficients are denoted with Δ and Δ^2, respectively. The dimension of this type of feature vector is $D = 3(L+1)$. Most often $D = 39$ which means that $L = 12$, i.e. the feature vector contains 12 static cepstral coefficients and an energy estimate. To enhance the recognition rates and robustness against different acoustic conditions, various feature vector normalization techniques have been presented [Hariharan 2001, Viikki and Laurila 1998].

2.4 Hidden Markov Models

The purpose of the back-end unit is to form a recognition hypothesis based on the acoustic evidence $\mathbf{O}$ derived from the front-end unit. The statistical formulation of the problem can be written as follows [Jelinek 1998]

$$\widehat{W} = \arg\max_{W} P(W \mid \mathbf{O}) \tag{2.5}$$

where $\widehat{W}$ is the recognition hypothesis, i.e. the chosen vocabulary item[4]. The HMMs provide an efficient means to compute the Equation (2.5). According to the Bayes' formula, the Equation (2.5) can be written as follows

$$\widehat{W} = \arg\max_{W} \frac{P(\mathbf{O} \mid W)P(W)}{P(\mathbf{O})} = \arg\max_{W} P(\mathbf{O} \mid W)P(W) \tag{2.6}$$

The likelihood $P(\mathbf{O} \mid W)$ is obtained by evaluating $P(\mathbf{O} \mid \lambda_W)$ where λ_W is the word HMM corresponding to vocabulary word W. The probability of the word $P(W)$ is determined by the language model. If the recognition system consists of subword acoustic modeling units, the combined word HMMs λ_W corresponding the vocabulary items W are constructed from the subword HMMs according the phonetic transcriptions determined in the lexicon.

The application of HMMs in speech recognition is based on two assumptions of the speech signal [Rabiner 1993]

[3] This is because the observation densities are often modeled as Gaussian mixture models with diagonal covariance matrices.

[4] The vocabulary item can be e.g. word or sentence.

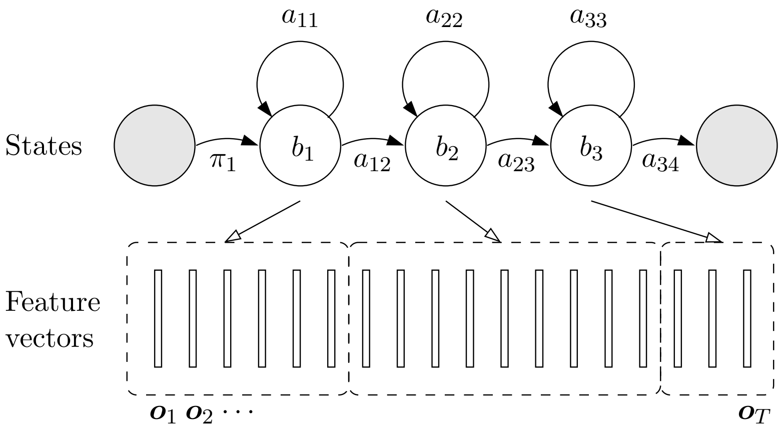

Figure 2.3: Graph of a HMM with three emitting states and an example feature vector sequence of this generative model. The gray circles denote non-emitting states. The type of this model is feedforward meaning that the only transitions allowed from each state are either self-transition or transition to the successor state. The symbols used in the figure are explained in Section 2.4.1.

- Speech signal is piecewise stationary, i.e. it can be segmented such that the stochastic characteristics of the signal do not change during each segment.
- The adjacent samples of the process, i.e. adjacent feature vectors, are independent of each other. This suggests that no inter-frame correlation exists.

These assumptions mean that the feature vectors are assumed to originate from a process exemplified in Figure 2.3. The assumptions are quite restrictive, and cannot be stated to hold in general. Anyhow, since HMMs provide computationally efficient training and decoding algorithms, and seem to work rather well in practise, they are used extensively in ASR [Rosti and Gales 2001].

2.4.1 HMM Representation

An example of a HMM with three emitting states is depicted in Figure 2.3. It is made up of states and transitions between these states. Each state is associated with an emission probability density function (PDF) $b_j(\boldsymbol{o})$ except the first and last state of the model, which are non-emitting. The non-emitting states do not generate outputs, and are used in the word model composition described in Figure 2.4. The probability of the current state of the model at a particular time instant depends only on the state at the preceeding time instant. Therefore, the HMMs consist of a Markov chain and state-dependent emission PDFs.

The HMMs contain essentially three parameters of which the two parameters $\boldsymbol{\pi}$ and $\mathbf{A}$ define the Markov process in a HMM. Firstly, the vector $\boldsymbol{\pi}$ contains the initial state distributions for each state. In other words, π_i is the probability of initially being in state i, i.e. $\pi_i = P(q_1 = i)$. The state-transition matrix $\mathbf{A}$ contains the transition probabilities from state i to state j, $P(q_t = j \mid q_{t-1} = i)$, as its elements a_{ij}. Therefore matrix $\mathbf{A}$ defines the structure of the HMM. For example, the transition matrix of the HMM in Figure 2.3 has non-zero values only near the diagonal.

The third parameter in a HMM are the state-dependent emission PDFs $b_j(\boldsymbol{o})$, which define the likelihood that a feature vector $\boldsymbol{o}$ was generated by the state j. The most common case is that these emission PDFs are continuous and of mixture type, i.e.

$$b_j(\boldsymbol{o}) = \sum_{k=1}^{M_j} w_{jk} f_{\mathcal{D}}(\boldsymbol{o}; \theta_{jk}) \tag{2.7}$$

where M_j is the number of mixture components, and w_{jk} the weight of the k:th mixture component density of the state j. The mixture component weights must satisfy the condition

$$\sum_{i=1}^{M_j} w_{ji} = 1 \qquad \text{with} \qquad w_{jk} \geq 0 \quad \forall j, k \tag{2.8}$$

in order $b_j(\boldsymbol{o})$ to fulfill the properties of a PDF [Bishop 1998]. The component densities $f_{\mathcal{D}}(\boldsymbol{o}; \theta_{jk})$ are density functions of some known distributions $\mathcal{D}$ with parameters θ_{jk}. Most often $f_{\mathcal{D}}(\boldsymbol{o}; \theta)$ is the multivariate Gaussian density function defined as [Johnson and Wichern 1998]

$$f_{\mathcal{N}}(\boldsymbol{o}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}\sqrt{|\boldsymbol{\Sigma}|}} \exp\left[(\boldsymbol{o} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{o} - \boldsymbol{\mu})\right] \tag{2.9}$$

where θ consists of $\boldsymbol{\mu}$, the mean, and $\boldsymbol{\Sigma}$, the covariance matrix of the distribution. In that case, the density in Equation (2.7) is also known as Gaussian mixture model (GMM). Furthermore, the covariance matrices of the Gaussian distributions are often constrained diagonal. Diagonality of the covariance matrices reduces the total number of parameters in the model significantly, especially when the dimension of the feature vector is large. Even though the diagonality of the covariance matrix of a component density suggests that the elements of the modeled random vector are independent[5], the mixture of these densities $b_j(\boldsymbol{o})$ can model correlation characteristics as well [Reynolds et al. 2000]. In addition, different parameter tying schemes have been proposed for covariance matrix modeling in HMMs, e.g. semi-tied covariance matrices [Gales 1999].

Alternative types of observation emission densities than the usual GMMs have been proposed for ASR applications, e.g. Richter, power exponential and Laplacian distributions [Gales and Olsen 1999]. However, no significant advantages have been achieved by using these different observation densities. The parameters of all the state-dependent emission densities in a HMM are denoted with Θ. For the clarity of notation, we use a single symbol $\lambda = (\boldsymbol{\pi}, \mathbf{A}, \Theta)$ to denote the whole parameter set of a single HMM. In Chapter 4, both symbols λ and κ are used to denote this parameter set.

2.4.2 Speech Recognition Using HMMs

In order to solve the classification task defined in Equation (2.6), the likelihood $P(\mathbf{O} \mid \lambda_W)$ and prior probability $P(\lambda_W)$ need to be determined for each word model λ_W.

The usual case is that the acoustic units, that are modeled as single HMMs, represent phonemes. In this case, the word models λ_W must be constructed before recognition. This word model composition is depicted in Figure 2.4. First of all, the phonetic transcription of the current word is obtained from the lexicon. Then, the corresponding phoneme models are concatenated after each other to form a single word model. This procedure is applied to all the vocabulary items. In the following, the evaluation of $P(\mathbf{O} \mid \lambda_W)$ is determined

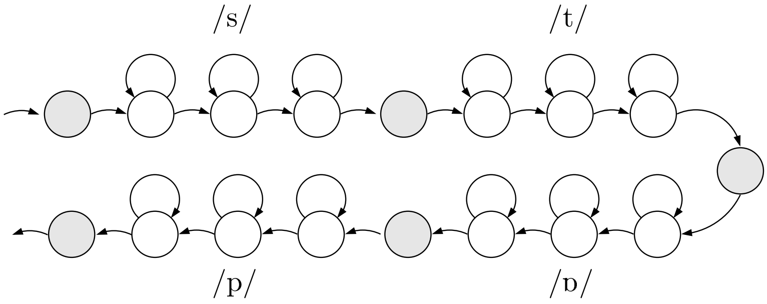

Figure 2.4: Concatenation of HMMs. The phonetic transcription of the english word "stop" written using IPA symbols is /stɒp/. The word consists of four phonemes, which are modeled using separate phoneme models. These monophone HMMs corresponding to the phonemes are concatenated using the non-emitting states in the beginning and at the end of the models. By defining the symbol "⊕" to indicate the model concatenation, the model for the word "stop" can be written as /s/⊕/t/⊕/ɒ/⊕/p/.

for only one HMM. The evaluation of $P(\mathbf{O} \mid \lambda_W)$ for concatenated HMMs can be found e.g. in [Jelinek 1998].

The likelihood of a feature vector sequence $\mathbf{O}$ given a HMM with parameters λ can be calculated as follows [Rabiner 1993]

$$P(\mathbf{O} \mid \lambda) = \sum_{\boldsymbol{q} \in \mathcal{Q}^{(T)}} P(\mathbf{O}, \boldsymbol{q} \mid \lambda) = \sum_{\boldsymbol{q} \in \mathcal{Q}^{(T)}} \pi_{q_1} \prod_{t=1}^{T-1} b_{q_t}(\boldsymbol{o}_t) a_{q_t q_{t+1}} \tag{2.10}$$

where $\mathcal{Q}^{(T)}$ is the set of all state sequences of length T. Usually, when the last observation vector is encountered, the state is restricted to be the last non-emitting state of the model, in which case

$$\mathcal{Q}^{(T)} = \{\boldsymbol{q} \in \mathbb{Z}^T : q_T = N\} \tag{2.11}$$

where N is the number of states in the model. The Equation (2.10) can be evaluated recursively using the forward or backward procedure explained in Algorithms 2.1 and 2.2, respectively [Rabiner 1993]. The forward and backward variables $\alpha_t(j)$ and $\beta_t(j)$ are defined as [Rabiner 1993]

$$\alpha_t(j) = P(\boldsymbol{o}_1, \boldsymbol{o}_2, \cdots, \boldsymbol{o}_t, q_t = j \mid \lambda) \tag{2.12}$$

$$\beta_t(j) = P(\boldsymbol{o}_{t+1}, \boldsymbol{o}_{t+2}, \cdots, \boldsymbol{o}_T \mid q_t = j, \lambda) \tag{2.13}$$

The constraint $q_T = N$ results in the termination rule in Algorithm 2.1. In the literature, the termination rule is determined without this constraint, i.e. $P(\mathbf{O} \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$ [Rabiner 1993]. Similarly, the initialization rule in Algorithm 2.2 can be given as $\beta_T(i) = 1$, $1 \leq i \leq N$ without the constraint $q_T = N$.

Often, instead of using the likelihood in Equation (2.10), the likelihood of the most likely

[5] In the case of Gaussian distributions, uncorrelatedness of the elements is equivalent with their independence [Johnson and Wichern 1998].

$$
\begin{array}{lrl}
\text{Initialization} & \alpha_1(i) = \pi_i b_i(\boldsymbol{o}_1) & 1 \le i \le N \\
\text{Recursion} & \alpha_t(j) = \left[\displaystyle\sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} \right] b_j(\boldsymbol{o}_t) & \begin{array}{r} 2 \le t \le T \\ 1 \le j \le N \end{array} \\
\text{Termination} & P(\mathbf{O} \mid \lambda) = \alpha_T(N) &
\end{array}
$$

Algorithm 2.1: The Forward procedure [Rabiner 1993].

$$
\begin{array}{lrl}
\text{Initialization} & \beta_T(i) = 0 & 1 \le i \le N-1 \\
 & \beta_T(N) = 1 & \\
\text{Recursion} & \beta_t(i) = \displaystyle\sum_{j=1}^{N} a_{ij} b_j(\boldsymbol{o}_{t+1}) \beta_{t+1}(j) & \begin{array}{r} T-1 \ge t \ge 1 \\ 1 \le i \le N \end{array} \\
\text{Termination} & P(\mathbf{O} \mid \lambda) = \displaystyle\sum_{j=1}^{N} \pi_j b_j(\boldsymbol{o}_1) \beta_1(j) &
\end{array}
$$

Algorithm 2.2: The Backward procedure [Rabiner 1993].

state sequence is employed in the decoding phase [Rabiner 1989]. It can be given as

$$
P^*(\mathbf{O} \mid \lambda) = P(\mathbf{O}, \boldsymbol{q}^* \mid \lambda) = \pi_{q_1^*} \prod_{t=1}^{T-1} b_{q_t^*}(\boldsymbol{o}_t) a_{q_t^* q_{t+1}^*} \tag{2.14}
$$

$$
\boldsymbol{q}^* = [q_1^*, \ldots, q_T^*]^T = \underset{\boldsymbol{q} \in \mathcal{Q}^{(T)}}{\arg\max}\, P(\mathbf{O}, \boldsymbol{q} \mid \lambda) \tag{2.15}
$$

where P^* is the likelihood of the most likely state sequence $\boldsymbol{q}^*$. This likelihood $P^*(\mathbf{O} \mid \lambda)$, as well as the most likely state sequence $\boldsymbol{q}^*$, can be obtained using the Viterbi decoding procedure described in Algorithm 2.3. The algorithm is presented for the case when the state sequence is constrained as given in Equation 2.11. The termination rules without this constraint are $P^* = \max_{1 \le i \le N} \delta_T(i)$ and $q_T^* = \arg\max_{1 \le i \le N} \delta_T(i)$. The reason of using of this kind of a sub-optimal score instead of likelihood in the decoding phase is the computational efficiency. This score, often referred to as Viterbi-score, is usually computed in logarithmic domain, which increases the dynamic range and prevents possible underflows. Furthermore, the search of the most likely state sequence is often implemented using the so called token passing scheme [Young et al. 2000; 1989]. In that case, when performing e.g. connected-word recognition with loop grammars, the tracking of the most likely state sequence in decoding is efficient [Young et al. 1989].

2.4.3 Training of HMMs

The structure and the type of the HMMs to be used in an application must be assigned by expert knowledge. The typical structure of a phoneme model HMM is depicted in Figure 2.3. The structure of a HMM is defined by constraining the probability of some transitions in the transition matrix $\mathbf{A}$ to zero. The other model parameters to be configured

$$
\begin{array}{lll}
\text{Initialization} & \delta_1(j) = \pi_j b_j(\boldsymbol{o}_1) & 1 \le j \le N \\
 & \psi_1(j) & \\
\text{Recursion} & \delta_t(j) = \max\limits_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(\boldsymbol{o}_t) & 2 \le t \le T \\
 & \psi_t(j) = \arg\max\limits_{1 \le i \le N} \delta_{t-1}(i) a_{ij} & 1 \le j \le N \\
\text{Termination} & P^* = \delta_T(N) & \\
 & q_T^* = N & \\
\text{Path backtracking} & q_t^* = \psi_{t+1}(q_{t+1}^*) & t = T-1, T-2, \ldots, 1
\end{array}
$$

Algorithm 2.3: The Viterbi decoding algorithm [Viterbi 1967]. Temporary variables $\psi_t(i)$ and $\delta_t(i)$ contain the information of the most likely state sequence up to the state i at time instant t. The index of the previous state is stored in $\psi_t(i)$ while $\delta_t(i)$ contains the accumulated likelihood.

include the type of mixture densities, the number of the mixture component densities in the state-dependent emission denisities and the number of states in one HMM. Furthermore, the acoustic units to be modeled with a single HMM must be fixed.

After defining the configuration of HMMs, the parameters should be estimated using some method. Typically, these parameters are estimated using a labeled speech corpus containing a huge amount of training utterances. Assuming that monophone models are used, i.e. every phoneme is modeled with a single HMM, a combined word model is created to correspond to the phonetic transcription of each train utterance. The model concatenation scheme used in word model composition is shown in Figure 2.4. The training procedure using the combined word models is also referred to as the embedded re-estimation of the parameters [Young et al. 2000].

Most often the parameters are estimated according to the maximum-likelihood criterion. The expectation-maximization (EM) algorithm provides an iterative procedure for estimating the maximum likelihood estimates (MLEs) of the parameters λ of a HMM. At each iteration of the EM algorithm, the value of the likelihood function increases, converging to a local MLE of λ [Dempster et al. 1977]. For the sake of clarity, this algorithm is presented in the following for a single observation sequence. The generalization of the algorithm to multiple observation sequences as well as detailed description of the derivation of the following re-estimation equations can be found e.g. in [Rabiner 1993].

At each iteration of the EM algorithm for HMMs, the Baum's auxiliary function given as [Baum 1972]

$$
Q(\lambda, \widehat{\lambda}) = \sum_{\boldsymbol{q} \in \mathcal{Q}} P(\boldsymbol{q} \mid \mathbf{O}, \lambda) \log P(\mathbf{O}, \boldsymbol{q} \mid \widehat{\lambda}) \tag{2.16}
$$

is maximized with respect to the new HMM parameter estimates $\widehat{\lambda}$. The detailed description of this maximization procedure can be found e.g. in [Rabiner 1993]. The maximization of Q results in an increased value of the likelihood function,

$$
\widehat{\lambda} = \arg\max_{\widetilde{\lambda}} Q(\lambda, \widetilde{\lambda}) \implies P(\mathbf{O} \mid \widehat{\lambda}) \ge P(\mathbf{O} \mid \lambda) \tag{2.17}
$$

The new estimates of the parameters $\widehat{\lambda}$ are given in the following Baum-Welch re-estimation formulae [Rabiner 1993]. The given formulae apply for HMMs with GMM emission densities.

$$\widehat{\pi}_i = \gamma_1(i) \tag{2.18}$$

$$\widehat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \sum_{l=1}^{M} \gamma_t(j,l)} \tag{2.19}$$

$$\widehat{w}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T} \sum_{l=1}^{M} \gamma_t(j,l)} \tag{2.20}$$

$$\widehat{\boldsymbol{\mu}}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot \boldsymbol{o}_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{2.21}$$

$$\widehat{\boldsymbol{\Sigma}}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot (\boldsymbol{o}_t - \boldsymbol{\mu}_{jk})(\boldsymbol{o}_t - \boldsymbol{\mu}_{jk})^T}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{2.22}$$

The symbols on the left hand side of the Equations (2.18)–(2.22) are the new estimates of the parameters λ explained in Section 2.4.1. The auxiliary variables used in Equations (2.18)–(2.22) are defined as $\xi_t(i,j) = P(q_t = i, q_{t+1} = j \mid \mathbf{O}, \lambda)$ and $\gamma_t(j,k) = P(q_t = j, \text{mixture component} = k \mid \mathbf{O}, \lambda)$. The latter is often referred as the *a posteriori* probability of the tth observation being emitted from the kth mixture of state j. They can be obtained as follows [Rabiner 1993]

$$\xi_t(i,j) = \frac{\alpha_t(i) a_{ij} b_j(\boldsymbol{o}_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(\boldsymbol{o}_{t+1}) \beta_{t+1}(j)} \tag{2.23}$$

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \right] \left[\frac{w_{jk} f_{\mathcal{N}}(\boldsymbol{o}_t; \boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk})}{b_j(\boldsymbol{o}_t)} \right] \tag{2.24}$$

where $f_{\mathcal{N}}(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the multivariate Gaussian density function. The forward and backward variables, $\alpha_t(j)$ and $\beta_t(j)$, respectively, used in the formulae are obtained from the procedures in Algortihms 2.1 and 2.2.

2.5 Speaker Adaptation

The estimation of the parameters of the statistical models used in a speech recognition system demands large amounts (hours) of speech material. Training a speaker independent (SI) speech recognition system is straightforward due to large annotated speech databases available. The SI recognition systems perform rather well in most cases, but the error rate is two to three times higher than with speaker dependent (SD) systems [Leggetter and Woodland 1994]. The SD recognition system provides better modeling of the speech characteristics for the target speaker. However, it is unfeasible to gather such a large amount of speech material for every speaker. Model adaptation techniques provide methods for adapting a SI recognition system for the target speaker using only small amount of acoustic adaptation material. A notable gain in recognition accuracy is achieved by the use of such techniques [Leggetter and Woodland 1994].

Often, speaker adaptation is performed using maximum *a posteriori* (MAP) or maximum likelihood linear regression (MLLR) methods, which both are model adaptation techniques.

The maximum *a posteriori* (MAP) technique provides a method for adapting the parameters of the HMMs with a relatively small amount of adaptation data per model [Gauvain and Lee 1994]. HMMs without adaptation data are left unchanged. Alternatively, the maximum likelihood linear regression (MLLR) can be employed for speaker adaptation [Leggetter and Woodland 1994]. This method is useful particularly when context-dependent models[6], are used in the speech recognition system. In the MLLR framework, all the HMMs are adapted, even such HMMs that do not have any adaptation data. This is achieved such that the mixture density components of the HMM states are allocated to so called transformation classes, which share the adaptation data of all the densities in such class. The transformation classes are formed using the acoustic similarities between the models. The MLLR technique is presented in brief in the following for HMMs with GMM emission densities.

2.5.1 Maximum Likelihood Linear Regression

The MLLR technique is based on an affine transformation of the mean vectors $\boldsymbol{\mu} \in \mathbb{R}^D$ for each mixture component. The transformation is defined as [Leggetter and Woodland 1994]

$$\widehat{\boldsymbol{\mu}} = \mathbf{W}\boldsymbol{\nu} = \mathbf{W}\begin{bmatrix}\omega \\ \boldsymbol{\mu}\end{bmatrix} \tag{2.25}$$

where $\widehat{\boldsymbol{\mu}}$ is the new (transformed) mean vector, $\mathbf{W} \in \mathbb{R}^{D\times(D+1)}$ is the transformation matrix and ω is the offset term for the regression.

The transformation matrix $\mathbf{W}$ is derived via similar optimization scheme as in the Baum-Welch training procedure described in Section 2.4.3. The optimization scheme results in that the transformation matrix $\mathbf{W}_\Gamma$ can be obtained for the mixture components in transformation class Γ by solving [Leggetter and Woodland 1994]

$$\sum_{(j,k)\in\Gamma}\sum_{t=1}^{T}\gamma_t(j,k)\boldsymbol{\Sigma}_{jk}^{-1}\boldsymbol{o}_t\boldsymbol{\nu}_{jk}^T = \sum_{(j,k)\in\Gamma}\sum_{t=1}^{T}\gamma_t(j,k)\boldsymbol{\Sigma}_{jk}^{-1}\mathbf{W}_\Gamma\boldsymbol{\nu}_{jk}\boldsymbol{\nu}_{jk}^T \tag{2.26}$$

where the *a posteriori* probabilities $\gamma_t(j,k)$ are described in Section 2.4.3. The outer sums in Equation (2.26) go through all the mixture density components i.e. corresponding (state, mixture) pairs. The matrix $\mathbf{W}_\Gamma$ can be solved from Equation (2.26) with the essential cost of D matrix inversions of dimension $D+1$, where D is the dimension of the feature vector [Leggetter and Woodland 1994]. The determination of the transformation classes Γ is discussed in Appendix A. These classes can be determined using the dissimilarity measures explained in Chapter 4, but the topic is beyond the scope of this thesis.

[6] The widely used context-dependent phone models in ASR are the so called triphone models. They have explicit left and right context phones. The number of unique triphone models is p^3, where p is the number of unique phonemes in the language. Typically p is around 40.

Chapter 3

Multilingual Speech Recognition

The research in the field of speech recognition has been rather intense on few major languages, namely on the American English [Adda-Decker 2001, Young et al. 1997]. Many potential applications, however, require support for multiple languages. These include e.g. voice dialing applications in mobile handsets, which are typically aimed for wide geographic areas covering numerous languages [Kiss 2001]. Even the support for minor languages can be considered a necessity in these kind of applications. The flight reservation systems are another area of applications which benefit from the support of multiple languages, non-native speakers and speakers with strong dialects. The benefits of a multilingual ASR system include also reduced development costs, as no different language dependent versions of the system need to be developed [Viikki et al. 2001].

The main topics in the field of multilingual speech recognition are the porting of existing recognition systems for new languages and the multilingual acoustic modeling. The latter concerns the development of a multilingual recognition system without separate acoustic models for each language. The porting covers development of a speech recognition system for new language using existing speech recognition systems and corpora. The issue of multilinguality in speech recognition is still fairly new. The first publications concerning multilinguality in speech technology are from the late 1980s. However, from the middle of the 1990s, this research are has started to gain increasing attention [Adda-Decker 2001, Andersen et al. 1994, Byrne et al. 2000, Fung et al. 1999, Imperl and Horvat 1999, Kiss 2001, Köhler 2001, Navrátil 2001, Uebler 2001, Van Compernolle 2001, Waibel et al. 2000, Young et al. 1997].

This chapter covers the issues of multilingual ASR discussed above. The Section 3.1 reviews the most basic resources, speech corpora needed for the development purposes of multilingual speech recognition. Next, in Section 3.2, the portability of ASR technology is discussed. The Section 3.3 covers the issues of multilingual acoustic modeling.

3.1 Multilingual Speech Corpora

The labeled speech corpora are the basic resources needed in the development of speech recognition systems. Considering multilingual speech recognition systems, extra requirements are set on the speech corpora. The corpora must include several language dependent speech corpora compatible to each other. This means that the acoustic conditions, such as background noise and microphone distortions, should be similar. Furthermore, the content of speech, i.e. the vocabulary and the type of speech should be similar. The training and test sets should also be similar to achieve comparable results across the languages.

As the research in multilingual ASR has gained momentum, a number of multilingual speech corpora have been introduced [Adda-Decker 2001]. These include SpeechDat, OGI, LDC CallHome[1] and GlobalPhone [Muthusamy et al. 1992, Schultz et al. 1997, Winski 1997].

3.2 Portability of Speech Technology

Since most of the technology used in modern ASR applications is developed for a single language we can ask what are the parts of the speech recognition systems[2] that can be considered language independent. Furthermore, can the same development methods explained in Section 2.4 be used for training a speech recognition system for any language provided that a suitable speech corpus exists? For many minor languages, such corpus does not exists, and the collection of such a large database is unfeasible. In such a case, is it possible to obtain a speech recognition system for a new language with no acoustic data, or very little acoustic data available? The following sections cover these issues.

3.2.1 Language-dependency of Speech Recognition Systems

The typical MFCC feature extraction unit explained in Section 2.3 can be considered relatively language-independent, because only low-level signal processing is performed at that level [Adda-Decker 2001, Kiss 2001]. The acoustic features commonly used in recognition of the English language can be used also in recognition of e.g. most of the other European languages. These acoustic features do not include information of pitch, which is vital when recognizing tonal languages, e.g. Mandarin Chinese. In such languages, the whole meaning of a phrase can change due to different pitch pattern [Lee 1997]. A modified front-end can be formed where the feature vector is augmented to comprise also information of pitch [Chang et al. 2000].

The acoustic models used in modern speech recognition systems are HMMs, as explained in Section 2.4.1. The training as well as the decoding algorithms of HMMs are general, and independent of the language [Adda-Decker 2001]. However, the topology of the HMMs and the decision of the acoustic units[3] to be modeled with a single HMM can demand language dependent research. Therefore, the framework of acoustic modeling is applicable in general without modifications, regardless of the language.

The lexicon and the language model are obviously the most language dependent parts of the system. The languages are very different in both written and spoken form. The segmentation into words, morphology and the used character set differs greatly between languages. The morphology and prosody reflect to the spoken form of the language [Waibel et al. 2000]. Most European languages seem to share many common features, but when comparing e.g. Asian languages to European languages, the differences are tremendous.

The porting of the existing speech recognition systems was researched in the European SQALE project [Young et al. 1997]. Four recognition systems designed originally for American English were compared in recognition of three European languages: British English, French and German. The systems were trained using a common predefined training data

[1] See Linguistic Data Consortium (LDC), University of Pennsylvania web page for details: `http://www.ldc.upenn.edu`.

[2] The structure of modern ASR system is depicted in Figure 2.1.

[3] The acoustic unit refers here to phoneme, allophone or syllable.

set and evaluated with corresponding test data set. The development time of the systems was rather limited, since the purpose of the project was to evaluate how easily the recognition systems can be modified for new languages. The results showed that the porting of the basic systems was quite straigthforward, although the last system refinements affecting performance were quite language dependent. It must be noted though, that the languages used in the SQALE project were quite similar in structure, and the common front-end was also suitable for the new languages.

3.2.2 Cross-language Transfer

The speech corpora containing hours of labeled speech samples are laborious and expensive to gather. Text material, however, can be obtained rather easily for most languages of interest [Adda-Decker 2001]. Usually, the material available covers such information as text-to-phoneme rules and lexicon, and possibly a language model. This introduces a question that how well the target language can be modeled by using a recognition system trained for one source language different to the target language. This procedure is referred to as cross-language transfer.

Schultz et.al. have addressed this problem and concluded that the choise of the source language is crucial for the resulting recognition accuracy after the cross-language transfer [Schultz and Waibel 2001]. Moreover, expert phonetic knowledge is needed in the transfer, since the phonemes of the target language need to be mapped to the phonemes of the source language. Considering the cross-language transfer, a better recognition accuracy can be obtained by training a recognition system with multiple source languages [Köhler 1998, Leppänen et al. 2001, Žgank et al. 2001]. This method, usually referred to as multilingual acoustic modeling, is discussed in Section 3.3. If the source languages used in the transfer are properly selected, the recognition accuracy of the unseen language[4] can be rather good as well [Viikki et al. 2001].

3.2.3 Language Adaptation

Language adaptation is a method placed between a full language dependent training procedure with a complete speech corpus and the cross-language transfer discussed above. A relatively small amount of labeled acoustic adaptation utterances is needed from the target language compared to a full speech corpus. This amount of data is insufficient for full training of a new speech recognition system, but can be used well for adapting a fully trained speech recognition system. The recognition accuracy has been observed to improve rapidly as the number of adaptation utterances increases. Even a hundred sentences is enough to gain about 25% reduction in word error rate from the results achieved with baseline systems [Fung et al. 1999, Harju et al. 2001, Köhler 1998, Leppänen et al. 2001, Schultz and Waibel 2000]. The baseline systems used as a starting point of language adaptation are either cross-language or multilingual systems. The language adaptation is typically performed using methods originally developed for speaker adaptation. These include MAP and MLLR adaptation techniques explained in Section 2.5.1.

Ordinary speaker adaptation of a multilingual system can be also viewed as language adaptation. The native language of the speaker is considered as one of the speaker-specific features when performing the speaker adaptation. The native language is, in fact, greater acoustic difference compared to variation between speakers. Therefore, the adaptation of

[4] The new language is referred to as “unseen”, when no training data is available for that language.

language-specific features is a big part of the speaker adaptation that is performed for multilingual ASR systems [Viikki et al. 2001].

3.3 Multilingual Acoustic Modeling

The concept of a phonetic typewriter issued in the 1950s was based on the idea of a machine capable of transcribing auditory speech signals [Gold and Morgan 2000]. The sound units in spoken languages, phonemes, were supposed to be distinguishable from spoken utterances, which could be then transferred e.g. to a written form. This showed to be inapplicable since the variation of the phonemes differs greatly according to context, speaking style, age, and the language used. Even a human listener cannot transcribe spoken utterances reliably if the linguistic content is unclear. This is also the situation when the listener is unfamiliar with the language.

All the spoken languages still do seem to share common acoustic features. The source of speech, i.e. the human speech production system, is the same regardless of the language used. This implies that the speech signals share common acoustic features, originating from the physical properties of the human speech production system. In all the spoken languages, the content of speech is determined by a stream of phonemes articulated after each other. Thus the use of left-to-right proceeding HMMs as acoustic models can be considered a language independent strategy for acoustic modeling in speech recognition applications. The multilingual acoustic modeling can be stated applicable based on these facts.

The Figure 3.1 shows how the sharing of acoustic models across languages effects the structure of a speech recognition system. Instead of using a separate set of acoustic models for each language, a common set of models is utilized in multilingual acoustic modeling. This results in the following benefits. The development costs are reduced as there is no need developing separate language dependent systems [Viikki et al. 2001]. In addition, the multilingual systems can cope with non-native speakers, accents, dialects and multilingual vocabulary items [Viikki et al. 2001].

Even when constructing a language dependent speech recognition system, the model parameters can be shared[5] [Woodland and Young 1993]. The reasons for this are twofold. Firstly, such units that are unseen in the training phase can share the parameters of other units. Secondly, the number of free parameters in a speech recognition system can be reduced without notable drop in recognition accuracy [Woodland and Young 1993]. In fact, when the number of free parameters is lower, the statistical estimation of the parameters is more robust, which can improve the performance and generality of the speech recognition system [Woodland and Young 1993]. The research concerning multilingual acoustic modeling indicated that much can be gained by modeling different languages with common acoustic models. The following sections outline the methods for developing multilingual speech recognition systems. The Sections 3.3.1 and 3.3.2 describe the knowledge-based and computational approaches for the definition of a set of multilingual acoustic models. The Section 3.3.3 outlines the functionality of the language identification (LID) unit needed in a multilingual ASR system.

[5] This is performed most often with ASR systems having context-dependent acoustic units, e.g. triphone HMMs.

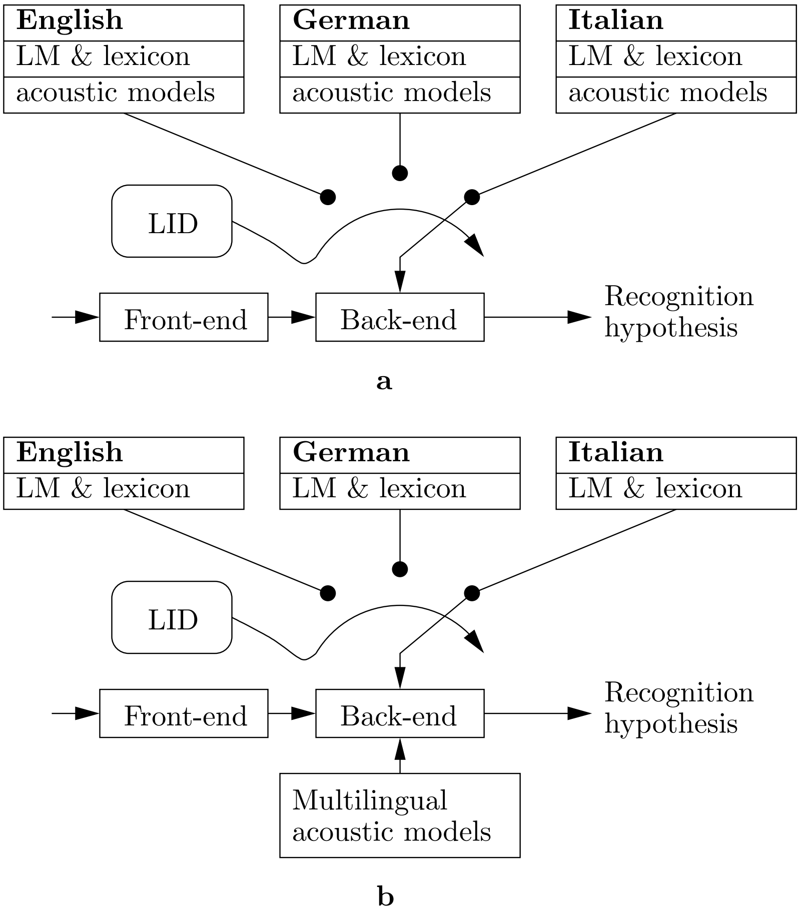

Figure 3.1: (a) Several monolingual acoustic models vs. (b) a set of multilingual models in speech recognition system. The language identification (LID) block controls the use of different recognition systems. The language model (LM) and the lexicon are always separate, as they are language dependent.

3.3.1 Knowledge-Based Methods

The International Phonetic Association (IPA) has defined a phonetic alphabet for describing the sounds used in human speech [Ladefoged et al. 1999]. This alphabet is designed to provide consistent means to characterize the pronunciation in spoken languages globally. The IPA phonetic transcription has been applied e.g. in many printed dictionaries. The phone inventory defined by IPA is still found to be subjective, and not all the phoneticians agree with all the definitions [Köhler 1999].

The most simple way to define a multilingual phone set is to gather all the distinct IPA-alphabet symbols from the source languages. In this method, referred to as IPA-MAP, each distinct phone corresponding to an IPA symbol is modeled with one acoustic model. This kind of definition of phone models is the most straightforward knowledge-based method for the task [Köhler 2001]. Further simplifications may be introduced to decrease the number of phone models in the recognition system, e.g. by substituting the double consonant and double vowel phonemes by two distinct single phones [Vihola et al. 2002].

The IPA-MAP method provides consistent definition of multilingual phone set. Furthermore, if a set of multilingual acoustic models is created according to the IPA alphabet[6], the recognition system should be potentially language independent. This means that an unseen target language can be recognized if the phone inventory of the source languages is sufficient, i.e. it contains all the phonemes in the target language. The IPA chart is defined for the purposes of the phonetic representation of the languages, and is based on articulatory features of the sounds. It may not describe the acoustic features as accurately as is needed for the purposes of speech recognition, e.g. the representation does not cover the allophone variation of one phoneme.

3.3.2 Computational Methods

When expert knowledge is not available or, e.g. the number of multilingual phone models is constrained to be very low, the IPA-MAP method may not be useful. In this case, an automatic, i.e. computational, method may provide an alternative approach for defining a set of multilingual phone models [Köhler 2001]. Using the same number of phone models as in the IPA-MAP method, the recognition accuracy of the resulting multilingual ASR system has been observed to be slightly better with the computational method [Köhler 2001]. Conversely, a recognition accuracy comparable to the IPA-MAP recognition system can be achieved with less phone models when using the computational method [Harju et al. 2001].

The computational methods used in defining the multilingual phone model set are based on some measure of dissimilarity of two language dependent phoneme models, namely HMMs. These dissimilarity measures are reviewed and described in detail in Chapter 4. Usually, this kind of a measure is employed, and the models are collected to certain number of clusters, and each cluster is modeled with a common multilingual phone model. The clusters are typically obtained using a bottom-up, i.e. agglomerative clustering algorithm [Harju et al. 2001, Köhler 2001]. This algorithm is described briefly in Algorithm 3.1.

3.3.3 Language Identification

A multilingual speech recognition system is fully operational after a language identification (LID) block is implemented [Kiss 2001]. The decision in LID block is made according to some knowledge about the spoken language. The knowledge of the language can be explicit, e.g. defined by the user, or automatically identified. The function of the LID block in the recognition system is depicted in Figure 3.1. In the case of multilingual recognition system constructed from a set of monolingual recognition systems, the LID block switches between the monolingual recognition systems as shown in Figure 3.1 (a). After the language selection, the recognition system performs the recognition using the chosen monolingual system. In the case of a multilingual recognition system with common multilingual acoustic models only the language model and the lexicon need to be selected. The same acoustic models are used for all the languages. This is shown in Figure 3.1 (b).

Automatic language identification has many real-world applications including telephony services, such as hotel reservation and emergency lines [Muthusamy et al. 1994]. The issue of language identification has been researched for decades, and it seems that the peak in the number of published reports and articles is just before middle of the 1990s. The systems implemented for language identification purposes are based on numerous methods

[6] Usually, the alphabet used in computer environments is Speech Assessment Methods Phonetic Alphabet, SAMPA. It is a mapping of IPA alphabet into ASCII codes [SAM].

$$
\begin{array}{lrll}
\text{Initialization} & \mathcal{I} &= \{1,\ldots,N\} & \\
 & \mathcal{P} &= \{\lambda_i : i \in \mathcal{I}\} & \\
 & G_i^1 &= \{\lambda_i\} & \forall i \in \mathcal{I} \\
n\text{:th iteration} & (i,j) &= \displaystyle\arg\min_{i,j\in\mathcal{I},\, i\neq j} \left[\max_{\lambda_k\in G_i^n,\, \lambda_l \in G_j^n} d_{kl} \right] & \\
 & G_i^{n+1} &= G_i^n \cup G_j^n & \\
 & \mathcal{I} &= \mathcal{I} \backslash \{j\} &
\end{array}
$$

Algorithm 3.1: Agglomerative clustering algorithm that has been used in phone model clustering [Theodoridis and Koutroumbas 1999]. The initial clusters G_i^1 consist of single phone models λ_i of the initial model set $\mathcal{P}$. The number of clusters is decreased by one at each iteration, as the two closest clusters are combined in symbol level. The closeness of the clusters is defined by the maximum dissimilarity d_{kl} between two phone models λ_k and λ_l in the clusters G_i^n and G_j^n, respectively. The algorithm is iterated until the desired number of clusters is achieved, i.e. the index set $\mathcal{I}$ has the desired number of elements.

ranging from expert systems to statistical classifiers [Muthusamy et al. 1994]. The field is wide also considering the number and nature of the acoustic features used in classification of the languages [Muthusamy et al. 1994].

Chapter 4

Dissimilarity Measures for Hidden Markov Models

In Chapter 3, the concept of multilingual acoustic modeling was discussed. The Section 3.3.2 summarized the computational method used in definition of a set of multilingual phone models. The explained method was based on a dissimilarity measurement of the LD phoneme models, i.e. HMMs. This dissimilarity has been evaluated in the previous research projects using one of the two methods: the Kullback-Leibler (KL) divergence estimate, or an estimate based on the confusion matrix [Andersen et al. 1994, Harju et al. 2001, Köhler 2001]. Both of the estimates have been obtained using some speech data set. However, these methods have mainly two drawbacks: Firstly, the estimation procedure is computationally expensive. Secondly, as the measures have been obtained from statistics computed using some speech data set[1], these measures have a considerable variation to the case when the measures are evaluated from another data set.

This chapter reviews the dissimilarity measures for HMMs. In addition to the above mentioned computational phoneme model clustering, the dissimilarity measures can be utilized in speech recognition e.g. in model selection and clustering [Juang and Rabiner 1985]. Furthermore, the measures could be applied in the MLLR adaptation framework, as described in Appendix A. A detailed description of such dissimilarity measures that are suitable for the purposes of phoneme model clustering is given in this chapter. Although these measures are presented only for the monophone HMMs, most of the techinques can be generalized for cases in which different acoustic modeling units, e.g. allophones or words, are used. In addition, it should be noted that all the measures described in this chapter are considered as dissimilarity measures. This is due to simpler representation, interpretation and comparison of the measures. Intuitively, a dissimilarity measure can be considered as a "distance" metric. All the dissimilarity measures described in this chapter, however, do not fulfill the properties of a proper metric[2]. Therefore, the presented measures are generally referred to as dissimilarity measures.

The natural criterion, that describes the dissimilarity of statistical models used in pattern recognition, is the classification characteristics, i.e. the nature of the errors made in classification task. This can be measured e.g. using a confusion matrix such as shown in Table 4.1. An interpretation of the dissimilarity in such a case is as follows. The more

[1] This means that the speech data is considered as independent random observations of the HMMs.

[2] In topology, a metric is a function that describes proximity of objects. It is defined formally as a function $d : X \times X \to [0, \infty)$, having the following properties:

1. $d(x, y) = 0$ if and only if $x = y$
2. $d(x, y) = d(y, x)$
3. $d(x, z) \leq d(x, y) + d(y, z)$

where x, y and z are elements of the topological space X [Gariepy and Ziemer 1994].

Table 4.1: An example of a confusion matrix. Observations of class 'a' are classified correctly every time, i.e. there are no confusions of 'a' to 'b' or 'c'. However the observations of class 'c' are confused to class 'a' ten times out of 100. The values in the table are percentages.

		Classified as		
		'a'	'b'	'c'
Source class	'a'	**100**	0	0
	'b'	0	**67**	33
	'c'	10	0	**90**

frequently the models are misclassified with each other, the closer they are, i.e. the value of the dissimilarity measure is small. Conversely, if the models are very discriminating, i.e. misclassifications occur rarely, the dissimilarity value is large. This dissimilarity can be obtained using either one of the alternative methods.

This chapter has briefly the following content. First, the confusion matrix approach is discussed in Section 4.1. The estimation of an confusion matrix is described for phoneme model HMMs. In addition, two methods are proposed to get improved, i.e. more accurate, confusion matrix estimates. The Section 4.2 describes the Kullback-Leibler divergence and reviews the previously proposed methods for evaluating this measure for HMMs. In addition, two modified measures are proposed that are obtained very similarly to the two proposed confusion matrix estimates. Finally, Section 4.2.3 discusses such measures that are based on Kullback-Leibler divergence, but employ simplifying assumptions, and can be presented in a closed form. One of such previously presented measure is generalized for HMMs with arbitrary emission densities[3]. After that, a modification of the latter is presented. These dissimilarity measures, that can be given in closed form with respect to the HMM parameters, have very low computational cost compared to the other measures, which are estimated using speech data.

4.1 Measures Based on Confusion Matrix

In pattern recognition, the classification errors are very often of greatest interest. The classification errors are usually presented in the form of a confusion matrix, such as shown in Table 4.1. The matrix is obtained usually using a labeled data set that contains sufficient amount of samples from each class. The columns of the matrix correspond to the results of classification. The rows correspond to the true pattern classes of the observations. Thus, the element c_{ij} of a confusion matrix $\mathbf{C}$, is the number of classifications of observations of class i as class j. The diagonal elements thus show the number of correctly classified observations for each case, and the off-diagonal elements show the number of misclassifications, i.e. substitution errors. The confusion matrices are usually represented such that the rows of the matrix are normalized by the total number of tokens evaluated. This way the value in the estimated matrix shows relative number of confusions, i.e. the confusion

[3] If the divergences of the state-dependent observation densities can be given in a closed form, this measure can also be given in a closed form.

frequency, which can be given as

$$\widehat{c}_{ij} = \frac{\operatorname{card}\{k : \mathbf{O}_k^{\kappa_i} \text{ is classified as } \kappa_j\}}{\operatorname{card}\{k : \mathbf{O}_k^{\kappa_i}\}} \tag{4.1}$$

where card denotes the number of elements in the set. Once a confusion matrix estimate $\widehat{\mathbf{C}}$ is obtained for the set of models, it contains the estimates of how likely it is that a given model is classified as other model. The confusion frequency between two models describes the dissimilarity of the corresponding models in certain manner. Obviously, the more frequently the phone models are confused, the more similar they must be. However, the confusion matrix is not applicable as a dissimilarity measure as such. The Section 4.1.1 outlines the practical issues that need to be considered when estimating a confusion matrix for ASR classifier with phoneme model HMMs. Next, the Section 4.1.2 describes how a meaningful dissimilarity measure can be obtained based on a confusion matrix estimate.

4.1.1 Estimation of a Confusion Matrix for Phoneme Model HMMs

The estimation of the confusion matrix is not straightforward in the case of phoneme HMMs. This is due to two reasons. Firstly, the most common case is that isolated phoneme observations are not available, but only the observations of sentences or words. Secondly, the phonemes are coarticulated to each other in natural speech and each transition from one phoneme to another is smooth. This means that strict phoneme boundaries cannot be set. In addition, many speech corpora do not contain any phoneme boundary information.

If a speech database with phoneme boundary information is available, a confusion matrix of the phoneme models is rather straightforward to obtain. A predefined number of tokens $\mathbf{O}_1^\lambda, \ldots, \mathbf{O}_N^\lambda$ corresponding to each phoneme model λ are extracted from the speech corpus. After that, these tokens are classified individually and the results are gathered into a confusion matrix as determined in Equation (4.1). On the other hand, when the speech corpus does not contain the phoneme boundary information, automatic phoneme segmentation can be obtained using a well-trained speech recognition system. This automatic segmentation is based on the Viterbi decoding described in Algorithm 2.3. The algorithm is forced to obtain the most likely state sequence for the known phoneme sequence. The phoneme boundary information of each sentence is set according to the obtained most likely state sequence.

The classification of the tokens $\mathbf{O}_i^\lambda$ is performed such that a token corresponding the model λ is classified as

$$\kappa^* = \arg\max_{\kappa \in \mathcal{P}} \mathrm{SC}(\mathbf{O}_i^\lambda \mid \kappa) \tag{4.2}$$

where $\mathcal{P}$ is the set of all the phoneme models and $\mathrm{SC}(\mathbf{O}_i^\lambda \mid \kappa)$ is the likelihood score function, e.g. the forced Viterbi-score. A confusion matrix is formed based on these classification results of the tokens as explained in Section 4.1. In the following, we define three score functions, $\mathrm{SC}_S(\mathbf{O} \mid \kappa)$, $\mathrm{SC}_{C1}(\mathbf{O} \mid \kappa)$ and $\mathrm{SC}_{C2}(\mathbf{O} \mid \kappa)$, to be used in Equation 4.2. The confusion matrix estimates obtained using these score functions are referred to as $\widehat{\mathbf{C}}_S$, $\widehat{\mathbf{C}}_{C1}$ and $\widehat{\mathbf{C}}_{C2}$, respectively.

The first score function is the logarithmic likelihood of the most likely state sequence

$$\mathrm{SC}_S(\mathbf{O}^\lambda \mid \kappa) = \log P^*(\mathbf{O}^\lambda \mid \kappa) \tag{4.3}$$

where the likelihood P^* is obtained using the Viterbi decoding described in Algorithm 2.3. This score is typically used as the classification criterion in the speech recognition phase.

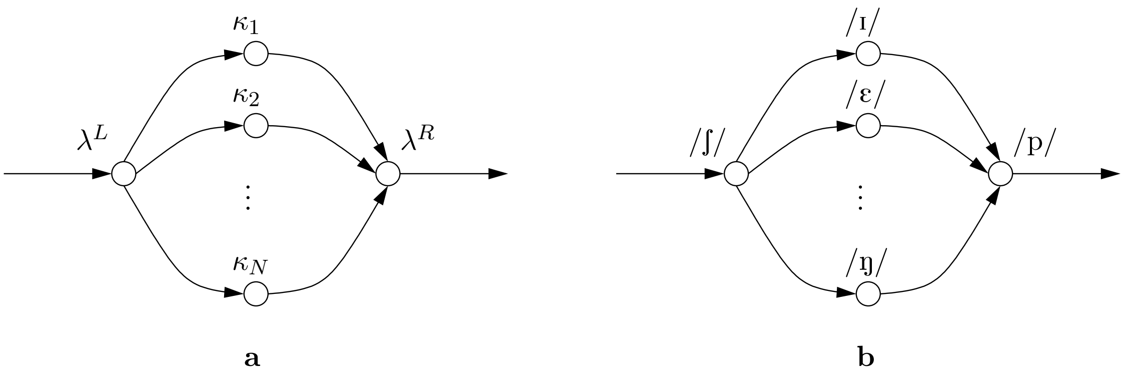

Figure 4.1: Evaluation of confusions using one left and right context model. The Figure (a) shows the symbols used in the Equations (4.4) and (4.5). For example, in Figure (b), the utterend English word /ʃɪp/, "ship", is used in the evaluation of confusions of the phoneme /ɪ/. Any phoneme model can appear in place of /ɪ/, and the model that gives the best overall likelihood score is chosen.

The effects of coarticulation and ambiguity of the phoneme boundaries can be one source of errors in estimation of the confusion frequencies. A method proposed to overcome this problem is based on the use of context phonemes in gathering of the tokens. This means that instead of gathering tokens $\mathbf{O}_i^{\lambda}$ corresponding the model λ, the tokens including the natural left and right context models $\mathbf{O}_i^{\lambda_i^L \oplus \lambda \oplus \lambda_i^R}$ are gathered. After that, the recognition of these tokens is performed by choosing the target center model as in Equation (4.2). The score function used in classification becomes

$$\mathrm{SC}_{C1}(\mathbf{O}^{\lambda^L \oplus \lambda \oplus \lambda^R} \mid \kappa) = \log P^*(\mathbf{O}^{\lambda^L \oplus \lambda \oplus \lambda^R} \mid \lambda^L \oplus \kappa \oplus \lambda^R) \tag{4.4}$$

where λ^L and λ^R are the left and right context models corresponding the current token of λ, respectively. The operator "$\oplus$" represents concatenation of models, as explained in Figure 2.4. This model substitution procedure is depicted in Figure 4.1.

Alternatively to the method based on Equation (4.4), one can eliminate the effect of the context models on the likelihood P^*. This means that only the contribution of the target model is included in the classification. This contribution is straightforward to evaluate, as the likelihood P^* is evaluated using the observation vectors as shown in Figure 4.2. The auxiliary variables $\delta_t(j)$ described in Algorithm 2.3 contain the desired information. The cumulative likelihood of the most likely state sequence of $\mathbf{O}^{\lambda^L \oplus \lambda \oplus \lambda^R}$ given the model $\lambda^L \oplus \kappa \oplus \lambda^R$ is denoted as $\delta_t^*(\kappa)$. The contribution of the target model can be written as

$$\mathrm{SC}_{C2}(\mathbf{O}^{\lambda^L \oplus \lambda \oplus \lambda^R} \mid \kappa) = \frac{\log \delta_m^*(\kappa) - \log \delta_{l-1}^*(\kappa)}{T_\kappa} \tag{4.5}$$

where l and m are the time indices of the left and right boundaries of κ, respectively. The denominator T_κ is the number of feature vectors of $\mathbf{O}^{\lambda^L \oplus \kappa \oplus \lambda_R}$ that lie in states of κ at the most likely state sequence of the combined model $\lambda_j^L \oplus \kappa \oplus \lambda_j^R$. The normalization by T_κ results in the mean log-likelihood per observation sample.

When the two latter score functions, $\mathrm{SC}_{C1}(\mathbf{O} \mid \cdot)$ and $\mathrm{SC}_{C2}(\mathbf{O} \mid \cdot)$, are used for estimation of the confusion frequencies, an extra paradigm arises. Due to the drift of the segmentation, the concept of confusion becomes fuzzy. The similarity of the segmentations can be included

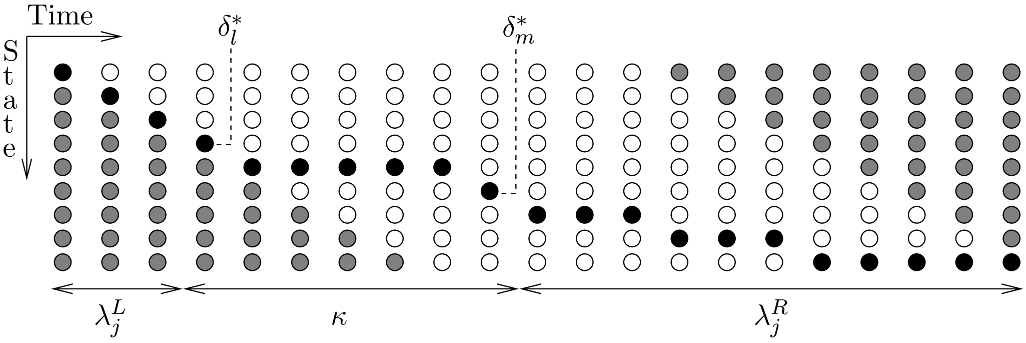

Figure 4.2: Contribution of the center model to the Viterbi-score of the HMM triplet. The most likely state sequence $\boldsymbol{q}^*$ is shown as black circles. The shaded circles represent impossible state-time combinations. Note also that each HMM has left-to-right topology, and three states.

to the confusion estimation framework. Two binary state segmentation vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, exemplified in Figure 4.3, are introduced. They are formed such that an element of the vector is one if the corresponding observation vector is assigned to the center model, and zero otherwise. Then, the similarity ratio between these two vectors is computed according to the Tanimoto measure, and a fuzzy confusion value is obtained. The Tanimoto similarity measure is defined for vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ as [Theodoridis and Koutroumbas 1999]

$$t(\boldsymbol{x},\boldsymbol{y}) = \frac{\boldsymbol{x}^T\boldsymbol{y}}{\boldsymbol{x}^T\boldsymbol{x} + \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{x}^T\boldsymbol{y}} \tag{4.6}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are such binary vectors as in Figure 4.3. This ratio of two binary vectors $t(\boldsymbol{x},\boldsymbol{y})$ can be interpreted as the ratio of the number of common elements having value of one in $\boldsymbol{x}$ and $\boldsymbol{y}$, and the number of elements that have the value one in either $\boldsymbol{x}$ or $\boldsymbol{y}$. Therefore, the effect of common zeros in $\boldsymbol{x}$ and $\boldsymbol{y}$ is discarded in the Tanimoto measure. In the confusion matrix estimation, which is determined in Equation (4.1), the number of confusions between the models are gathered to form the confusion matrix estimate. Similarly, when using the framework described above, the fuzzy confusion values are summed up to obtain the elements of the confusion matrix

$$\widehat{c}_{ij} = \frac{\displaystyle\sum_{\{k:\mathbf{O}_k^{\lambda_i} \text{ is classified as } \lambda_j\}} t(\boldsymbol{x}_k,\boldsymbol{y}_k)}{\operatorname{card}\{k : \mathbf{O}_k^{\lambda_i}\}} \tag{4.7}$$

where $\boldsymbol{x}_k$ and $\boldsymbol{y}_k$ are state segmentation vectors, such as in Figure 4.3, for observation $\mathbf{O}_k^{\lambda_i}$. It is easy to see that for arbitrary vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ the measure takes values $0 \leq t(\boldsymbol{x},\boldsymbol{y}) \leq 1$. Furthermore, if the alignments of the two target models do not overlap, $t(\boldsymbol{x},\boldsymbol{y}) = 0$, and no confusion is observed. The confusion matrix estimates corresponding to the confusion matrix estimates $\widehat{\mathbf{C}}_{C1}$ and $\widehat{\mathbf{C}}_{C2}$, but using the fuzzy confusion values are denoted as $\widehat{\mathbf{C}}_{C1}^t$ and $\widehat{\mathbf{C}}_{C2}^t$, respectively.

4.1.2 Conversion of a Confusion Matrix into a Dissimilarity Matrix

When deriving a dissimilarity matrix from a confusion matrix, the following issues need to be considered. The confusion matrix is not symmetric, but the dissimilarity matrix should

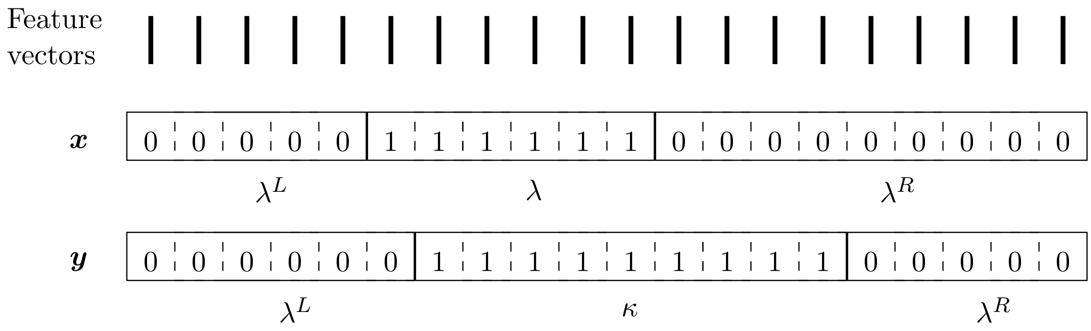

Figure 4.3: Similarity of the segmentations of a token. The token $\mathbf{O}^{\lambda^L \oplus \lambda \oplus \lambda^R}$ is classified as $\lambda^L \oplus \kappa \oplus \lambda^R$. The value of the Tanimoto measure of the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is 0.5.

be symmetric. In addition, the confusion frequencies describe the similarity of the models, not dissimilarity. It is obvious that the same information is contained in similarity and dissimilarity matrices, therefore performing the conversion of similarity into dissimilarity rather straightforward.

One conversion method of a confusion matrix into a symmetric similarity matrix is the Houtgast algorithm [Andersen et al. 1994, Imperl and Horvat 1999, Žgank et al. 2001]. It is given as

$$s_{ij} = \sum_{k=1}^{N} \min\left[c_{ik}, c_{jk}\right] = \frac{1}{2} \sum_{k=1}^{N} \left[c_{ik} + c_{jk} - |c_{ik} - c_{jk}|\right] \tag{4.8}$$

where c_{ij} are the elements of the confusion matrix $\mathbf{C}$ and s_{ij} is the similarity between the models i and j, i.e. an element of the similarity matrix $\mathbf{S}$. The conversion in Equation (4.8) essentially measures the simultaneous confusability of the both models i and j to all models k. If the both models are very often confused to a particular model, the minimum function gives a large value. This means that the sum over all the models, i.e. the value s_{ij}, is large and the models are considered similar. On the other hand, if one of the models is often confused to a model, and the other is not confused to that particular model, the output of the minimum function is small. In such a case the sum over all the models is small, and the models are considered dissimilar.

The Houtgast algorithm can be viewed almost as a special case of fuzzy similarity[4], a concept used in soft computing [Turunen 2001]. The only differences between the Houtgast and this fuzzy similarity relation are that, the similarity value is the mean of the summed elements, and the diagonal values of the similarity matrix equal to one, i.e.

$$s'_{ij} = \begin{cases} \frac{1}{N} \sum_{k=1}^{N} \min\left[c_{ik}, c_{jk}\right], & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases} \tag{4.9}$$

where s'_{ij} is the fuzzy similarity corresponding the Houtgast similarity s_{ij}. Therefore, the elements of the similarity matrix $\mathbf{S}'$ satisfy $0 \leq s'_{ij} \leq 1$, and the maximum similarity, i.e. value one, is achieved when the models are the same. These modifications do not, however, have any effect on the clustering framework, in which the dissimilarity measures are applied in this thesis. This is because of the following: Only the off-diagonal values

[4] This is the case when the chosen T-norm is the minimum function, i.e. the Gödel algebra is employed [Turunen 2001].

are employed in the clustering procedure. In other words, only the HMMs having different model indices are compared against each other. In addition, the similarity values are compared only against each other, meaning that the scaling does not have effect on the clustering procedure.

The conversion from a similarity matrix into a dissimilarity matrix can be performed using basically any monotonically decreasing function. The choice of such function depends on the nature of the application in which the dissimilarity matrix is to be used. The clustering can be performed using agglomerative clustering algorithm with complete linkage criterion, explained in Algorithm 3.1. As this procedure uses maximum dissimilarity in the cluster merging criterion, it is obvious that the clustering does not depend on the choise of the function used in the similarity-to-dissimilarity conversion. For example, the following simple function can be applied for this coversion

$$d_{ij} = 1 - s'_{ij} \tag{4.10}$$

where d_{ij} is the element of the dissimilarity matrix $\mathbf{D}$ corresponding the fuzzy similarity matrix $\mathbf{S}'$.

4.2 Measures Based on Kullback-Leibler Divergence

The dissimilarity measures described in Section 4.1 were based on the classification errors made by the recognition system. Alternatively, the dissimilarity of stochastic models can be measured based on the distributions of the models. A well-known dissimilarity measure between two probability distributions is the Kullback-Leibler divergence. It characterizes the discriminating properties of two probabilistic models λ and κ, and is defined as [Kullback 1968]

$$J(\lambda, \kappa) = I(\lambda : \kappa) + I(\kappa : \lambda) \tag{4.11}$$

where

$$I(\lambda : \kappa) = E\left\{\log \frac{f(\mathbf{O}^\lambda; \lambda)}{f(\mathbf{O}^\lambda; \kappa)}\right\} \tag{4.12}$$

is the directed divergence from λ to κ. In Equation (4.12), the expectation E is taken with respect to the random variable $\mathbf{O}^\lambda$ corresponding the distribution of the model λ, and the probability density functions of the corresponding models are denoted by $f(\mathbf{O}^\lambda; \cdot)$.

The Kullback-Leibler (KL) divergence measure cannot usually be presented for HMMs in a closed form. Therefore, some approximations of this measure have been introduced [Falkhausen et al. 1995, Juang and Rabiner 1985, Köhler 2001]. These approximations are based on Monte Carlo (MC) methods or simplifying approximations that lead to a closed form solution. In practice, the drawbacks of MC techniques are the extensive computational cost and slow convergence properties. On the other hand, the closed form approximations presented are limited to very specific class of HMMs, e.g. HMMs with discrete observation densities [Falkhausen et al. 1995]. The above mentioned methods are reviewed in Sections 4.2.1–4.2.3. The previously presented closed-form solution is extended to continuous distributions in Section 4.2.3. In addition, a modified approximation is presented in Section 4.2.3, that is supposed to be more accurate.

4.2.1 Monte-Carlo Techniques

Juang et al. studied the KL-divergence between HMMs using MC simulations. The models were assumed ergodic, and the measure was defined as the mean divergence per observation sample [Juang and Rabiner 1985]. The measure was given as

$$\widehat{I}_{MC}(\lambda : \kappa) = \frac{1}{T} \log \frac{P(\mathbf{O}^{\lambda} \mid \lambda)}{P(\mathbf{O}^{\lambda} \mid \kappa)} \tag{4.13}$$

where $\mathbf{O}^{\lambda}$ is an observation sequence generated by model λ. In addition, the length of the sequence is denoted with T, and the likelihoods given the models λ and κ are given as $P(\mathbf{O}^{\lambda} \mid \lambda)$ and $P(\mathbf{O}^{\lambda} \mid \kappa)$, respectively. The method is valid for HMMs with arbitrary observation probability distributions [Juang and Rabiner 1985].

Some approximations of the measure in Equation (4.13) were proposed and compared in [Falkhausen et al. 1995]. In that article, the assumed ergodicity property of the originally left-to-right models was achieved by substituting the transitions to non-emitting state with transitions to the first emitting state. The resulting measure, denoted by $\widehat{I}^{*}_{MC}$, was obtained by substituting the likelihoods $P(\mathbf{O} \mid \cdot)$ in Equation (4.13) with likelihoods of the most likely state sequences $P^{*}(\mathbf{O} \mid \cdot) = \max_{\boldsymbol{q} \in \mathcal{Q}} P(\mathbf{O}, \boldsymbol{q} \mid \cdot)^{5}$. Only minor differences were observed when comparing the behavior of $\widehat{I}_{MC}$ and $\widehat{I}^{*}_{MC}$ [Falkhausen et al. 1995].

4.2.2 Monte-Carlo Estimates based on Speech Data

A variant of the measure given in Equation (4.13) was proposed in [Köhler 2001]. Tokens extracted from speech corpus were used in the evaluation of the measure instead of a generated random sequence. Furthermore, the models were not ergodic as in Equation (4.13), but left-to-right phone models. Estimate of the divergence was defined as a sample mean of the log-likelihood differences over the tokens

$$\widehat{I}(\lambda : \kappa) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{T_i} \log \frac{P(\mathbf{O}_i^{\lambda} \mid \lambda)}{P(\mathbf{O}_i^{\lambda} \mid \kappa)} \tag{4.14}$$

where $\mathbf{O}_i^{\lambda}$ is the i:th token of length T_i corresponding the model λ. Experimental results showed that the obtained estimates of the divergence measure were applicable in phoneme model clustering [Köhler 2001]. As Falkhausen et. al. proposed, the likelihood of the most likely state sequence $P^{*}(\mathbf{O} \mid \cdot)$ can be used instead of $P(\mathbf{O} \mid \cdot)$ in Equation (4.14), which results in the estimate

$$\widehat{I}_{S}(\lambda : \kappa) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{T_i} \log \frac{P^{*}(\mathbf{O}_i^{\lambda} \mid \lambda)}{P^{*}(\mathbf{O}_i^{\lambda} \mid \kappa)} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{LLR}_{S}(\mathbf{O}_i^{\lambda} \mid \lambda, \kappa) \tag{4.15}$$

where the function $\mathrm{LLR}_{S}(\mathbf{O} \mid \lambda, \kappa)$ denotes the logarithmic likelihood ratio of $\mathbf{O}$ given λ and κ, respectively. The logarithmic likelihoods in Equation (4.15) are in fact the very same values that were used as likelihood scores in evaluation of the confusion matrix $\widehat{\mathbf{C}}_{S}$. Therefore, the function can be written as

$$\mathrm{LLR}_{S}(\mathbf{O}_i^{\lambda} \mid \lambda, \kappa) = \frac{1}{T_i} \left[\mathrm{SC}_{S}(\mathbf{O}_i^{\lambda} \mid \lambda) - \mathrm{SC}_{S}(\mathbf{O}_i^{\lambda} \mid \kappa) \right] \tag{4.16}$$

[5] This can be obtained using the Viterbi-decoding described in Algorithm 2.3.

where $\mathrm{SC}_S(\mathbf{O} \mid \kappa)$ are given in Equation (4.3). This measure has been applied for phoneme HMM clustering e.g. in [Harju et al. 2001].

In Section 4.1.1, the classification criterion used in estimation of $\widehat{\mathbf{C}}_S$ was modified in order to obtain the two new confusion matrix estimates $\widehat{\mathbf{C}}_{C1}$ and $\widehat{\mathbf{C}}_{C2}$. Similarly, two new directed divergence estimates $\widehat{I}_{C1}$ and $\widehat{I}_{C2}$ can be introduced. These new estimates are averages of N logarithmic likelihood ratios as the estimate $\widehat{I}_S$ given in Equation (4.15). The function $\mathrm{LLR}_S(\mathbf{O}_i \mid \cdot, \cdot)$ in Equation (4.15) is replaced with two new functions $\mathrm{LLR}_{C1}(\mathbf{O}_i \mid \cdot, \cdot)$ and $\mathrm{LLR}_{C2}(\mathbf{O}_i \mid \cdot, \cdot)$. The first new function is given as

$$\begin{aligned}\mathrm{LLR}_{C1}(\mathbf{O}_i^{\lambda_i^L \oplus \lambda \oplus \lambda_i^R} \mid \lambda, \kappa) = \frac{1}{T_i}\Big[&\mathrm{SC}_{C1}(\mathbf{O}_i^{\lambda_i^L \oplus \lambda \oplus \lambda_i^R} \mid \lambda) \\ &- \mathrm{SC}_{C1}(\mathbf{O}_i^{\lambda_i^L \oplus \lambda \oplus \lambda_i^R} \mid \kappa)\Big]\end{aligned} \tag{4.17}$$

where λ_i^L and λ_i^R are the left and right context models of the ith token of λ, respectively. The function $\mathrm{SC}_{C1}(\mathbf{O} \mid \cdot)$ is defined in Equation (4.4). The likelihood score function $\mathrm{SC}_{C2}(\mathbf{O} \mid \cdot)$ in Equation (4.5) can be used directly to obtain the second new likelihood ratio function

$$\mathrm{LLR}_{C2}(\mathbf{O}_i^{\lambda_i^L \oplus \lambda \oplus \lambda_i^R} \mid \lambda, \kappa) = \mathrm{SC}_{C2}(\mathbf{O}_i^{\lambda_i^L \oplus \lambda \oplus \lambda_i^R} \mid \lambda) - \mathrm{SC}_{C2}(\mathbf{O}_i^{\lambda_i^L \oplus \lambda \oplus \lambda_i^R} \mid \kappa) \tag{4.18}$$

The normalization by the lenght T_i is not needed, because the functions $\mathrm{SC}_{C2}(\mathbf{O} \mid \cdot)$ already include this normalization term.

4.2.3 Closed Form Solutions based on Simplifying Approximations

Another class of methods proposed for computing the Kullback-Leibler divergence for HMMs are based on some simplifying approximations. These approximations consist of a set of assumptions of the model topologies and approximations made on their behavior. These approximations enable to present the divergence measure finally in a closed form. This section presents three approximations of the above kind. The first of them is presented in [Falkhausen et al. 1995]. The other two are extensions of the latter measure proposed by the author [Vihola et al. 2002].

The closed-form approximation described by Falkhausen et. al. assumed discrete observation density HMMs with similar topologies [Falkhausen et al. 1995]. Moreover, the most likely state sequences of both models were assumed to be equal with the state sequence that generated the token $\mathbf{O}^\lambda$, i.e. $\boldsymbol{q} = \boldsymbol{q}_\lambda^* = \boldsymbol{q}_\kappa^*$. It is straightforward to evaluate the obtained measure, as Monte Carlo simulations are not needed anymore. The resulting approximation can be written in closed form as [Falkhausen et al. 1995]

$$\widehat{I}(\lambda : \kappa) = \sum_i r_i \sum_j a_{ij}^\lambda \log\left(a_{ij}^\lambda / a_{ij}^\kappa\right) + \sum_i r_i \sum_k b_{ik}^\lambda \log\left(b_{ik}^\lambda / b_{ik}^\kappa\right) \tag{4.19}$$

where $A = [a_{ij}]$ is the $N \times N$ transition matrix and $B = [b_{ik}]$ the $N \times M$ observation probability matrix. The probabilities r_i are solved from $\boldsymbol{r}^T = \boldsymbol{r}^T A^\lambda$ with constraint $\sum_i r_i = 1$. They can be interpreted as the probabilities of being in state i in the long-run proportion of time [Ross 1983].

The approximation in Equation (4.19) assumed discrete observation densities. However, the last sum term in Equation (4.19) is the directed divergence between the observation

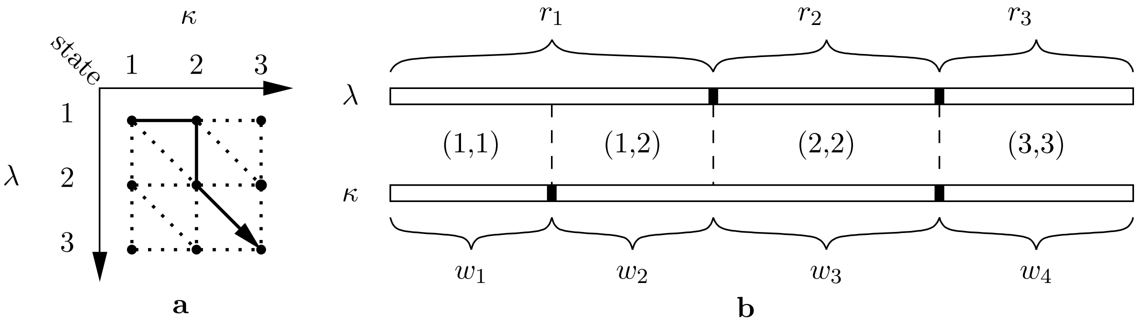

Figure 4.4: State alignment in divergence approximation. The possible state alignments of the models λ and κ are shown as dotted lines in (a). The solid line corresponds the alignment of the models shown in (b). The symbols r_i and w_i are the ones used in Equation 4.22.

distributions of state i of the models λ and κ. Thus, we can rewrite the Equation (4.19) in the form

$$\widehat{I}_{A1}(\lambda:\kappa) = \sum_i r_i \sum_j a^{\lambda}_{ij} \log \left(a^{\lambda}_{ij} / a^{\kappa}_{ij} \right) + \sum_i r_i I(b^{\lambda}_i : b^{\kappa}_i) \tag{4.20}$$

where b^{λ}_i and b^{κ}_i are the observation probability distributions of the models λ and κ in state i, respectively. Now, the only terms that are dependent of the observation probability distributions are the directed divergences between the corresponding distributions of the states, $I(b^{\lambda}_i : b^{\kappa}_i)$. The Equation (4.20) generalizes the Equation (4.19) for arbitrary observation densities. The other simplifying assumptions made in deriving the approximation in Equation (4.19) still remain. In the case of Gaussian observation distributions, the cross-state directed divergences can be expressed in a closed form as [Kullback 1968]

$$\begin{aligned} I(b_1:b_2) = \frac{1}{2}\Bigg[& \log \frac{|\mathbf{\Sigma}_2|}{|\mathbf{\Sigma}_1|} + \mathrm{tr}\left(\mathbf{\Sigma}_1(\mathbf{\Sigma}_2^{-1} - \mathbf{\Sigma}_1^{-1})\right) \\ & + \mathrm{tr}\left(\mathbf{\Sigma}_2^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\right) \Bigg] \end{aligned} \tag{4.21}$$

where $\mathbf{\Sigma}$ and $\boldsymbol{\mu}$ denote the covariance matrices and the mean vectors of the distributions, respectively.

The assumption made in deriving the approximation in Equation (4.20) is that the most likely state sequences of both of the models are equal to the state sequence of the generating model, i.e. $\boldsymbol{q} = \boldsymbol{q}^*_{\lambda} = \boldsymbol{q}^*_{\kappa}$. This assumption can be considered realistic with the model λ. However, there is no reason to assume that the model κ would follow the same sequence. Let us draw one sample from the observation distribution of i:th state of the model λ. It is easy to agree that in most of the cases the likelihood of the generating distribution is greater than likelihoods given by other distributions. This justifies why the most likely state sequence of model λ should correspond the generating sequence, $\boldsymbol{q}^*_{\lambda} = \boldsymbol{q}$ in average. Considering the state sequence $\boldsymbol{q}^*_{\kappa}$ of the model κ, the best likelihood for this model is most likely given by the state-dependent density closest to the corresponding density of λ. This gives us a reason to find such a state alignment between the models λ and κ that the states with best matching distributions coincide.

Assume now that the both HMMs have left-to-right topologies, such as the model shown in Figure 2.3. The transitions in these models are either self-transitions or transitions to the successor state[6]. We introduce now a set of possible state alignments $\mathcal{S}$ between the models λ and κ. The alignments are illustrated in Figure 4.4 (a) for HMMs having three emitting states. The alignments are restricted such that the start and end points of the models are fixed, and occur at the same time. The transitions of model κ can drift under the restriction that the segmentation of one state of λ can be divided into segments of equal duration. A new measure can be obtained for this kind of left-to-right models in the following way. The divergence measure is evaluated for each possible state-to-state alignment in a similar way to Equation (4.20), and the alignment is picked that produces the minimum value. In other words, the divergence measures between the states are minimized when the corresponding observation distributions are most similar. The derived measure can be expressed as [Vihola et al. 2002]

$$\begin{aligned}\widehat{I}_{A2}(\lambda : \kappa) = \min_{(\boldsymbol{q},\boldsymbol{s})\in\mathcal{S}} \Bigg\{ \sum_{i=1}^{L-1} v_{q_i} \Bigg[a^{\lambda}_{q_i q_i} \log \frac{a^{\lambda}_{q_i q_i}}{a^{\kappa}_{s_i s_i}} \\ + a^{\lambda}_{q_i q_{i+1}} \log \frac{a^{\lambda}_{q_i q_{i+1}}}{a^{\kappa}_{s_i s_{i+1}}} + I(b^{\lambda}_{q_i} : b^{\kappa}_{s_i}) \Bigg] \Bigg\}\end{aligned} \tag{4.22}$$

where $\mathcal{S}$ is the set of all possible alignments $(\boldsymbol{q}, \boldsymbol{s})$. The state vectors $\boldsymbol{q}$ and $\boldsymbol{s}$ are defined such that the models λ and κ are at states q_i and s_i, respectively, at step i. The last, non-emitting states of the models are appended to $\boldsymbol{q}$ and $\boldsymbol{s}$. The length of the state alignment vectors is $L = \dim \boldsymbol{q} = \dim \boldsymbol{s}$. The weight vector $\boldsymbol{v}$ is defined as $v_i = r_i/u_i$ where u_i denotes the count of state q_i in the state vector[7] $\boldsymbol{q}$. For example, the values corresponding to the case in Figure 4.4 (b) are $\boldsymbol{q} = [1, 1, 2, 3, 4]^T$, $\boldsymbol{s} = [1, 2, 2, 3, 4]^T$ and $\boldsymbol{v} = [r_1/2, r_1/2, r_2, r_3]^T$.

The approximation in Equation (4.22) can be evaluated for HMMs with different number of states unlike the approximation in Equation (4.20). In both Equations (4.20) and (4.22), the directed divergence measure between state-dependent observation densities can be expressed as in Equation (4.21), if the observation densities are Gaussian. The number of alignments to be evaluated in the minimization task of Equation (4.22) increases rapidly as the number of states in the models increases. The search of this minimun value, however, can be performed more efficiently using dynamic programming algorithm similar to the Viterbi decoding described in Algorithm 2.3.

[6] This kind of model is shown in Figure 2.3.

[7] More precisely, u_i is the number of elements in the set, i.e. $u_i = \text{card}\,\{j \mid q_j = q_i\}$.

Chapter 5

Experimental Setup and Results

The dissimilarity measures covered in Chapter 4 were experimented in the task of phoneme model clustering. The clustering was used for defining a set of multilingual phone models for a multilingual speech recognition system. The word recognition accuracy of the resulting speech recognition systems was evaluated in speaker independent isolated word recognition task. The experiments were evaluated with The Hidden Markov Model Toolkit (HTK), which provided efficient implementations of algorithms used in both training and testing of the speech recognition systems [Young et al. 2000]. The estimation of the dissimilarity measures and the clustering framework were implemented with Matlab 5.3 [MAT].

The content of this chapter is briefly following. Section 5.1 describes the speech corpora and the front-end unit employed for the experiments. After that, the configuration and training of the language dependent baseline ASR systems is explained in Section 5.2. In addition, the test setup and the word recognition accuracies of the baseline systems are described in detail. Section 5.3 covers the derivation and testing of the different multilingual ASR systems. Finally, in Section 5.4, a summary and a comparison of the recognition results of the ML systems are given.

5.1 Speech Corpora and Front-End

All the experiments were performed using the SpeechDat(II) speech corpora of seven languages: English, Finnish, French, German, Italian, Spanish and Swedish [Winski 1997]. The speech utterances in the SpeechDat(II) corpora have been recorded over a fixed telephone network. The sample files are in raw 8bit A-law data format, with 8kHz sampling frequency. The speech utterances in the corpora are annotated in word level, but without any word boundary information. However, the annotation contains additional information of e.g. strong background noise, truncation of a sentence or speaker-specific noise occured during the recording session. In some of the languages, the utterances with speaker-specific noise were included in the experiments, due to shortage of data. The last column in Tables 5.1–5.3 indicate whether such utterances were included in the data sets. The word pronunciation lexicons in the SpeechDat(II) corpora are written with the SAMPA phonetic symbols [SAM]. These phonetic symbols are used in this chapter to denote the phonemes.

The speech material was parametrized using a Mel-frequency cepstral coefficient (MFCC) front-end described in Section 2.3. The frame interval and window length used in the front-end were 10ms and 25ms, respectively. For each frame, the front-end produced a feature vector consisting of 13 Mel-cepstral coefficients of which the zeroth, C_0, was replaced with frame energy E. The first and second time derivatives of the elements were appended

Table 5.1: Summary of the training sets of the five languages used in training of the multilingual speech recognition systems.

Language	Utterances	Speakers	Phonemes	Speaker noise
English	4000	917	44	no
Finnish	4000	1019	46	no
German	4000	1011	47	yes
Italian	4000	1000	51	yes
Spanish	4000	1090	31	no
Total	20000	5037	219	yes

Table 5.2: Summary of the training set of the two unseen languages.

Language	Utterances	Speakers	Phonemes	Speaker noise
French	4052	1427	38	no
Swedish	4000	526	46	no

to the feature vector. After that, mean normalization technique explained in [Viikki and Laurila 1998] was applied to the elements of the feature vector. In addition, the variance of the energy coefficient E and its derivatives ΔE and $\Delta^2 E$ were normalized as described in [Viikki and Laurila 1998].

5.2 Baseline Speech Recognition Systems and Data Sets

The language dependent (LD) baseline recognition system was trained for each of the seven languages mentioned in Section 5.1. Five of the languages shown in Table 5.1 were used for training the multilingual recognition systems, and the two languages in Table 5.2 were used for testing the portability of the multilingual system into new languages. These recognition systems consisted of 31 to 51 monophone HMMs, as shown in Tables 5.1 and 5.2. The model topology was common to all phoneme models in the recognition systems. The phoneme models had the structure shown in Figure 2.3: three emitting states, with self-transitions and transitions to the successor state. Two extra models were used in each recognition system to model silence (SIL) and short pause (SP). The SIL model was similar to the phoneme models, except that it had backward transition from the last state to the first, and a skip transition from the first state to the last. The SP model had only one emitting state tied to the center state of the silence model[1]. The emission probability density functions were GMMs with eight mixture component densities with diagonal covariance matrices in every state.

The speech recognition systems were trained using phonetically rich sentences, labeled with corpus codes S1-9 in SpeechDat(II). The summary of the contents of the training set is shown in Tables 5.1 and 5.2. The training procedure can be outlined as follows

1. All the states of the HMMs shared initially a Gaussian observation density with global mean vector and covariance matrix of the training data. This is known as flat-start initialization of the model emission densities [Young et al. 2000]. All the

[1] The tied states share a common emission probability density.

HMMs, including the SIL model had equivalent initial topology. The SP model was omitted at this stage of the training. The initial state distribution and the transition probabilities of the HMMs were set to $\pi_1 = 1$, $a_{11} = a_{22} = 0.6$, $a_{23} = a_{34} = 0.4$, $a_{33} = 0.7$ and $a_{34} = 0.3$. Other elements of π and $\mathbf{A}$ were constrained zero.

2. The parameters of HMMs were re-estimated using the embedded Baum-Welch (BW) re-estimation procedure explained in Section 2.4.3. Three iterations of the algorithm were performed over the full training data set.
3. Two extra transitions, the skip and the backward transition, were included to the SIL model, $a_{13} = a_{31} = 0.2$. The other transition probabilities of the model were modified such that $\sum_j a_{ij} = 1$, for all i. The SP model was included at this time. The model consisted of one emitting state, namely the state two of SIL. The observation density parameters of the SP state and the second state of SIL were tied. A so called tee-transition, i.e. a transition over the emitting state of the model $\pi_2 = 0.3$, was included to the SP model. Finally, two iterations of BW were performed for the HMMs in the recognition system.
4. A realignment of the training utterances was performed next. This means that the training utterances were recognized with current HMMs, and the best matching pronunciations of the words in the utterance were chosen, when there were duplicate pronunciations in the lexicon. After the realignment of the training utterances, two iterations of the BW re-estimation were performed to the HMMs.
5. The Gaussian emission densities of the HMMs were replaced gradually with mixture-Gaussian densities. The number of mixture components was incremented by one at a time[2], and after each incrementation of the components, an iteration of BW re-estimation was performed over the full training data set. This mixture incrementing was repeated until the final number of mixtures, eight, was achieved. Finally, eight more iterations of the BW re-estimation were performed.

The final LD recognition systems were tested in isolated word recognition task. The summary of the test sets is shown in Table 5.3. The number of vocabulary items was around 200 in each language. The test set of a language covered 3000–4000 samples including application words (A1-3), isolated digits (I1) and forename-surname combinations[3] (O7). The average word recognition rates of the baseline LD recognition systems are shown in Table 5.4. The differences between languages as well as different vocabulary items are substantial. The shorter items, i.e. application words and digits were recognized poorly compared to the longer forename-surname combinations, especially in English and German languages. This reflects into the inferior average word recognition rate (WRR) of these languages.

5.3 Multilingual Recognition Systems

The multilingual (ML) recognition systems were trained using the training data for the five source languages shown in Table 5.1. The training procedure as well as the model configuration were identical to the LD recognizers described in Section 5.2. The only difference was that the label files and the lexicon were changed to correspond to the phone cluster

[2] This mixture incrementation was performed such that the mixture component with the greatest weight was split, meaning that the mixture component was copied, the weights were divided by two, and the means were perturbed by plus or minus 0.2 times the standard deviations [Young et al. 2000].

[3] The forename-surname combinations were recognized as one unit, i.e. the arbitrary combinations of the forenames and surnames were not allowed.

Table 5.3: Summary of the test sets

	Number of utterances							Vocab.	Speaker
Language	A1	A2	A3	I1	O7	Total	Speakers	items	noise
English	800	800	800	800	800	4000	1228	194	no
Finnish	800	800	800	800	800	4000	1024	194	no
German	800	800	800	800	800	4000	1582	196	yes
Italian	800	800	800	800	800	4000	1349	201	yes
Spanish	800	800	800	800	800	4000	1513	193	no
French	600	600	600	600	600	3000	1158	190	yes
Swedish	800	805	807	803	781	3996	1435	190	no
Total	5400	5405	5407	5403	5381	26996	9289	-	yes

Table 5.4: Average word recognition rates of the seven LD baseline recognition systems.

Language	A1	A2	A3	I1	O7	Avg.
English	78.75	78.25	78.62	64.38	91.93	78.40
Finnish	93.91	95.15	94.05	96.50	98.38	95.35
German	82.75	83.50	80.75	75.62	95.16	83.32
Italian	89.50	94.88	87.25	92.38	96.52	92.07
Spanish	98.38	97.23	95.12	93.00	96.44	95.78
French	88.05	87.72	88.28	62.67	92.12	83.57
Swedish	90.06	87.27	89.52	71.64	95.40	85.29

symbols of each ML recognition system. The determination of these phone clusters was based on either expert knowledge or agglomerative clustering. This clustering was based on the dissimilarity measures between the phoneme HMMs of the baseline LD recognition systems. These dissimilarity measures and the agglomerative clustering algorithm are described in Chapter 4 and Algorithm 3.1, respectively. The obtained ML systems were tested similarly as the LD systems described in Section 5.2. During the recognition, only the vocabulary of the target language was set active. The Sections 5.3.1–5.3.3 review the derivation and the experiments performed with the ML recognition systems.

5.3.1 Knowledge-based Multilingual Recognition Systems

The multilingual recognition system SAMPA was based on the method IPA-MAP described in Section 3.3.1. The phonemes of different languages were clustered according to their phonetic SAMPA symbol. The derived SAMPA system had a total of 105 multilingual phone models corresponding to all the unique SAMPA symbols present within the databases of the five source languages.

The second knowledge-based recognition system, referred to as SR, was obtained with straightforward simplifications of the phone cluster definitions of SAMPA. The recognition system had no explicit models for long vowels and double consonants. Such phones were replaced with two single ones, e.g. /e:/ and /ee/ → /e/$\oplus$/e/. In addition, the geminate affricates in Italian language were replaced with the preceding plosive and the following

Table 5.5: Tying of the rare phoneme models in Finnish, Italian and German.

Finnish	Italian	German
/gg/ → Italian /gg/	/@/ → English /@/	/o˜/ → Italian /o/
/bb/ → Italian /bb/	/dz/ → Italian /dZ/	/Z/ → English /Z/
/dd/ → Italian /dd/	/J/ → Spanish /J/	/a˜/ → Italian /a/
/ff/ → Italian /ff/	/L/ → Spanish /L/	/dZ/ → English /dZ/
/hh/ → Finnish /h/	/S/ → German /S/	

affricate, e.g. /ddz/ → /d/$\oplus$/dz/. When these simplifications were employed, the total number of phone models was reduced to 64 from 105 of the SAMPA system.

5.3.2 Multilingual Recognition Systems based on Dissimilarity Measures

All the multilingual systems based on dissimilarity measures had a total of 64 phone models. The phone model definitions in such ML recognition systems were created as follows. The symmetric dissimilarity matrices **D** were created according to the particular dissimilarity estimate. Each dissimilarity matrix was applied in the clustering framework, and the derived phone cluster configuration was used for training a ML recognition system having 64 phone models. The agglomerative clustering used in the phone cluster definition is described in Algorithm 3.1.

The dissimilarity measure estimates were obtained for both Gaussian and eight-mixture GMM observation density HMMs. The phoneme models of the fully trained LD recognition systems were the HMMs with eight-mixture GMM observation densities. The Gaussian observation density HMMs were obtained from the training procedure of the LD recognition systems, just before the first mixture split. The estimates of the dissimilarity measures were obtained using up to 1000 tokens for each LD phoneme model. These tokens were extracted from training speech material according to phoneme level segmentation of the utterances. The segmentation of the utterances was obtained using the fully trained LD recognition systems explained in Section 5.2. The phoneme models that did not have sufficient number of tokens for estimating the dissimilarity measures (under 50 tokens) were not included in the clustering framework. Instead, they were assigned manually to the cluster having the most similar LD phoneme. The tying of these rare phoneme models is shown in Table 5.5. In addition, the clustering was constrained such that phoneme models of the same language were not allowed to locate in a same cluster, excluding the manually tied rare phonemes. This clustering procedure was applied identically with all the dissimilarity measures.

The experiments were carried out with the dissimilarity measures evaluated from the confusion matrix estimates $\widehat{\mathbf{C}}_S$, $\widehat{\mathbf{C}}_{C1}$, $\widehat{\mathbf{C}}_{C2}$, $\widehat{\mathbf{C}}^t_{C1}$ and $\widehat{\mathbf{C}}^t_{C2}$, and from the KL divergence estimates. These KL divergence estimates $\widehat{J}_X$ were obtained according to Equation (4.11) from the corresponding directed divergence measures $\widehat{I}_X$. The three directed divergence estimates $\widehat{I}_S$, $\widehat{I}_{C1}$ and $\widehat{I}_{C2}$, as well as the two closed-form approximations $\widehat{I}_{A1}$ and $\widehat{I}_{A2}$, were employed in the experiments. The latter two approximations were evaluated only for models with Gaussian observation densities, as these approximations cannot be presented in a closed form with mixture observation density HMMs. The measures are referred here according the terminology in Chapter 4. Specifically, the definitions of these estimates are given in Equations (4.2)–(4.22). The number of mixture density components in the

Table 5.6: Approximate values of the dissimilarity measures between English /i:/ and a set of phonemes. The two smallest values in each row are bolded, and the two largest values are in italic.

Diss.	English				German				Spanish		
measure	/I/	/eI/	/e/	/m/	/i:/	/I/	/e:/	/m/	/i/	/e/	/m/
$\widehat{I}_{A1}(1)$	6.57	8.66	21.96	21.10	8.98	**5.52**	**5.43**	*23.00*	9.17	10.65	*27.29*
$\widehat{I}_{A2}(1)$	11.38	15.38	39.24	37.97	15.72	**8.91**	**10.12**	*41.68*	14.24	19.99	*47.34*
$\widehat{I}_{S}(1)$	**3.73**	5.07	15.86	16.31	4.42	4.17	5.50	*17.87*	**3.52**	8.01	*20.05*
$\widehat{I}_{C1}(1)$	**1.28**	2.30	5.20	5.37	1.81	1.57	1.76	*5.70*	**1.33**	3.04	*6.22*
$\widehat{I}_{C2}(1)$	**3.85**	5.07	16.83	17.59	4.57	4.65	5.07	*19.42*	**3.19**	8.84	*21.68*
$\widehat{\mathbf{C}}_{S}(1)$	0.56	0.66	0.83	0.94	**0.52**	0.55	0.55	*0.94*	**0.51**	0.70	*0.95*
$\widehat{\mathbf{C}}_{C1}(1)$	0.58	0.66	0.80	0.90	**0.56**	0.58	**0.57**	*0.91*	0.57	0.70	*0.93*
$\widehat{\mathbf{C}}_{C2}(1)$	**0.45**	0.51	0.65	0.66	**0.45**	0.51	0.52	*0.69*	0.46	0.63	*0.74*
$\widehat{\mathbf{C}}^{t}_{C1}(1)$	0.67	0.71	0.85	0.95	**0.64**	0.67	0.64	*0.95*	**0.64**	0.76	*0.96*
$\widehat{\mathbf{C}}^{t}_{C2}(1)$	0.67	0.65	0.79	0.81	0.61	0.68	**0.59**	*0.83*	**0.58**	0.71	*0.83*
$\widehat{I}_{S}(8)$	**6.14**	8.39	23.71	25.89	6.58	8.44	10.55	*25.65*	**5.36**	13.56	*29.51*
$\widehat{I}_{C1}(8)$	**1.77**	3.23	7.13	7.66	2.39	2.61	3.12	*7.73*	**1.86**	4.24	*8.68*
$\widehat{I}_{C2}(8)$	**5.66**	7.01	23.95	23.47	5.73	7.61	8.90	*24.07*	**4.66**	13.01	*26.94*
$\widehat{\mathbf{C}}_{S}(8)$	0.70	0.74	0.88	0.98	**0.67**	0.70	0.69	*0.98*	**0.64**	0.80	*0.99*
$\widehat{\mathbf{C}}_{C1}(8)$	0.67	0.72	0.86	0.97	**0.64**	0.68	0.66	*0.97*	**0.61**	0.78	*0.98*
$\widehat{\mathbf{C}}_{C2}(8)$	0.58	0.63	0.75	0.78	**0.56**	0.62	0.63	*0.79*	**0.56**	0.71	*0.84*
$\widehat{\mathbf{C}}^{t}_{C1}(8)$	0.73	0.76	0.89	0.98	**0.69**	0.74	0.71	*0.99*	**0.66**	0.82	*0.99*
$\widehat{\mathbf{C}}^{t}_{C2}(8)$	0.80	0.76	0.91	0.94	**0.73**	0.81	0.74	*0.96*	**0.70**	0.85	*0.96*

emission densities of the HMMs employed for the computation of the dissimilarity measures is indicated by the number shown in parenthesis after the symbol of each measure. All the trained ML recognition systems, however, have eight mixture components in emission densities.

In Table 5.6, the values of the dissimilarity measures are shown between English /i:/ and a set of phones from three languages. When these values are ordered, all the measures indicate that either the German or Spanish /m/ is the most dissimilar phoneme model among the phoneme models shown in Table 5.6. The model that is ranked the most similar to the English /i:/ varies according to the different measures. The divergence measures based on approximations, $\widehat{I}_{A1}$ and $\widehat{I}_{A2}$, pick either German /I/ or /e:/ to be most similar to English /i:/. The divergence estimates based on speech data, $\widehat{I}_{S}$, $\widehat{I}_{C1}$ and $\widehat{I}_{C2}$, claim that either English /I/ or Spanish /i/ is the most similar. The measures based on confusion matrix estimates suggest that either German /i:/ or Spanish /i/ is the most similar. Thereby, some variation can be observed in the ranking of the phonemes in Table 5.6, but all the measures seem reasonable.

As an example, the derived full phone cluster definition corresponding to the estimate $\widehat{I}_{C2}(1)$, is shown in Table 5.7. In addition, Table 5.8 shows the number of common clusters between the phone cluster definitions derived using the different dissimilarity measures.

Table 5.7: Phone cluster definitions derived from the dissimilarity estimate $\widehat{I}_{C2}(1)$. The manually assigned rare phonemes are shown in parenthesis. This table continues on Page 41.

Phone cluster	English	Finnish	German	Italian	Spanish
1	U		Y		
2		d		g	
3		oo	aU	O	
4	U@				
5	h	h, (hh)	h		
6				dZ, (dz)	jj
7	I	i	I		
8				LL, (L)	L
9	u:	yy	2:		
10				JJ, (J)	J
11	N	N	N		N
12		(dd)		dd	
13	V		6		
14		nn		nn	D
15	aI		aI		
16		ll		ll	
17	@U	A	a		
18				b	b
19			j	ddZ	
20	3:	22			

Most of these definitions are very different compared against each other. In most of the cases, only less than half of the phone clusters in the definitions are common.

5.3.3 Unseen Languages

There were two unseen languages, French and Swedish, included in the tests. They are unseen, since no data was used from these languages in the training of the ML recognition systems. The tying of the phonemes of these languages was based on phonetic knowledge [Raimo and Savela 2001]. Each phoneme was tied explicitly to one language dependent phoneme model of the five source languages. This tying is shown in Table 5.9. Based on this information, each phoneme of the unseen languages was mapped to a ML phone model accordingly. The tying shown in Table 5.9 was same within all the evaluated ML recognition systems.

The cross-language transfer (CLT) was experimented also for the two unseen languages. The LD recognition systems of the five source languages were employed for this transfer. The systems were obtained for French and Swedish by gathering the necessary phoneme models from the source LD recognizers to form a new CLT recognition system. The gathering of the models was performed according to the tying in Table 5.9, thus similarly as in the case of ML recognition systems. The silence and short pause models of the CLT systems were obtained from the Spanish LD recognition system.

Table 5.7: Continued from Page 40.

Phone cluster	English	Finnish	German	Italian	Spanish
21	OI		OY		
22		NN		vv	g
23			ts	ts	
24	e@	2			
25	tS		tS	tS	
26	@	v	@	v, (@)	B
27	S		C	SS	
28	e	e	E		e
29	l		u:	u	G
30	T	f, (ff)	pf	ff	
31	D	y	y:	ddz	
32	Q	o	O, (o˜)	o	o
33	Z	s	S, (Z)	(S)	s
34		u	U		u
35	r	rr	9	rr	rr
36	A:	AA	a:		
37	b	b, (bb)	b	bb	
38	I@	ee	E:	E	
39	d	(gg)	d	gg	
40	eI		e:	e	
41	f		f	f	f
42	g	g	g	d	d
43	k	k	k	k	k
44		l	l	l	l
45	m	m	m	m	m
46	n	n	n	n	n
47	i:	ii	i:	i	i
48	p	p	p	p	p
49		r	r	r	r
50	s		s	s	
51	t	t	t	t	t
52	v		v	tts	T
53		kk	x	kk	x
54	z		z	z	z
55	O:	uu	o:		
56	dZ		(dZ)	ttS	tS
57	j	j		j	j
58	w			w	w
59	aU	{{			
60	{	{	(a˜)	a	a
61		ss		ss	
62		pp		pp	
63		tt		tt	
64		mm		mm	

Table 5.8: The number of common phone clusters in the phone cluster definitions obtained using the dissimilarity measures. The phone cluster definitions that have at least 32 common clusters are bolded. This number is half of the total number of the phone clusters in each ML system.

	$\widehat{I}_{A2}(1)$	$\widehat{I}_{S}(1)$	$\widehat{I}_{C1}(1)$	$\widehat{I}_{C2}(1)$	$\widehat{\mathbf{C}}_{S}(1)$	$\widehat{\mathbf{C}}_{C1}(1)$	$\widehat{\mathbf{C}}_{C2}(1)$	$\widehat{\mathbf{C}}^{t}_{C1}(1)$	$\widehat{\mathbf{C}}^{t}_{C2}(1)$	$\widehat{I}_{S}(8)$	$\widehat{I}_{C1}(8)$	$\widehat{I}_{C2}(8)$	$\widehat{\mathbf{C}}_{S}(8)$	$\widehat{\mathbf{C}}_{C1}(8)$	$\widehat{\mathbf{C}}_{C2}(8)$	$\widehat{\mathbf{C}}^{t}_{C1}(8)$	$\widehat{\mathbf{C}}^{t}_{C2}(8)$
$\widehat{I}_{A1}(1)$	23	12	12	11	8	9	9	10	7	11	9	9	8	8	8	7	6
$\widehat{I}_{A2}(1)$	–	17	20	15	14	15	13	14	13	12	15	13	12	11	13	10	9
$\widehat{I}_{S}(1)$	–	–	22	26	21	24	20	24	18	22	14	20	22	19	17	19	16
$\widehat{I}_{C1}(1)$	–	–	–	28	25	26	21	21	16	21	22	26	20	18	21	17	15
$\widehat{I}_{C2}(1)$	–	–	–	–	23	24	19	21	20	26	19	26	22	20	19	20	20
$\widehat{\mathbf{C}}_{S}(1)$	–	–	–	–	–	**35**	22	29	22	23	23	29	29	29	24	26	20
$\widehat{\mathbf{C}}_{C1}(1)$	–	–	–	–	–	–	27	**38**	23	**32**	28	**38**	29	30	26	30	27
$\widehat{\mathbf{C}}_{C2}(1)$	–	–	–	–	–	–	–	21	20	20	21	24	23	25	24	22	21
$\widehat{\mathbf{C}}^{t}_{C1}(1)$	–	–	–	–	–	–	–	–	23	30	21	31	30	28	24	28	19
$\widehat{\mathbf{C}}^{t}_{C2}(1)$	–	–	–	–	–	–	–	–	–	19	19	20	26	23	19	23	22
$\widehat{I}_{S}(8)$	–	–	–	–	–	–	–	–	–	–	25	**36**	30	28	20	30	26
$\widehat{I}_{C1}(8)$	–	–	–	–	–	–	–	–	–	–	–	**34**	31	29	26	29	19
$\widehat{I}_{C2}(8)$	–	–	–	–	–	–	–	–	–	–	–	–	**34**	31	28	**32**	25
$\widehat{\mathbf{C}}_{S}(8)$	–	–	–	–	–	–	–	–	–	–	–	–	–	**45**	24	**46**	24
$\widehat{\mathbf{C}}_{C1}(8)$	–	–	–	–	–	–	–	–	–	–	–	–	–	–	26	**53**	29
$\widehat{\mathbf{C}}_{C2}(8)$	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	24	25
$\widehat{\mathbf{C}}^{t}_{C1}(8)$	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	28

5.4 Comparison of Recognition Results

The average word recognition rates of the LD and ML speech recognition systems are shown for the test sets in Tables 5.10 and 5.11. The two highest average WRRs of the ML systems are bolded, and the two lowest WRRs are shown in italic. In Tables 5.10 and 5.11, the ML recognition systems SAMPA and SR are based on expert knowledge, while the other ML recognition systems are based on dissimilarity measures. Table 5.10 shows that the degradation in WRR of the test set was moderate within the five source languages, when comparing the LD recognition systems and the ML recognition system SAMPA having 105 phone models. The further reduction of the WRR was small, when the number of ML phone models was dropped into 64 in the SR system. All the ML recognition systems based on computational model tying and employing the data-driven dissimilarity measures can be considered comparable to the knowledge based SR system having 64 phone models. The systems based on approximations $\widehat{I}_{A1}$ and $\widehat{I}_{A2}$ had comparable WRRs to the other ML systems. The WRRs of all these ML recognition systems did not differ significantly. However, these results show that a ML speech recognition system can be built, in this case, using only 30% of the number of parameters that are used in the LD recognition systems.

The test set WRRs of the unseen languages are shown in Table 5.11. The WRRs of the

Table 5.9: Mapping of the phonemes of the two unseen languages to the phones included in ML recognition systems. In Swedish, the new retroflex consonant phonemes were split into separate phones, e.g. /rn/ → /r/⊕/n/, and then mapped accordingly. Similarly, the unseen nasal vowels of French were split into separate vowel and velar nasal, e.g. /9/ → /9/⊕/N/.

Mapping of the French phonemes								
/A/	→	English /A:/	/e/	→	Italian /e/	/p/	→	Italian /p/
/E/	→	Italian /E/	/f/	→	Italian /f/	/2/	→	German /2:/
/e˜/	→	German /E/	/9˜/	→	German /9/	/s/	→	Italian /s/
/H/	→	Finnish /y/	/g/	→	Italian /g/	/t/	→	Italian /t/
/J/	→	Italian /J/	/h/	→	English /h/	/u/	→	Italian /u/
/N/	→	German /N/	/i/	→	Italian /i/	/v/	→	Italian /v/
/O/	→	Italian /O/	/j/	→	Italian /j/	/9/	→	German /9/
/R/	→	German /r/	/k/	→	Italian /k/	/w/	→	Italian /w/
/S/	→	Italian /S/	/l/	→	Italian /l/	/o˜/	→	German /o˜/
/Z/	→	English /Z/	/m/	→	Italian /m/	/y/	→	German /y:/
/a/	→	Italian /a/	/n/	→	Italian /n/	/z/	→	German /z/
/b/	→	Italian /b/	/o/	→	Italian /o/			
/d/	→	Italian /d/	/a˜/	→	German /a˜/			

Mapping of the Swedish phonemes								
/C/	→	German /C/	/b/	→	English /b/	/n/	→	German /n/
/E/	→	German /E/	/E:/	→	German /E:/	/i:/	→	German /i:/
/u:/	→	German /u:/	/d/	→	English /d/	/2/	→	Finnish /2/
/I/	→	German /I/	/e:/	→	German /e:/	/p/	→	German /p/
/9:/	→	German /9/	/e/	→	Finnish /e/	/r/	→	Finnish /r/
/N/	→	German /N/	/f/	→	German /f/	/s/	→	German /s/
/O/	→	German /O/	/g/	→	English /g/	/t/	→	German /t/
/y:/	→	German /y:/	/h/	→	German /h/	/{/	→	Finnish /{/
/S/	→	German /S/	/{:/	→	Finnish /{{/	/v/	→	English /v/
/U/	→	German /U/	/j/	→	German /j/	/9/	→	German /9/
/Y/	→	German /Y/	/k/	→	German /k/	/o:/	→	German /o:/
/A:/	→	English /A:/	/l/	→	German /l/	/u0/	→	English /U/
/2:/	→	German /2:/	/m/	→	German /m/			
/a/	→	Italian /a/	/}:/	→	English /u:/			

ML systems were significantly lower compared to the corresponding LD systems, especially for French. The variation on the average WRR is much greater in the unseen languages compared to the source languages of the ML recognition systems. The systems that were based on cross-language transfer (CLT) achieved WRRs that are comparable to the ML systems. Despite the fact that the WRRs of the ML systems was low compared to the corresponding results of the baseline LD recognition systems of the unseen languages, this can be considered as a fair starting point for language or speaker adaptation.

All the ML recognition systems having 64 phone models had comparable WRRs. The phone cluster definitions of these systems were, however, fundamentally different as shown in Table 5.8. Small differences in values of the dissimilarity measures may cause big difference in the cluster definitions, due to the nature of the employed clustering algorithm. Furthermore, the dissimilarity measure approximations $\widehat{I}_{A1}$ and $\widehat{I}_{A2}$ showed to be applicable to the task of multilingual phoneme model clustering. They have, however, significantly lower computational cost compared to the data-driven dissimilarity measures.

Table 5.10: Average test set WRRs of the source languages of the ML recognition systems.

Recognition system	English	Finnish	German	Italian	Spanish	Avg.
LD	78.40	95.35	83.32	92.07	95.78	88.98
SAMPA	63.38	92.30	79.60	93.18	94.55	**84.60**
SR	59.67	91.07	79.07	92.68	93.22	83.14
$\widehat{I}_{A1}(1)$	65.75	89.40	78.10	89.10	91.28	82.73
$\widehat{I}_{A2}(1)$	64.43	90.05	78.60	91.75	91.45	83.26
$\widehat{I}_{S}(1)$	63.00	90.97	80.12	92.00	93.00	83.82
$\widehat{I}_{C1}(1)$	65.40	90.65	78.80	92.40	91.70	83.79
$\widehat{I}_{C2}(1)$	65.53	90.95	78.88	93.00	93.62	**84.40**
$\widehat{\mathbf{C}}_{S}(1)$	63.12	90.88	78.78	92.32	92.88	83.60
$\widehat{\mathbf{C}}_{C1}(1)$	65.55	90.07	78.95	92.20	93.47	84.05
$\widehat{\mathbf{C}}_{C2}(1)$	63.90	88.10	77.70	91.15	92.78	*82.73*
$\widehat{\mathbf{C}}^{t}_{C1}(1)$	65.80	89.68	79.28	91.35	93.55	83.93
$\widehat{\mathbf{C}}^{t}_{C2}(1)$	64.40	90.32	79.32	91.62	92.72	83.68
$\widehat{I}_{S}(8)$	64.53	90.55	76.85	92.10	93.30	83.47
$\widehat{I}_{C1}(8)$	67.78	91.15	77.64	90.60	92.20	83.87
$\widehat{I}_{C2}(8)$	63.47	89.20	77.38	92.70	93.43	83.24
$\widehat{\mathbf{C}}_{S}(8)$	64.35	89.75	78.20	92.05	94.38	83.75
$\widehat{\mathbf{C}}_{C1}(8)$	66.05	88.97	77.72	92.10	92.95	83.56
$\widehat{\mathbf{C}}_{C2}(8)$	63.03	89.32	77.45	89.75	91.57	*82.22*
$\widehat{\mathbf{C}}^{t}_{C1}(8)$	66.88	88.53	79.25	93.35	93.20	84.24
$\widehat{\mathbf{C}}^{t}_{C2}(8)$	66.47	89.60	78.78	92.62	92.20	83.93

Table 5.11: Average test set WRRs of the unseen languages.

Rec. system	French	Swedish	Avg.	Rec. system	French	Swedish	Avg.
LD	82.77	85.29	84.03	CLT	55.90	68.31	62.11
SAMPA	49.73	69.52	59.62	$\widehat{I}_{A1}(1)$	46.72	66.54	*56.63*
SR	50.03	66.27	*58.15*	$\widehat{I}_{A2}(1)$	54.77	66.68	60.73
$\widehat{I}_{S}(1)$	56.50	68.82	62.66	$\widehat{I}_{S}(8)$	54.80	65.54	60.17
$\widehat{I}_{C1}(1)$	52.90	68.74	60.82	$\widehat{I}_{C1}(8)$	60.00	67.49	63.75
$\widehat{I}_{C2}(1)$	58.30	68.69	63.50	$\widehat{I}_{C2}(8)$	56.29	68.39	62.34
$\widehat{\mathbf{C}}_{S}(1)$	54.33	66.42	60.38	$\widehat{\mathbf{C}}_{S}(8)$	58.97	65.39	62.18
$\widehat{\mathbf{C}}_{C1}(1)$	53.13	66.14	59.64	$\widehat{\mathbf{C}}_{C1}(8)$	58.63	66.79	62.71
$\widehat{\mathbf{C}}_{C2}(1)$	58.23	65.89	62.06	$\widehat{\mathbf{C}}_{C2}(8)$	56.33	67.67	62.00
$\widehat{\mathbf{C}}^{t}_{C1}(1)$	54.57	64.46	59.52	$\widehat{\mathbf{C}}^{t}_{C1}(8)$	61.43	66.79	**64.11**
$\widehat{\mathbf{C}}^{t}_{C2}(1)$	57.03	67.46	62.24	$\widehat{\mathbf{C}}^{t}_{C2}(8)$	61.50	67.39	**64.44**

Chapter 6

Conclusions

This thesis covered the dissimilarity measures for hidden Markov models (HMMs). They were researched especially for the purposes of multilingual (ML) speech recognition. These measures are needed when a compact set of ML phone models is created from a large set of language dependent (LD) phoneme models by using a computational clustering method. This clustering can be done such that the dissimilarity measures are evaluated between the LD phoneme models, and the phonemes that are close to each other are represented with a common ML phone model. Such a phone model is shared across the source languages and may also be ported for new languages. The source languages of the ML recognition systems were English, Finnish, German, Italian and Spanish. The ML recognition were experimented also with two unseen languages: French and Swedish.

The dissimilarity measures researched for HMMs can be grouped into two categories: the methods that are based on confusion matrix estimation, and those methods that are based on Kullback-Leibler divergence. The different dissimilarity measures were compared against each other in the task of phoneme model clustering. The experiments covered the evaluation of 18 different dissimilarity measures for the total of 219 LD phoneme model HMMs of five languages. The data-driven dissimilarity measures were estimated using the training data of the LD recognition systems. For each measure, a dissimilarity matrix was formed. Based on that matrix, these LD phoneme models were clustered into 64 clusters, each of which corresponded to one ML phone model. After that, new vocabulary independent ML recognition system was trained according to each phone cluster definition. These ML systems having 64 phone models had approximately 30% of the number of parameters in the five LD recognition systems. In addition, the recognition systems based on the computational definition of phone clusters were compared to the recognition systems with knowledge-based phone cluster definitions. Particularly, a ML recognition system having 105 ML phone models corresponding to unique SAMPA phonemes in the source languages, was included in the experiments. The second knowledge-based system, that was included in the experiments, had no separate models for long vowels, double consonants and geminate affricates. They were represented with the corresponding single phonemes. This system, having reduced set of SAMPA phones, had a total of 64 ML phone models.

The ML recognition systems were tested in isolated word recognition with approximately 200 word vocabulary for each language. Only the vocabulary of the target language was set active during the recognition. The test sets consisted of approximately 4000 utterances for each language. The baseline LD recognition systems of the source languages had an average word recognition rate (WRR) of 89.0%. The average WRRs of all the ML recognition were surprisingly close to each other. The SAMPA recognition system having 105 ML phone models had an average WRR of 84.6%. The other knowledge-based system, which had 64 phone models, had WRR of 83.1%. The systems having phone cluster definitions derived using different dissimilarity measures had WRRs between 82.2% and 84.4%. All these ML recognition systems had 64 phone models.

All the ML recognition systems were also tested with two new languages. These languages were not included in the training of the ML recognition systems. The baseline LD recognition systems of French and Swedish had the average WRRs of 82.8% and 85.3%, respectively. The mapping of the phonemes of the new languages into the ML phone models was defined using phonetic knowledge. This mapping was performed for each ML recognition system. Only moderate differences were observed in the test set WRRs when the two new languages were recognized with the ML recognition systems. The WRRs of French and Swedish ranged from 46.7% to 61.5%, and from 58.2% to 69.5%, respectively. Despite the fact that these WRRs are very low compared to corresponding WRRs of the baseline LD recognition systems, they can be considered as a fair starting point for language or speaker adaptation.

All of the measures employed in the experiments can be considered applicable in the task of phoneme HMM clustering. It is interesting, that WRRs in these experiments were close to each other, as the phone cluster definitions differed substantially among the measures. The phone cluster definitions, that were derived using the different dissimilarity measures had, in most of the cases, less than half common clusters. Moreover, the proposed closed form measures having low computational cost proved to be equally applicable to the task. The assumptions that were made with these measures are fulfilled with common left-to-right proceeding phoneme HMMs. Furthermore, an closed form approximation was proposed, that allows the HMMs to consist of different number of states. These measures, that have closed form representation with respect to the model parameters, are also useful when no training data, but a fully trained LD HMMs are available.

A possible future work topic is the definition of transformation class tree for MLLR model adaptation scheme. This definition could be performed using the dissimilarity measures presented in this thesis. The HMM level definition of the transformation classes, which is achieved by using the dissimilarity measures for HMMs, has been mentioned as an advantage, when performing MLLR speaker adaptation.

References

Adda-Decker M. Towards multilingual interoperability in automatic speech recognition. *Speech Communication*, 35(1-2):5–20, Aug. 2001.

Andersen O., Dalsgaard P. and Barry W. On the use of data-driven clustering technique for identification of poly- and mono-phonemes for four european languages. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, volume 1, pp. 121–124, Adelaide, Australia, Apr. 1994.

Baum L. An inequality and associated maximization technique in statistical estimation of probabilistic functions of a Markov process. *Inequalities*, 3:1–8, 1972.

Bishop C. *Neural Networks for Pattern Recognition.* Oxford University Press, New York, 1998.

Byrne W., Beyerlein P., Huerta J., Khudanpur S., Marthi B., Morgan J., Peterek N., Picone J., Vergyri D. and Wang W. Towards language independent acoustic modeling. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, volume II, pp. 1029–1032, Istanbul, Turkey, 2000.

Chang E., Zhou J., Di S., Huang C. and Lee K.-F. Large vocabulary Mandarin speech recognition with different approaches in modeling tones. In *Proceedings of the International Conference on Spoken Language Processing*, pp. 16–20, Beijing, China, Oct. 2000.

Cheour M., Martynova O., Näätänen R., Erkkola R., Sillanpää M., Kero P., Raz A., Kaipio M.-L., Hiltunen J., Aaltonen O., Savela J. and Hämäläinen H. Psychobiology: Speech sounds learned by sleeping newborns. *Nature*, 415:599–600, 2002.

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

Deller J., Proakis G. and Hansen J. *Discrete-Time Processing of Speech Signals.* The Institute of Electrical and Electronics Engineers Inc., New York, 2nd edition, 2000.

Dempster A., Laird N. and Rubin D. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

Falkhausen M., Reininger H. and Wolf D. Calculation of distance measures between hidden Markov models. In *Proceedings of the European Conference of Speech Communicationn and Technology*, pp. 1487–1490, Madrid, 1995.

Fung P., Ma C. and Liu W. MAP-based cross-language adaptation augmented by linguistic knowledge from English to Chinese. In *Proceedings of the European Conference of Speech Communication and Technology*, volume II, pp. 871–874, Budapest, Hungary, Sept. 1999.

Furui S. Speaker independent isolated word recognition using dynamic features of speech spectrum. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(1):52–59, Feb. 1986.

Furui S. Flexible speech recognition. In *Proceedings of the European Conference of Speech Communication and Technology*, volume 4, pp. 352–359, Sept. 1995.

Gales M. The generation and use of regression class trees for MLLR adaptation. Technical Report CUED/F-INFENG/TR. 263, Cambridge University Engineering Department, U.K., Aug. 1996.

Gales M. Semi-tied covariance matrices for hidden Markov models. *IEEE Transactions on Speech and Audio Processing*, 7(3):272–281, May 1999.

Gales M. and Olsen P. Tail distribution modelling using the richter and power exponential distributions. In *Proceedings of the European Conference of Speech Communication and Technology*, volume 4, pp. 1507–1510, Budapest, Hungary, 1999.

Gariepy R. and Ziemer W. *Modern Real Analysis.* International Tomson Publishing, 1994.

Gauvain J.-L. and Lee C.-H. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *IEEE Transactions on Speech and Audio Processing*, 2(2):291–298, Apr. 1994.

Gold B. and Morgan N. *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*. John Wiley & Sons, Inc., New York, 2000.

Haeb-Umbach R. Automatic generation of phonetic regression class trees for MLLR adaptation. *IEEE Transactions on Speech and Audio Processing*, 9(3), Mar. 2001.

Hariharan R. *Robust Signal Parametrisation Techniques for Speech Recognition*. Dr. Tech. thesis, Tampere University of Technology, Publications 317, Finland, 2001.

Harju M., Salmela P., Leppänen J., Viikki O. and Saarinen J. Comparing parameter tying techniques for multilingual acoustic modelling. In *Proceedings of the European Conference of Speech Communication and Technology*, pp. 2729–2732, Aalborg, Denmark, Sept. 2001.

Imperl B. and Horvat B. The clustering algorithm for the definition of multilingual set of context dependent speech models. In *Proceedings of the European Conference of Speech Communication and Technology*, pp. 887–890, Budabest, Hungary, 1999.

Jelinek F. *Statistical Methods for Speech Recognition*. The MIT Press, Cambridge, MA, 1998.

Johnson R. and Wichern D. *Applied Multivariate Statistical Analysis*. Prentice-Hall, Inc., Upper Saddle River, NJ, 4th edition, 1998.

Juang B.-H. and Rabiner L. A probabilistic distance measure for hidden Markov models. *AT&T Technical Journal*, 64(2):391–408, 1985.

Junqua J.-C. *Robust Speech Recognition in Embedded System and PC Application*. Kluwer Academic Publishers Group, Boston, 2000.

Karjalainen M. Kommunikaatioakustiikka. Technical Report 51, Helsinki University of Technology, Laboratory of Acoustics and Audio Signal Processing, Espoo, Finland, 1999. Preprint, In Finnish.

Kiss I. *On Speech Recognition In Mobile Communications*. Dr. Tech. thesis, Tampere University of Technology, Publications 326, Finland, 2001.

Kullback S. *Information Theory and Statistics*. Dover Publications, New York, 1968.

Köhler J. Language adaptation of multilingual phone models for vocabulary independent speech recognition tasks. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, volume 1, pp. 417–420, Seattle, WA, May 1998.

Köhler J. Comparing three methods to create multilingual phone models for vocabulary independent speech recognition tasks. In *Proc. ESCA-NATO Tutorial and Research Workshop: Multi-lingual Interoperability in Speech Technology*, pp. 79–84, Sept. 1999.

Köhler J. *Erstellung einer statistisch modellierten multilingualen Lautbibliotek für die Spracherkennung*. PhD thesis, der Technischen Universität München, 2000. In German.

Köhler J. Multilingual phone models for vocabulary-independent speech recognition tasks. *Speech Communication*, 35(1-2):21–30, Aug. 2001.

Ladefoged P., Local J. and Shockey L., editors. *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge University Press, U.K., 1999.

Laurila K. *Robust Speech Recognition Methods for Voice Dialing*. Dr. Tech. thesis, Tampere University of Technology, Publications 298, Finland, 2000.

Lee L.-S. Voice dictation of Mandarin Chinese. *IEEE Signal Processing Magazine*, 14(4):63–101, July 1997.

Leggetter C. and Woodland P. Speaker adaptation of HMMs using linear regression. Technical Report CUED/F-INFENG/TR. 181, Cambridge University Engineering Department, U.K., June 1994.

Leggetter C. and Woodland P. Flexible speaker adaptation using maximum likelihood linear regression. In *Proceedings of the European Conference of Speech Communication and Technology*, pp. 1155–1158, Madrid, Spain, 1995.

Leppänen J., Salmela P., Harju M., Pärssinen K., Viikki O. and Saarinen J. Language adaptation of multilingual recognition system using MAP and MLLR. In *Proceedings of Internationa Conference on Speech Processing*, volume 1, pp. 87–92, Taejon, Korea, Aug. 2001.

MAT. *Using Matlab (Revised for Matlab 5.3, Release 11)*. The MathWorks Inc., 1999.

Muthusamy Y., Barnard E. and Cole R. Reviewing automatic language identification. *IEEE Signal Processing Magazine*, 11(4):33–41, Oct. 1994.

Muthusamy Y., Cole R. and Oshika B. The OGI multi-language telephone speech corpus. In *Proceedings of the International Conference on Spoken Language Processing*, volume 2, pp. 895–898, Banff, Alberta, Canada, Oct. 1992.

Navrátil J. Spoken language recognition—a step toward multilinguality in speech recognition. *IEEE Transactions on Speech and Audio Processing*, 9(6):678–685, Sept. 2001.

Rabiner L. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, Feb. 1989.

Rabiner L. *Fundamentals of Speech Recognition*. PTR Prentice-Hall Inc., New Jersey, 1993.

Raimo I. and Savela J. Personal communication. Department of Phonetics, University of Turku, Finland, 2001.

Reynolds D., Quatieri T. and Dunn R. Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing*, 10(1):19–41, Jan. 2000.

Ross S. *Stochastic Processes*. John Wiley & Sons, Inc., New York, 1983.

Rossing T. *The Science of Sound*. Addison-Wesley Publishing Company, Reading, Massachusetts, 2nd edition, 1990.

Rosti A.-V. and Gales M. Generalised linear Gaussian models. Technical Report CUED/F-INFENG/TR. 420, Cambridge University Engineering Department, U.K., Nov. 2001.

SAM. ESPRIT project 2589: Final report – year three. Technical Report SAM-UCL-G004, University College London, Department of Phonetics and Linguistics, England, 1989. URL `http://www.phon.ucl.ac.uk/home/sampa/home.htm`.

Schultz T. and Waibel A. Polyphone decision tree specialization for language adaptation. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, volume 3, pp. 1707–1710, Istanbul, Turkey, June 2000.

Schultz T. and Waibel A. Experiments on cross-language acoustic modeling. In *Proceedings of the European Conference of Speech Communication and Technology*, pp. 2721–2724, Aalborg, Denmark, Sept. 2001.

Schultz T., Westphal M. and Waibel A. The GlobalPphone project: Multilingual LVCSR with JANUS-3. In *Multilingual Information Retrieval Dialogs: 2nd SQEL Workshop*, pp. 20–27, Plzen, Czech Republic, Apr. 1997.

Stoica P. and Moses R. *Introduction to Spectral Analysis*. Prentice-Hall Inc., New Jersey, 1997.

Theodoridis S. and Koutroumbas K. *Pattern Recognition*. Academic Press, San Diego, CA, 1999.

Turunen E. Survey of theory and applications of Lukasiewicz-Pavelka fuzzy logic. In di Nola A. and Gerla G., editors, *Lectures on Soft Computing and Fuzzy Logic. Advances in Soft Computing*, pp. 313–337. Physica-Verlag, Heidelberg, 2001.

Uebler U. Multilingual speech recognition in seven languages. *Speech Communication*, 35(1-2): 53–69, Aug. 2001.

Van Compernolle D. Recognizing speech of goats, wolves, sheep and ... non-natives. *Speech Communication*, 35(1-2):71–79, Aug. 2001.

Vihola M., Harju M., Salmela P., Suontausta J. and Savela J. Two dissimilarity measures for HMMs and their application in phoneme model clustering. Accepted to *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, Orlando, USA, 2002.

Viikki O. *Adaptive Methods for Robust Speech Recognition*. Dr. Tech. thesis, Tampere University of Technology, Publications 257, Finland, 1999.

Viikki O., Kiss I. and Tian J. Speaker- and language-independent speech recognition in mobile communication systems. In *Proceedings of International Conference in Acoustics, Speech and Signal Processing*, volume I, pp. 5–8, Salt Lake City, Utah, May 2001.

Viikki O. and Laurila K. Cepstral domain segmental feature normalization for noise robust speech recognition. *Speech Communication*, 25(1-3):133–147, Aug. 1998.

Viterbi A. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13:260–267, 1967.

Waibel A., Geutner P., Tomokiyo L., Schultz T. and Woszczyna M. Multilinguality in speech and spoken language systems. *Proceedings of the IEEE*, 88(8), Aug. 2000.

Winski R. SpeechDat: Definition of corpus, scripts and standards for fixed networks. Technical Report LE2-4001-SD1.1.1, http://www.speechdat.org/, Jan. 1997.

Woodland P. and Young S. The HTK tied-state continuous speech recogniser. In *Proceedings of the European Conference of Speech Communication and Technology*, volume 3, pp. 2207–2210, Berlin, 1993.

Young S., Adda-Decker M., Aubert X., Dugast C., Gauvain J.-L., Kershaw D., Lamel L., Leeuwen D., Pye D., Robinson A., Steeneken H. and Woodland P. Multilingual large vocabulary speech recognition: The European SQALE project. *Computer Speech and Language*, 11(1): 73–89, 1997.

Young S., Kershaw D., Odell J., Ollason D., Valtchev V. and Woodland P. *The HTK Book (for HTK Version 3.0)*. Cambridge University Engineering Department, U.K., July 2000.

Young S., Russell N. and Thornton J. Token passing: A simple conceptual model for connected speech recognition systems. Technical Report CUED/F-INFENG/TR. 38, Cambridge University Engineering Department, U.K., July 1989.

Žgank A., Imperl B. and Johansen F. Crosslingual speech recognition with multilingual acoustic models based on agglomerative and tree-based triphone clustering. In *Proceedings of the European Conference of Speech Communication and Technology*, pp. 2725–2728, Aalborg, Denmark, Sept. 2001.

Appendix A

Transformation Classes for MLLR

In the MLLR adaptation framework, discussed in Chapter 2, the target densities are assigned into transformation classes to ensure the sufficient data to estimate the transform matrices. All the densities in one transformation class are assumed to have similar acoustic properties, as they are transformed using a common transformation matrix [Leggetter and Woodland 1994]. Usually a transformation class tree is built to enable dynamic use of the transformation classes. This means that the more adaptation data is available, the more specific transforms can be used [Leggetter and Woodland 1995]. An example of the transformation class tree is exemplified in Figure A.1. This transformation class tree is usually built either by using phonetic knowledge, or by some acoustic clustering algorithm.

The dissimilarity measures, discussed in Chapter 4, can also be employed for the definition of the transformation class tree. The Sections A.1 and A.2 briefly describe the different methods that have been used for defining a transformation class tree. The Section A.1 describes some acoustic clustering algorithms that form a tree consisting of the mixture components of the state-dependent densities of HMMs. The Section A.2 discusses the HMM level definition of the transformation class tree. Most commonly used method for defining a HMM level transformation class tree is the phonetic knowledge, but also a computational method has been proposed [Haeb-Umbach 2001].

A.1 Mixture Component Level Definition

Some acoustic clustering algorithms used in the definition of the transformation classes utilize a distance measure between two mixture components. Based on the measure, a tree of mixture component classes can be created using e.g. the agglomerative clustering scheme shown in Algorithm 3.1. The advantages of this method are that expert phonetic knowledge is not needed, and the construction of the tree can be done using the available adaptation data [Leggetter and Woodland 1995, Young et al. 2000]. The disadvantage of this mixture component level method is that the mixture components of a single HMM, or even state, can be transformed using several different transformation matrices [Leggetter and Woodland 1994].

The acoustic clustering method used by Leggetter and Woodland employed the Kullback-Leibler divergence between two Gaussian mixture density components as a distance measure [Leggetter and Woodland 1995]. The algorithm implemented in Hidden Markov Model Toolkit (HTK) introduces occupation counters[1] in addition to the KL divergence to create

[1] The occupation counters are the mean number of feature vectors that are assigned to a particular mixture component. They can be obtained using the posteriori probabilities $\gamma_t(j,k)$ defined in Section 2.4.3.

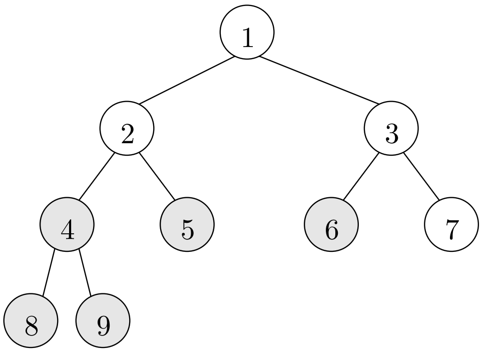

Figure A.1: An example of a transformation class tree that may be used in MLLR adaptation. Gray circles denote nodes with insufficient data. In this example, only three transformation matrices are computed: for nodes 2, 3 and 7. The components in terminal nodes (TNs) 8, 9 and 5 are transformed using one common transformation matrix. The TN 7 has its own specific transformation, and the components in TN 6 are transformed with a transformation matrix calculated using the data from both TNs 6 and 7.

the transformation classes [Young et al. 2000]. This enables the creation of such clusters that utilize the specific adaptation data more efficiently. Gales has proposed a method that is optimal in the sense of maximizing the likelihood of the transformation class usage. The results, however, differed very little from the acoustic clustering used as a baseline [Gales 1996]. Furthermore, this method is computationally expensive.

A.2 HMM Level Definition

The use of phonetic knowledge in the generation of transformation class tree ensures that all the mixture component densities in one HMM are transformed using a common transform matrix [Leggetter and Woodland 1994]. The disadvantage of this method is the use of expert knowledge which may not be available. Moreover, the phonetic trees are not unique as the phonetic categories are not strictly hierarchical [Raimo and Savela 2001]. An example of a phonetic transformation class tree is shown in Figure A.2.

The dissimilarity measures presented in Chapter 4 could be also used to define a transformation class tree in HMM level. The agglomerative clustering procedure described in Algorithm 3.1 produces a binary tree [Theodoridis and Koutroumbas 1999]. The question of using dissimilarity measures for HMMs for defining a transformation class tree is out of the scope of this thesis, but is mentioned here for possible future work. A computational HMM level transformation class tree definition has been reported earlier in [Haeb-Umbach 2001]. The dissimilarity measure employed in that research was based on inter-speaker correlation [Haeb-Umbach 2001]. The advantages of this type of transformation class tree definition is the HMM level definition, which has been mentioned as an advantage of the phonetic transformation class tree definition.

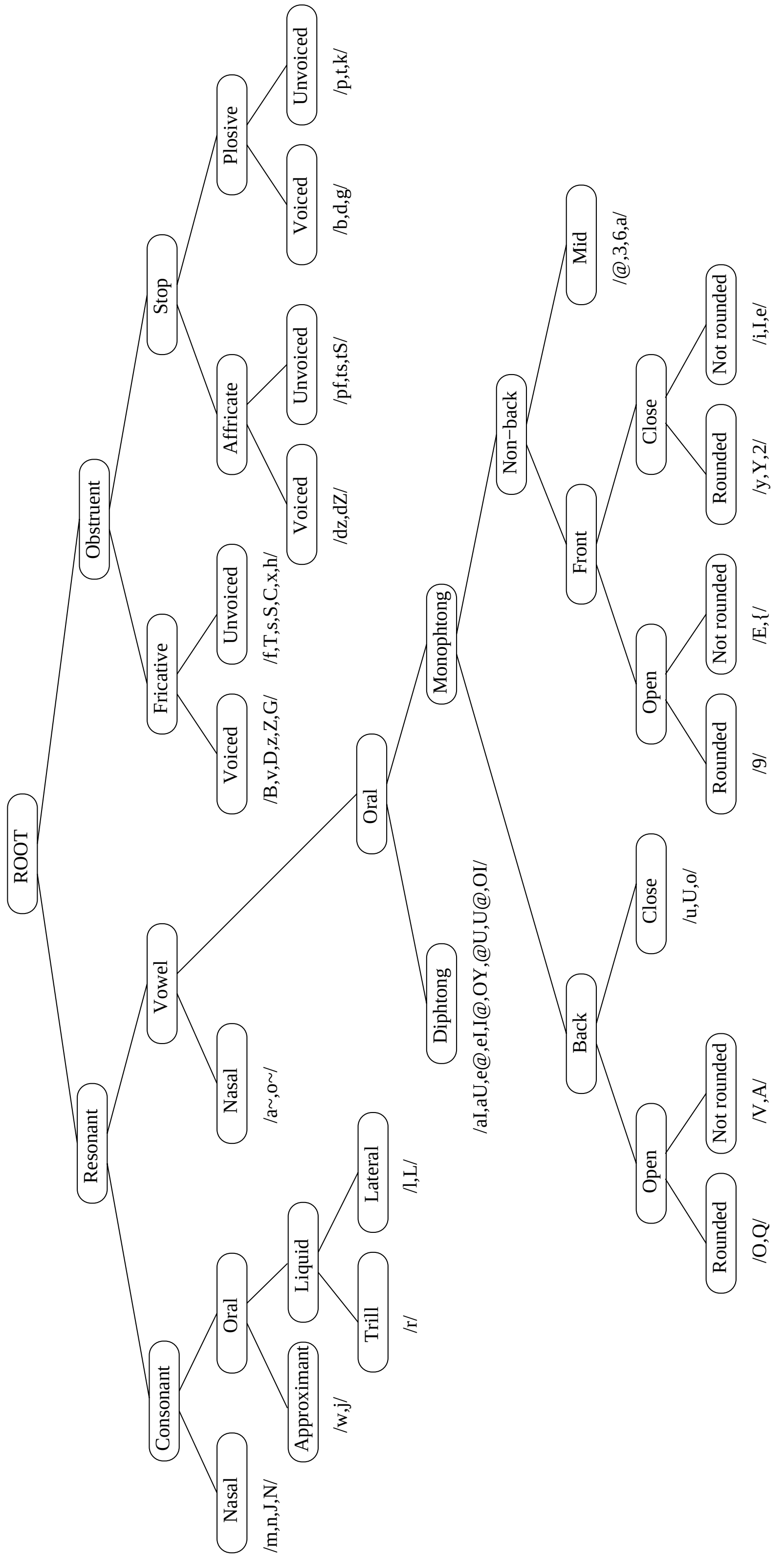

Figure A.2: A phonetic transformation class according to [Raimo and Savela 2001]. The tree contains the phonemes of the five languages: English, Finnish, German, Italian and Spanish. The phonemes are shown as SAMPA symbols [SAM]