Phonetische Entscheidungsbäume für die automatische Spracherkennung mit großem Vokabular

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der Rheinisch-Westfälischen Technischen Hochschule Aachen zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Diplom–Informatiker Klaus Beulen

aus

Rheydt

Berichter: Universitätsprofessor Dr.–Ing. Hermann Ney
Universitätsprofessor Dr.–Ing. Günther Ruske

Tag der mündlichen Prüfung: 30. Juli 1999

D 82 (Diss. RWTH Aachen)

Inhaltsverzeichnis

Kurzfassung

Um optimale Ergebnisse zu erhalten, verwenden Spracherkennungssysteme für großen Wortschatz zur Modellierung der gesprochenen Sprache in der Regel mehrere Millionen Parameter, die mit einer begrenzten Menge von Trainingsdaten geschätzt werden müssen. Der dabei auftretende Schätzfehler hängt sowohl von der Anzahl der Parameter des Systems als auch von der Menge der vorhandenen Trainingsdaten ab. Da die Trainingsdatenmenge nicht ohne weiteres vergrößert werden kann, wurden Verfahren entwickelt, die die Anzahl der Parameter des Systems verringern. Ein solches Verfahren ist das sog. State-Tying, bei dem die Parameter verschiedener akustischer Modelle des Systems aufgrund von akustischen Ähnlichkeiten zusammengefaßt werden. Die phonetische Darstellung des Vokabulars wird dabei von phonetischen Entscheidungsbäumen auf die verallgemeinerten akustischen Modelle abgebildet. Zu diesen Entscheidungsbäumen werden verschiedene Erweiterungsverfahren vorgestellt und bewertet.
Ein Vorteil der phonetischen Entscheidungsbäume ist deren Verallgemeinerungsfähigkeit, d.h. jeder möglichen Phonemfolge können passende akustische Modelle zugeordnet werden. Dies spielt vor allem eine Rolle bei der sog. Wortgrenzenmodellierung. Die Wortgrenzenmodellierung erfaßt die Koartikulationseffekte, die bei fließender Sprache an Wortgrenzen auftreten, d.h. die Ausprägung des Anfangslautes eines Wortes wird vom Endlaut des vorhergehenden Wortes beeinflußt und umgekehrt. Im Rahmen dieser Arbeit wurden Algorithmen zum Training und zur Erkennung mit Wortgrenzenmodellierung implementiert und evaluiert.
Ein Problem beim Einsatz von phonetischen Entscheidungsbäumen ist, daß die verwendeten Entscheidungsregeln phonetisches Vorwissen voraussetzen. Konkret benötigt man eine Menge von phonetischen Fragen von der Form “Ist der rechte Kontext ein Vokal?”. Diese Fragenliste mußte bisher durch einen Experten von Hand generiert werden. In dieser Arbeit wurde ein automatischer Algorithmus zur Erzeugung einer solchen Fragenliste entwickelt und an verschiedenen Korpora getestet. Wie die Ergebnisse zeigen, sind diese automatisch generierten Fragenlisten denen durch Experten erzeugten ebenbürtig.

Danksagung

Zunächst einmal möchte ich Prof. Dr.-Ing. H. Ney danken, der mir die Möglichkeit gegeben hat, im Rahmen meiner Tätigkeit als wissenschaftlicher Angestellter diese Dissertation anzufertigen. Seine fachliche Kompetenz und Unterstützung haben mich in die Lage versetzt, mich auf die wesentlichen Fragestellungen der von mir bearbeiteten Themen zu konzentrieren. Weiterhin möchte ich mich bei meinen Kollegen Lutz Welling, Stefan Ortmanns und Sven Martin bedanken, die mir in jeder Phase mit Rat und Tat zur Seite gestanden haben. Das von ihnen implementierte Spracherkennungssystem bildet die Basis für die von mir untersuchten Verfahren, die Kompaktheit und gute Handhabbarkeit dieses Systems habe ich dabei schätzen gelernt. Den Diplomanden Michael Kramer, Elmar Bransch, Johannes Overmann und Christian Elting danke ich für die gute Kooperation und die geleistete Arbeit, die in diese Dissertation eingeflossen sind. Ich danke außerdem allen Mitarbeitern des Lehrstuhls für Informatik VI, denen die angenehme und produktive Atmosphäre am Lehrstuhl zu verdanken ist.
Herrn Prof. Dr. G. Ruske möchte ich für die Übernahme des Koreferats ebenfalls herzlich danken.
Ganz besonders möchte ich meiner Familie danken, durch deren Unterstützung ich die Zeit gefunden habe, diese Arbeit anzufertigen.

Kapitel 1

Einführung

Die vorliegende Arbeit behandelt Verfahren, die im Bereich der automatischen Spracherkennung eingesetzt werden. Diese hat gerade in den letzten Jahren durch den technischen Fortschritt im Bereich der Computer-Hardware und die Entwicklung verbesserter Algorithmen den Weg aus den Forschungslabors in die praktische Anwendung gefunden. Beispielhaft seien hier Diktiersysteme und Telefonauskunftsysteme genannt:

- Heutige, auf Spracherkennung basierende Diktiersysteme sind in der Lage, kontinuierlich gesprochene Sprache mit akzeptabler Genauigkeit in geschriebenen Text umzusetzen. Die Vokabularien von 50 000 und mehr Wörtern können vom Benutzer auf einfache Weise um eigene Wörter ergänzt werden. Da auch eventuelle Korrekturen mit der Stimme durchgeführt werden können, ist es auch Ungeübten möglich, Texte auf dem Computer in akzeptabler Zeit zu verfassen [Dragon Web Site] [IBM Web Site].

- Automatische Telefonauskunftsysteme gehören zu der Gruppe der Dialogsysteme. Diese können eine Anfrage in natürlicher Sprache erkennen, die vom Benutzer benötigte Art der Auskunft ermitteln, die entsprechende Information aus einer Datenbank extrahieren, eine passende Antwort generieren und diese per Sprachsynthese über das Telefon ausgeben [Aust 98]. Ein typischer Dialog könnte so aussehen:

  System: “ Von wo nach wo möchten sie fahren? ”
  Benutzer: “ Von Hamburg nach München. ”
  System: “ Wann möchten sie fahren? ”
  Benutzer: “ Sonntagnachmittag. ”
  System: “ Es existieren folgende Verbindungen: Um ... ”

Derartige Systeme benötigen neben leistungsfähigen Suchalgorithmen, die die Echtzeitfähigkeit gewährleisten, vor allem auch exakte statistische Modelle der Sprache, die für die Erkennungsgenauigkeit eines Spracherkennungssystems verantwortlich sind. Diese statistischen Modelle sind das Sprachmodell und das akustische Modell. Während das Sprachmodell die Wahrscheinlichkeit einer Wortfolge beschreibt, modelliert das akustische Modell die akustische Realisierung eines Lautes.

Für eine optimale Schätzung der Parameter dieser Modelle ist es notwendig, die Anzahl der Modellparameter gegenüber der Anzahl der Trainingsdaten zu balancieren. Dies ist insbesondere dann wichtig, wenn das Spracherkennungssystem *kontextabhängige* Phonemmodelle verwendet, d.h. das akustische Modell berücksichtigt, daß die akustische Realisierung eines Lautes von den umgebenden Lauten abhängt. Dieser mit *Koartikulation* bezeichnete Effekt wird in heutigen Spracherkennungssystemen mit Hilfe von Phonemen im Triphonkontext, sogenannten Triphonen, modelliert. Da die Menge der möglichen Triphone kubisch mit der Zahl der Phoneme wächst, wächst auch die Zahl der zu schätzenden Parameter des Systems.
Ein weit verbreitetes Verfahren zur Reduktion der Zahl der Modellparameter eines Spracherkennungssystems im Zusammenhang mit kontextabhängiger Phonemmodellierung ist das *State-Tying*, das ein zentrales Thema dieser Arbeit sein wird. Für das State-Tying mit *phonetischen Entscheidungsbäumen* werden verschiedene Erweiterungen beschrieben und auf einer Teststichprobe evaluiert.
Ein weiteres wichtiges Problem in der kontinuierlichen Spracherkennung ist die Koartikulation an *Wortgrenzen*. In *kontinuierlich* gesprochener Sprache tritt Koartikulation auch an Wortgrenzen auf. Die Modellierung dieses Phänomens für ein Spracherkennungssystem erfordert Änderungen sowohl des Trainings als auch der Erkennung. Im Rahmen dieser Arbeit wurde das am Institut verwendete Spracherkennungssystem um diese Wortgrenzenmodellierung erweitert und systematisch optimiert.
Der Einsatz von State-Tying mit phonetischen Entscheidungsbäumen erfordert u.a. auch die Definition von phonetischen Fragen. Phonetische Fragen sind Mengen von Phonemen, die im allgemeinen Phonemklassen entsprechen. Diese Fragen werden i.A. von einem phonetischen Experten definiert werden. Dies führt zu folgenden Problemen:

- Eine sinnvolle Definition von phonetischen Fragen ist ohne Experten schwierig,
- die so definierten Fragen sind für die spezielle Anwendung des State-Tying nicht zwangsläufig optimal.

Daher wurde im Rahmen dieser Arbeit ein Verfahren entwickelt, das solche phonetischen Fragen automatisch erzeugt. Dadurch ist es z. B. auch möglich, beim Wechsel auf einen neuen Korpus innerhalb kürzester Zeit phonetische Fragen für das State-Tying zu generieren. Die Erkennungsgenauigkeit eines Systems mit automatisch generierten Fragen entspricht der eines Systems mit von einem Experten definierten Fragen bzw. übertrifft diese sogar.
Im weiteren Verlauf dieses Kapitels soll nun kurz die Architektur und die Komponenten eines Spracherkennungssystems beschrieben werden. Darauf aufbauend wird die akustische Modellierung des Basissystems beschrieben, das als Ausgangspunkt für die entwickelten bzw. implementieren Verfahren dient.

1.1 Architektur eines Spracherkennungssystems

Aufgabe eines automatischen Spracherkennungssystems ist es, zu einer Äußerung als Folge von gesprochenen Wörtern möglichst genau die Wortfolge zu erkennen. Als Kriterium für die Güte eine Erkennung wird dabei meist die Wortfehlerrate verwendet, d. h. die Anzahl

von Auslassungen, Einfügungen und Vertauschungen zwischen der erkannten Wortfolge und der gesprochenen Wortfolge. Aus Sicht der statistischen Entscheidungstheorie läßt sich das Problem folgendermaßen formulieren:

$$\left[w_1^N\right]_{opt} = \underset{w_1^N}{\operatorname{argmax}} \left\{P(w_1^N|x_1^T)\right\}.$$

Dabei bezeichnet $w_1^N = w_1...w_N$ eine Wortfolge unbekannter Länge N, $x_1^T = x_1...x_T$ eine Folge von akustischen Vektoren der Länge T und $Pr(w_1^N|x_1^T)$ die Wahrscheinlichkeit einer Wortfolge w_1^N gegeben eine Folge von akustischen Vektoren x_1^T. Dieser Ausdruck läßt sich mit Hilfe des Bayes'schen Gesetzes folgendermaßen umformulieren:

$$\begin{aligned}
\left[w_1^N\right]_{opt} &= \underset{w_1^N}{\operatorname{argmax}} \left\{\frac{p(x_1^T, w_1^N)}{Pr(x_1^T)}\right\} && (1.1)\\
&= \underset{w_1^N}{\operatorname{argmax}} \left\{\frac{P(w_1^N) \cdot p(x_1^T|w_1^N)}{p(x_1^T)}\right\} && (1.2)\\
&= \underset{w_1^N}{\operatorname{argmax}} \left\{P(w_1^N) \cdot p(x_1^T|w_1^N)\right\}. && (1.3)
\end{aligned}$$

Das heißt, die optimale Wortfolge läßt sich finden, indem man das Produkt aus a-priori-Wahrscheinlichkeit $Pr(w_1^N)$ und bedingter Wahrscheinlichkeit $Pr(x_1^T|w_1^N)$ über alle möglichen Wortfolgen maximiert. Dabei stellen die Faktoren dieses Produkts getrennte Wissensquellen dar, die bestimmte Eigenschaften der Sprache modellieren, nämlich zum einen das Sprachmodell $Pr(w_1^N)$ und zum anderen das akustische Modell $Pr(x_1^T|w_1^N)$. Das Sprachmodell ordnet jeder möglichen Wortfolge eine Wahrscheinlichkeit zu, unterscheidet somit 'typische' Wortfolgen von 'untypischen'. Das akustische Modell beschreibt, wie wahrscheinlich eine Folge von akustischen Vektoren ist, wenn eine bestimmte Wortfolge gesprochen wurde.
Folgende Anmerkungen können zu diesem Kriterium gemacht werden:

- Genaugenommen optimiert das Kriterium nicht die Wortfehlerrate, sondern die Satzfehlerrate. Diese sagt aus, mit welcher Wahrscheinlichkeit ein Spracherkennungssystem bei der Erkennung eines Satzes *mindestens* einen Fehler macht.
- In der Praxis wird die Sprachmodellwahrscheinlichkeit $P(w_1^N)$ mit einem Exponenten $\alpha > 1$ potenziert. Dies erhöht den Einfluß des Sprachmodells, was sich in der Praxis als vorteilhaft herausgestellt hat.

Aus diesen Betrachtungen ergibt sich die Architektur eines Spracherkennungssystems wie in Abb. 1.1. Das abgetastete Sprachsignal wird zunächst einer akustischen Analyse unterzogen. Diese Analyse liefert die Folge von akustischen Vektoren x_1^T. In der globalen Suche wird dann das oben genannte Produkt aus Sprachmodellwahrscheinlichkeit und akustischer Wahrscheinlichkeit über alle Wortfolgen maximiert. Die bzgl. dieser Maximierung optimale Wortfolge wird dann als erkannte Wortfolge ausgegeben. Diese Optimierung kann in der Praxis nicht über alle möglichen Wortfolgen durchgeführt werden, da schon bei einem Vokabular von 1 000 Wörtern und einer mittleren Satzlänge von zehn Wörtern

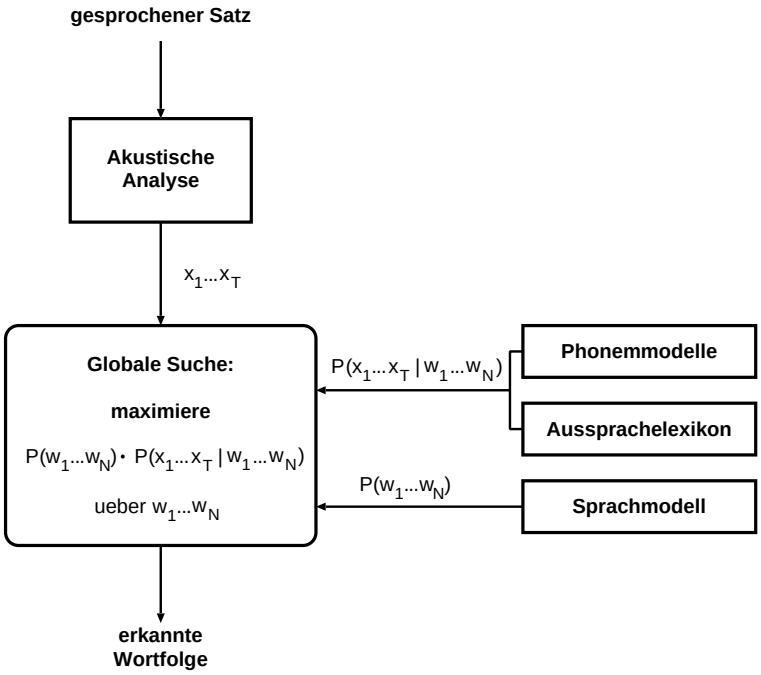


Abbildung 1.1: Architektur eines statistischen Spracherkennungsssystems

theoretisch $10^{1\,000}$ mögliche Wortfolgen zu betrachten wären. Daher muß der Suchraum zum einen geeignet strukturiert werden und zum anderen die Suche auf vielversprechende Hypothesen beschränkt werden.
In den nächsten Kapiteln soll nun genauer auf die verschiedenen Komponenten des für diese Arbeit verwendeten Spracherkennungssystems eingegangen werden.

1.2 Akustische Analyse

In der akustischen Analyse wird das abgetastete Sprachsignal in eine für die Spracherkennung möglichst günstige Form transformiert. Diese sollte für ein bestimmtes Lautereignis möglichst invariant gegenüber Einflüssen sein wie:

- verschiedene Sprecher,
- Umgebungsgeräusche,
- akustischer Kanal,
- ...

Heutzutage werden dazu meist Verfahren verwendet, die auf der schnellen Fourier-Transformation (FFT) oder einer linearen Prädiktion (*linear prediction coding*, LPC) beruhen [Press *et al.* 86] [Rabiner & Juang 93]. Die so gewonnenen Merkmale werden danach oft noch durch Verfahren wie z. B. Cepstrum dekorreliert bzw. durch Verfahren wie die lineare Diskriminanzanalyse bezüglich ihrer Diskriminierungsfähigkeit verbessert.
In der vorliegenden Arbeit wurden zwei Verfahren verwendet:

- Filterbank

  Alle 10 Millisekunden wird auf ein Segment von 25 Millisekunden der abgetasteten Sprachdaten ein Hamming-Fenster angewendet. Nachdem das Segment mit Nullwerten (*Zero-Padding*) aufgefüllt wurde, wird eine FFT mit 512 Punkten ausgeführt. Die logarithmierten Frequenzanteile werden mit der Funktion $sin(x)/x$ geglättet und im Bereich von 200 Hertz bis 6 400 Hertz an 30 Frequenzpunkten übernommen, die in etwa der Mel-Skala entsprechen. Jeder spektrale Anteil wird in Bezug auf den Mittelwert dieses Anteils im jeweiligen Satz normalisiert. Mit jedem der 30 normalisierten spektralen Anteile und dem Mittelwert ergibt sich nach diesen Schritten ein 31-dimensionaler akustischer Vektor. Dieser Vektor wird um erste und zweite Ableitungen ergänzt. Die Komponenten der Ableitungen werden jeweils paarweise zusammengefaßt, so daß sich 15 erste und 15 zweite Ableitungen ergeben. Zusammen mit dem Originalvektor ergibt sich so ein 63-komponentiger der Vektor. Schließlich werden drei zeitlich benachbarte Vektoren zusammengefaßt und mit Hilfe der linearen Diskriminanzanalyse auf 35 Komponenten reduziert [Steinbiss *et al.* 93].

- Cepstrum

  Alle 10 Millisekunden wird auf ein Segment von 25 Millisekunden der höhenangehobenen Sprachdaten ein Hamming-Fenster angewandt. Nach *Zero-Padding* wird das gefensterte Signal mit einer 1024-Punkt-FFT in den Frequenzraum transformiert. Das Betragsspektrum wird dann bezüglich einer Mel-Frequenzskala verzerrt. Die so erhaltenen spektralen Intensitäten werden mit 20 auf der Mel-Frequenzskala äquidistanten Dreiecksfiltern integriert. Die Mittenfrequenz von Filter n ist $n/2 \cdot 270.48$ Hz, die Bandbreite aller Filter ist 270.48 Hz. Die Ausgabe eines Filters ist der Logarithmus der Summe der gewichteten Beträge. Die 20 Filterbankausgänge werden dann mit einer Cepstrum-Transformation dekorreliert, die 16 Cepstrum-Koeffizienten liefert. Dieser 16-dimensionale Vektor wird dann noch mit 16 ersten und einer zweiten Ableitung ergänzt, die über ein Fenster von 5 zeitlich benachbarten Vektoren mit linearer Regression berechnet werden. Um Veränderungen des Übertragungskanals zu berücksichtigen, wird eine Normalisierung auf den Mittelwert durchgeführt. Abschließend werden drei zeitlich benachbarte Vektoren zusammengefaßt und mit Hilfe der linearen Diskriminanzanalyse auf 33 Komponenten reduziert [Welling *et al.* 97].

1.3 Akustische Modellierung

Das akustische Modell dient dazu, zu einer gegebenen Wortfolge w_1^N die Wahrscheinlichkeit einer Folge von akustischen Ereignissen x_1^T zu bestimmen. Diese akustische Wahrscheinlichkeit wird dann zusammen mit der Sprachmodellwahrscheinlichkeit dazu verwendet, in der globalen Suche die optimale Wortfolge zu bestimmen. Die Parameter des akustischen Modells werden dazu in einer vorausgehenden Trainingsphase anhand von Beispieläußerungen bestimmt. Daraus ergeben sich folgende Forderungen an das akustische Modell [Rabiner & Juang 93]:

- exakte Modellierung

Das akustische Modell soll die Wahrscheinlichkeit einer Vektorfolge zu einer gegebenen Wortfolge möglichst genau modellieren. Daher muß es sowohl die spektrale Variabilität des Sprachsignals aufgrund verschiedener Sprecher, Betonung etc. als auch die zeitliche Variabilität des Sprachsignals aufgrund verschiedener Sprechgeschwindigkeiten berücksichtigen.

- effiziente Trainierbarkeit

  Das akustische Modell wird im allgemeinen aus einer großen Menge von Beispielsätzen generiert, die typischerweise aus 10 bis 100 Stunden Sprache besteht. Aufgrund der Komplexität der Modelle muß das Training iterativ durchgeführt werden, bis eine Konvergenz der Parameter eintritt. Daher sollten die akustischen Modelle möglichst effizient trainierbar sein, um die Dauer eines solchen Trainings in vernünftigen Grenzen zu halten (ca. 1-2 Wochen).

- Beschreibung möglicher Wortfolgen

  Da die Wortfolge, der das akustische Modell eine Wahrscheinlichkeit zuordnen soll, beliebig sein kann, muß das akustische Modell so strukturiert sein, daß es alle möglichen Wortfolgen modellieren kann. Da es nicht möglich ist, alle Wortfolgen getrennt zu modellieren, müssen daher Modelle für Phrasen, Worte oder Wortuntereinheiten verwendet werden.

- schnelle Auswertung

  In der globalen Suche, in die das akustische Modell eingeht, muß im allgemeinen eine große Anzahl von Hypothesen ausgewertet werden. Damit diese Auswertung möglichst schnell abläuft, muß die akustische Wahrscheinlichkeit effizient bestimmt werden können.

Um diese Forderungen erfüllen zu können, werden in heutigen Spracherkennungssystemen meist sogenannte *Hidden-Markov-Modelle* (HMM) verwendet. Mit HMM ist es möglich, sowohl die spektrale als auch die zeitliche Variabilität des Sprachsignals zu beschreiben. Weiterhin lassen sich HMM effizient trainieren, und die Wahrscheinlichkeit einer gegebenen Vektorfolge läßt sich mit Hilfe der sogenannten *dynamischen Programmierung* für ein HMM schnell bestimmen.
Im weiteren soll nun zunächst das Konzept des HMM für die Spracherkennung genauer erläutert werden, wobei auch auf Trainingsverfahren eingegangen wird. Weiterhin soll erklärt werden, welche Wortuntereinheiten für die Erkennung von kontinuierlicher Sprache mit großem Wortschatz geeignet sind.

1.3.1 Hidden-Markov-Modelle

Im Gegensatz zu klassischen Mustererkennung, bei der *ein* Vektor fester Länge klassifiziert werden soll, muß in der Spracherkennung eine *Folge* von (akustischen) Vektoren einem Wort oder einer Wortfolge zugeordnet werden. Diese Folge von Vektoren beschreibt eine Äußerung anhand der spektralen Intensitäten über den Zeitraum der Äußerung (siehe auch Abschnitt 1.2). Dabei kann sowohl die Form des Spektrums für einen bestimmten

Zeitpunkt t (z. B. aufgrund veränderter Artikulation) als auch die Abfolge der Vektoren (aufgrund von Schwankungen der Sprechgeschwindigkeit) variieren. Ein Modell, das diese Variabilität im Sprachproduktionsprozeß gut beschreibt, ist das sogenannte *Hidden-Markov-Modell*, kurz HMM. Ein HMM ist eine spezielle Form eines zeitdiskreten Markov-Prozesses, bei dem der Zustand, indem sich der Prozeß befindet, nicht beobachtet werden kann (daher die Bezeichnung "Hidden"). Stattdessen emittiert der Markov-Prozeß bei jedem Zustandsübergang, der mit einer für den jeweiligen Übergang spezifischen *Transitionswahrscheinlichkeit* stattfindet, eine Beobachtung gemäß einer dem Zustand zugeordneten Wahrscheinlichkeitsverteilung, *Emissionsverteilung* genannt. Die so erzeugte Folge von Beobachtungen kann von außen wahrgenommen werden. Somit kann das HMM als Modell für die Sprachproduktion verwendet werden, indem die Folge von akustischen Vektoren als Folge von emittierten Beobachtungen und der zugrundeliegende Sprachproduktionsprozeß als Markov-Prozeß modelliert werden.
In der Spracherkennung wird für die Markov-Kette i.A. die Struktur "links-rechts" verwendet, meist in einer der Ausprägungen "Bakis" oder "linear". Hierbei sind nur Übergänge in einen Zustand gleicher oder höherer Numerierung erlaubt (siehe Abbildung 1.2).

- links-rechts

  Die allgemeinste Form läßt von jedem Zustand aus Übergänge zu, die in einen Zustand mit gleicher oder höherer Numerierung führen.

- Bakis

  Das Bakis-Modell verwendet für jeden Zustand drei Übergänge, nämlich

  - *Loop* (Übergang in denselben Zustand),
  - *Next* (Übergang in den nächsthöheren Zustand),
  - *Skip* (Überspringen des nächsthöheren Zustandes).

- linear

  Hier sind nur Übergänge in denselben oder den nächsthöheren Zustand erlaubt.

Die Transitionswahrscheinlichkeiten a_{ij} sind diskrete Verteilungen, die die Wahrscheinlichkeit, daß ein HMM von einem Zustand i in einen Zustand j übergeht, beschreiben. Als Emissionsverteilungen b_j sind diskrete, semikontinuierliche und kontinuierliche Verteilungen gebräuchlich. Da die HMM des für diese Arbeit verwendeten Spracherkennungssystems kontinuierliche Mischverteilungen verwenden, soll im weiteren nur auf diese Form der Modellierung eingegangen werden.
Die HMM, die in dieser Arbeit verwendet werden, besitzen die "Bakis"-Topologie mit sechs Zuständen pro Phonemmodell (siehe auch Kapitel 1.3.5), wobei das HMM in drei Segmente eingeteilt wird, d.h. die Zustandspaare $\{1, 2\}$, $\{3, 4\}$ und $\{5, 6\}$ dieselbe Emissionsverteilung verwenden. Die Transitionswahrscheinlichkeiten werden über alle HMM des Systems gepoolt, d.h. es gibt jeweils eine Wahrscheinlichkeit für *Loop, Next* und *Skip*.

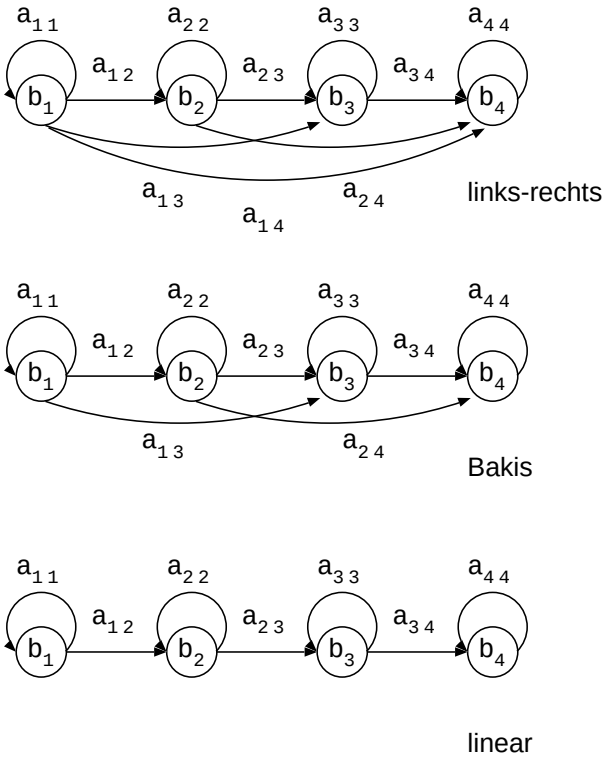


Abbildung 1.2: Typische HMM-Strukturen

1.3.2 Emissionsverteilungen

Eine kontinuierliche Verteilung beschreibt die Verteilung einer Zufallsvariable über einem Wahrscheinlichkeitsraum mit Dimension D und Wertebereich $\mathbb{R}^D$. Da ein bestimmtes Ereignis in einem solchen Wahrscheinlichkeitsraum die Wahrscheinlichkeit 0 hat, wird eine kontinuierliche Verteilung durch ihre Wahrscheinlichkeitsdichtefunktion $p(x)$ beschrieben. Für ein HMM ist diese Wahrscheinlichkeitsdichte abhängig vom Wort w und vom Zustandsindex s des Wortes w, das durch dieses HMM modelliert wird, d.h. die Wahrscheinlichkeitsdichtefunktion wird als bedingte Wahrscheinlichkeitsdichtefunktion $p(x|s,w)$ geschrieben. Mit dieser Funktion kann jedem *Bereich* des Definitionsbereichs von $p(x|s,w)$ eine Wahrscheinlichkeit zugeordnet werden, indem $p(x|s,w)$ über diesem Bereich integriert wird. Für das Integral von $p(x|s,w)$ über dem gesamten Definitionsbereich $\mathbb{R}^D$ gilt:

$$\int_{\mathbb{R}^D} p(x|s,w)dx = 1$$

Beispiele für in der Spracherkennung gebräuchliche Wahrscheinlichkeitsdichtefunktionen sind die Gauß-Verteilung

$$p(x|s,w) = \frac{1}{\sqrt{2\pi}\prod_{d=1}^{D}\sigma_{swd}} \exp\left(-\frac{1}{2}\sum_{d=1}^{D}\frac{(x_d-\mu_{swd})^2}{\sigma_{swd}^2}\right)$$

und die Laplace-Verteilung (diese Arbeit)

$$p(x|s,w) = \frac{1}{2\prod_{d=1}^{D} v_{swd}} \exp\left(-\sum_{d=1}^{D}\left|\frac{x_d - \mu_{swd}}{v_{swd}}\right|\right)$$

Der Parameter μ_{swd} ist dabei der Mittelwert (Gauß) bzw. Median (Laplace) für eine Komponente d, der Parameter σ_{swd} die Wurzel der Varianz (Gauß), v_{swd} die Absolutabweichung (Laplace) für Komponente d. Bei der Gauß-Verteilung wurde außerdem die in der Spracherkennung übliche Annahme gemacht, daß die einzelnen Merkmale unkorreliert sind, was bei entsprechender Vorverarbeitung (z. B. Cepstrum) näherungsweise zutrifft. Sind die zu modellierenden Verteilungen annähernd Gauß- bzw. Laplace-verteilt und die einzelnen Merkmale unkorrelliert, lassen sie sich mit diesen Verteilungsdichtefunktionen relativ genau beschreiben. Diese Annahme ist i.A. aber nicht zutreffend, da die akustische Realisierung eines Lautes von Sprecher zu Sprecher und auch von Äußerung zu Äußerung schwankt. Daher benötigt man für eine hinreichend genaue Modellierung des Sprachsignals eine allgemeinere Form der Verteilung. Eine Möglichkeit der Definition einer solchen Verteilung ist die Mischverteilung:

$$p(x|s,w) = \sum_{l=1}^{L(s,w)} p(x,l|s,w) \tag{1.4}$$

$$p(x,l|s,w) = p(l|s,w) \cdot p(x|s,w,l) \tag{1.5}$$

Eine Mischverteilung ist eine Linearkombination von $L(s,w)$ einfacheren Verteilungen, meist unimodalen Verteilungen wie Gauß- oder Laplace-Verteilungen. Die Zahl der Verteilungen ist dabei abhängig von w und s. $p(x|s,w)$ ist die multimodale Verteilung für Zustand s des Wortes w, $p(x|s,w,l)$ die l-te unimodale Verteilung für Zustand s des Wortes w. $p(l|s,w)$ ist das sogenannte *mixture weight*, das die Gewichtung der Einzelverteilung l innerhalb der Linearkombination 1.5 festlegt. Dieses *mixture weight* ist normiert mit

$$\sum_{l} p(l|s,w) = 1 \ .$$

In der Praxis, d.h. bei der Implementierung von Spracherkennungsalgorithmen, hat es sich als sinnvoll herausgestellt, die Wahrscheinlichkeiten in negativer logarithmierter Form darzustellen. Damit ergibt sich

$$-\log p(x|s,w) = -\log \sum_{l=1}^{L(s,w)} p(l|s,w) \cdot p(x|s,w,l) \ .$$

Unter der Annahme, daß wegen des exponentiellen Abfalls von Laplace-Verteilungen meist eine Verteilung $\hat{l}$ innerhalb der Mischverteilung existiert, die die Summe dominiert, kann durch die sogenannte *Maximum-Approximation* dieser Ausdruck vereinfacht werden:

$$p(x|s,w) = \sum_{l=1}^{L(s,w)} p(x,l|s,w) \tag{1.6}$$

$$= \sum_{l=1}^{L(s,w)} p(l|s,w) \cdot p(x|s,w,l) \tag{1.7}$$

$$\approx \max_{l} \{p(l|s,w) \cdot p(x|s,w,l)\} \tag{1.8}$$

Damit ergibt sich die negative logarithmierte Emissionswahrscheinlichkeit einer Mischverteilung mit Laplace-Einzelverteilungen zu

$$-\log p(x|s,w) = \min_{l} \left\{ \frac{1}{2} \sum_{d=1}^{D} \left| \frac{x_d - \mu_{swld}}{v_{swd}} \right| - \log p(l|s,w) + \frac{1}{2} \sum_{d=1}^{D} \log v_{swd} \right\} .$$

In dieser Formel sind die Absolutabweichungen v_{swld} bzgl. einer Mischverteilung gepoolt, d.h. alle v_{swld} mit gleichem s, w und d besitzen denselben Wert, hier mit v_{swd} bezeichnet. Diese Annahme ist für Mischverteilungen mit einer hohen Komponentenzahl $L(s,w)$ ($L(s,w) >> 10$) meist recht gut erfüllt, d.h. die Fehlerraten sind im Vergleich zu einer Modellierung ohne Pooling der Standardabweichungen bei hohen $L(s,w)$ mit Pooling gleich oder sogar besser aufgrund der robusteren Schätzung der Standardabweichungen.
Aufgrund des Poolings der Standardabweichungen erfordert diese Formel keine wiederholten komplexen arithmetischen Berechnungen, da der Beobachtungsvektor x nicht in den Logarithmen auftaucht und diese daher konstant sind. Daher kann die Auswertung der Formel relativ effizient erfolgen. Trotzdem verursacht diese Auswertung noch ca. 70% des gesamten Rechenaufwandes bei der Spracherkennung mit großem Vokabular. Durch geeignete Verfahren [Ortmanns 98] kann dieser Aufwand auf ca. 35% gesenkt werden, ohne daß die Fehlerrate des Systems signifikant ansteigt. Diese Verfahren wurden in dieser Arbeit allerdings nicht eingesetzt.

1.3.3 Bestimmung der Wahrscheinlichkeit einer Folge von Beobachtungsvektoren

Die im vorigen Abschnitt dargestellte Modellierung von Emissionsverteilungen mit Laplace'schen Mischverteilungen soll nun im Kontext der HMM dazu verwendet werden, die Wahrscheinlichkeit $p(x_1^T|w)$ einer Folge von Beobachtungsvektoren x_1^T gegeben ein Wort w zu bestimmen. Diese kann geschrieben werden als

$$p(x_1^T|w) = \sum_{[s_1^T]} p(x_1^T, s_1^T|w) \tag{1.9}$$

$$p(x_1^T, s_1^T|w) = \prod_{t=1}^{T} p(x_t, s_t|x_1^{t-1}, s_1^{t-1}, w) \tag{1.10}$$

d.h. die Wahrscheinlichkeit $p(x_1^T|w)$ wird berechnet, indem die Wahrscheinlichkeit $p(x_1^T, s_1^T|w)$ über alle möglichen Zustandsfolgen s_1^T mit $s_t \in \{1, \ldots, S\}$ summiert wird.
Diese Summe wird nun vereinfacht, indem folgende Modellannahmen getroffen werden:

- Die Abhängigkeit der Wahrscheinlichkeit besteht nur über die abstrakten Zustände $s = 1, \ldots, S(w)$ des Wortes w:

$$p(x_t, s_t | x_1^{t-1}, s_1^{t-1}, w) = p(x_t, s_t | s_1^{t-1}, w)$$

- Die Abhängigkeit der Wahrscheinlichkeit bezieht sich nur auf den Vorgängerzustand s_{t-1}:

$$\begin{aligned} p(x_t, s_t | s_1^{t-1}, w) &= p(x_t, s_t | s_{t-1}, w) && (1.11) \\ &= p(s_t | s_{t-1}, w) \cdot p(x_t | s_{t-1}, s_t, w) && (1.12) \end{aligned}$$

  $p(s_t|s_{t-1}, w)$ ist dabei die Transitionswahrscheinlichkeit, $p(x_t|s_{t-1}, s_t, w)$ die Emissionswahrscheinlichkeit für einen Zustandsübergang (s_{t-1}, s_t), d.h. hier ist die Emissionswahrscheinlichkeit noch von einem Zustandspaar abhängig.

Damit kann der Ausdruck für die Emissionswahrscheinlichkeit der Vektorfolge x_1^T geschrieben werden als

$$\begin{aligned} p(x_1^T | w) &= \sum_{[s_1^T]} \prod_{t=1}^{T} [p(s_t | s_{t-1}, w) \cdot p(x_t | s_{t-1}, s_t, w)] && (1.13) \\ &= \sum_{[s_1^T]} \prod_{t=1}^{T} p(x_t, s_t | w) \ . && (1.14) \end{aligned}$$

Diese Summe muß nun für alle Zustandsfolgen $[s_1^T]$ ausgewertet werden. Da der naive Ansatz, alle möglichen Kombinationen von Zuständen zu bilden und über diese zu summieren, exponentielle Komplexität bzgl. T besitzt, wurden geschicktere Verfahren zur Auswertung dieser Summe entwickelt, z.B. der *Forward-Algorithmus* [Rabiner & Juang 93]. Eine weitere Vereinfachung, die Viterbi-Approximation, bei der die Summe über alle Zustandsfolgen durch die Maximumbildung ersetzt wird, führt zu

$$p(x_1^T | w) = \max_{[s_1^T]} \{p(x_t, s_t | w)\} \ .$$

Durch Logarithmierung und Negation erhält man die Formel

$$-\log p(x_1^T | w) = \min_{[s_1^T]} \{-\log p(x_t, s_t | w)\} \ .$$

Die Modellierung der HMM im RWTH-System sieht weiterhin vor, daß

- die Zustände $s = 1, \ldots, S$ linear angeordnet sind,
- die Transitionswahrscheinlichkeiten der HMM lediglich von der "Sprungweite", d.h. von der Differenz der Zustandsindizes abhängen

$$p(s_t | s_{t-1}, w) = \begin{cases} q(s_t - s_{t-1}) & : \quad s_t \in \{s_{t-1} + 0, s_{t-1} + 1, s_{t-1} + 2\} \\ 0 & : \quad sonst \end{cases} \ ,$$

- und die Abhängigkeit der Emissionswahrscheinlichkeit nur vom aktuellen Zustand s_t besteht

$$p(x_t|s_{t-1}, s_t, w) = p(x_t|s_t, w) \ .$$

Damit ergibt sich für die Verbundwahrscheinlichkeit $p(x_t, s_t|w)$ die logarithmierte Form

$$-\log p(x_t, s_t|w) = -\log p(x_t|s_t, w) - \log q(s_t - s_{t-1})$$

Zur Vereinfachung definiert man zunächst

$$d(x_t; s, w) = -\log p(x_t|s_t, w) \tag{1.15}$$
$$\mathcal{T}(s_t - s_{t-1}) = -\log q(s_t - s_{t-1}) \tag{1.16}$$

Damit lautet das zu lösende Optimierungsproblem

$$-\log p(x_1^T|w) = \min_{[s_1^T]} \sum_{t=1}^{T} \{d(x_t; s, w) + \mathcal{T}(s_t - s_{t-1})\} \ .$$

Zur Lösung dieses Problems kann das Verfahren der *Dynamischen Programmierung* verwendet werden [Bellman 57]. Dazu wird folgende Hilfsgröße eingeführt:

$$D(t, s; w) = \min_{[s_1^T]} \left\{ \sum_{\tau=1}^{t} \{d(x_\tau; s, w) + \mathcal{T}(s_\tau - s_{\tau-1}) : s_t = s \right\}$$

Die Rekursionsformel für die dynamische Programmierung lautet dann

$$D(t, s; w) = \min_{s_{t-1}} \{D(t-1, s_{t-1}; w) + d(x_t; s, w) + \mathcal{T}(s_t - s_{t-1}\} \ .$$

Der Wert $D(T, S; w)$ entspricht dann der gesuchten (negativen logarithmierten) Wahrscheinlichkeit der Vektorfolge x_1^T gegeben das Wort w. Dieser Wert kann berechnet werden, indem beginnend mit dem Zeitpunkt $t = 1$ und dem Zustand $s = 1$ die Werte $D(t, s; w)$ zeitsynchron berechnet werden. Dieses Verfahren wird auch *Zeitanpassung* genannt. Will man die optimale Zustandsfolge $\hat{s}_1^T$ zu $D(T, S; w)$ ermittelt, kann dies mit sogenannten *Backpointern* erfolgen [Ney 84] (siehe Abbildung 1.3).

1.3.4 Training

Die im vorherigen Abschnitt beschriebenen Methoden zur Modellierung der menschlichen Sprache verwenden Modelle, deren Parameter auf geeignete Weise bestimmt werden müssen. Dies geschieht bei statistischen Spracherkennungssystemen in einer Trainingsphase, in der diese Parameter anhand von Beispielen gelernt werden. Diese Beispiele bestehen im allgemeinen aus einer Menge von Sätzen, die sowohl als akustisches Signal als auch in transkribierter Form, d.h. als Folge von Wortindizes des Vokabulars, vorliegen. Ein Satz aus der Menge der Sätze in akustischer Form soll im folgenden mit X_i, ein Satz aus der Menge der Sätze in transkribierter Form mit W_i bezeichnet werden. Die Parameter Θ

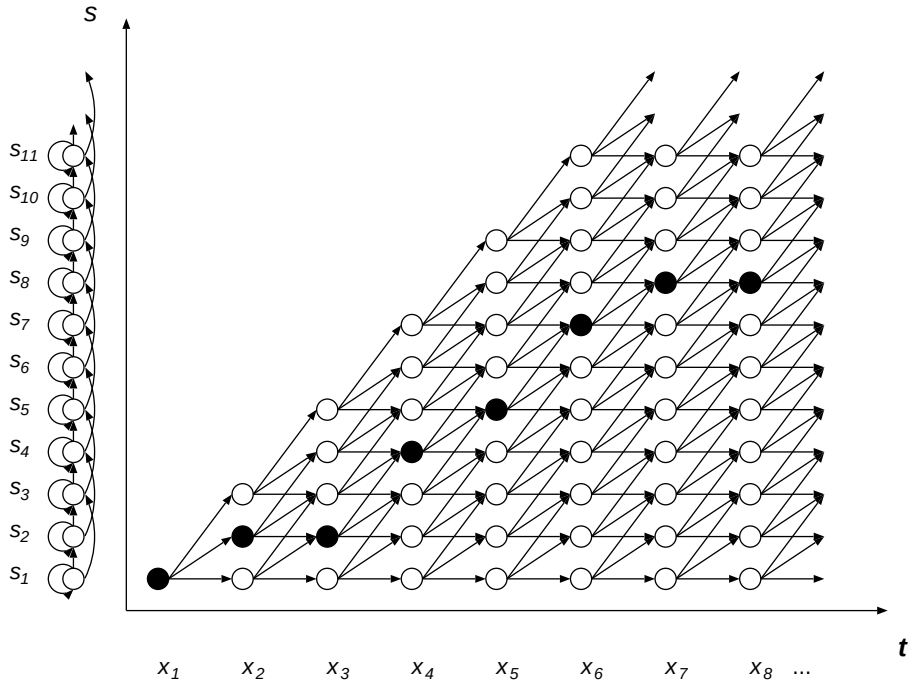


Abbildung 1.3: Viterbi-Training: Wahrscheinlichster Pfad für eine Vektorfolge x_1^T und ein HMM mit Zuständen $s_i \in \{1, 2, \ldots, S\}$

des Spracherkennungssystems sollen nun so trainiert werden, daß das Produkt über die Wahrscheinlichkeiten $p(X_i|W_i, \Theta)$ der X_i gegeben die W_i maximiert wird:

$$\hat{\Theta} = \underset{\Theta}{\operatorname{argmax}} \prod_{i=1}^{N} p(X_i|W_i, \Theta)$$

Für dieses sogenannte *Maximum-Likelihood*-Trainingskriterium kann gezeigt werden, daß für $N \to \infty$ die Fehlerrate eines auf der Bayesschen Entscheidungsregel basierenden Klassifikators für das jeweilige Modell minimiert wird. Durch Ableiten des Produktes nach den verschiedenen Parametern können Schätzformeln für die Parameter Θ hergeleitet werden. Allerdings sind diese Formeln für HMM nicht geschlossen lösbar, daher werden iterative Verfahren verwendet. Die im RWTH-System verwendete Variante ist das Viterbi-Training, das mit der im vorigen Abschnitt beschriebenen Viterbi-Approximation bei der Zeitanpassung arbeitet. Die Vorgehensweise beim Viterbi-Training besteht dann aus folgenden Schritten:

1. **Aufbau eines Satzmodells**. Für jeden gesprochenen Satz des Trainingskorpus wird entsprechend der Wortfolge ein HMM erzeugt, indem die HMM der einzelnen Worte hintereinandergehängt werden. Die Zustandsübergänge an den Wortgrenzen werden dabei äquivalent zum Wortinnern behandelt. Zwischen die Wörter wird jeweils ein Pausezustand eingefügt, der optionale Zwischenwortpausen modelliert. Falls keine Zwischenwortpause vorliegt, kann der Zeitanpassungsalgorithmus die Pause durch die Transition mit Länge 2 auslassen.

2. **Zeitanpassung**. Für jeden Satz des Trainingskorpus wird dann eine Zeitanpassung der Vektorfolge des Satzes x_1^T mit dem erzeugten Satz-HMM berechnet. Diese Zeitanpassung ordnet jeden Vektor x_t genau einem Zustand s_t des Satzmodells zu. Dieser Vektor geht dann in die nachfolgende Schätzung der Parameter der diesem Zustand zugeordneten Emissionsverteilung $p(x_t|s_t, w)$ ein.

   Da in der ersten Trainingsiteration noch keine Schätzungen der Modellparameter vorliegen, kann zu diesem Zeitpunkt auch keine Zeitanpassung durchgeführt werden. Daher kann alternativ

   - eine Zuordnung eines vorangegangenen Trainings verwendet werden oder
   - mit dem Verfahren der *linearen Segmentierung* [Bridle & Sedgewick 77] eine grobe Zuordnung gefunden werden.

     Die lineare Segmentierung nimmt dabei an, daß der gesprochene Satz aus einer Folge "Pause, Sprache, Pause" besteht. Aufgrund dieser Annahme werden über die Signalenergie der akustischen Vektorfolge x_1^T Start- und Endzeitpunkt des Sprachsegments berechnet und dann die Vektorfolge linear auf die Zustandsfolge abgebildet.

3. **Parameterschätzung**. Aufgrund der Zuordnung der akustischen Vektoren aus der Zeitanpassung werden dann die Parameter der Emissionsverteilungen, d.h. Median (der im RWTH-System durch den Mittelwert approximiert wird), Standardabweichung und Gewichte der Einzelverteilungen geschätzt. D.h. die verwendeten Parameter sind empirische Schätzungen der tatsächlichen Parameter, die nicht bekannt sind. Spezifisch für das RWTH-System sind weiterhin folgende Eigenschaften:

   - Statt eines Standardabweichungsvektors pro Verteilung wird nur ein globaler Vektor geschätzt.
   - Die Transitionswahrscheinlichkeiten sind ebenfalls für alle HMM gleich (siehe Abschnitt 1.3.3) und werden nicht aus den Trainingsdaten berechnet, sondern durch manuelle Optimierung bestimmt.

   Weiterhin wird eine Mischverteilungskomponente gelöscht, wenn die Anzahl der ihr zugeordneten Trainingsvektoren kleiner als ein Schwellwert wird. Dieser beträgt typischerweise 5-10.

4. **Splitten der Verteilungen**. Im allgemeinen startet für das RWTH-System ein Training mit nur einer Komponente pro Mischverteilung. Die Parameter des Systems werden dann mit den oben beschriebenen Schritten bis zur (approximativen) Konvergenz trainiert. Danach werden *die* Einzelverteilungen dupliziert, deren mittlere negative Log-Likelihood (Log-Likelihood durch Anzahl der Trainingsvektoren) kleiner als die mittlere negative Log-Likelihood über alle Verteilungen ist. Die Duplizierung wird durchgeführt, indem die Verteilung durch zwei neue Verteilungen ersetzt wird, deren Mittelwerte um einen Offset $+\epsilon/-\epsilon$ gegenüber dem alten Mittelwert verschoben sind. Falls die Anzahl der dieser Verteilung zugeordneten Trainingsvektoren kleiner als ein Schwellwert ist, wird diese Aufteilung nicht durchgeführt. Dieser Schwellwert beträgt typischerweise 10-20, sollte aber auf jeden Fall

mehr als doppelt so groß sein wie der im vorigen Punkt angesprochene Schwellwert, da sonst eine der beiden Verteilungen in der nächsten Iteration wieder gelöscht wird.

Die Schritte 2 und 3 (und optional auch 4) werden im Verlauf eines Trainings ca. 30-50 mal iteriert (abhängig von der Anzahl der Aufsplittungen), bis die Parameter der akustischen Modelle konvergiert sind.
Durch zwei zusätzliche Maßnahmen wird das Trainingverfahren des RWTH-Systems weiter beschleunigt:

- Bei der Zeitanpassung werden zu einem Zeitpunkt nicht alle $D(t, s; w)$ berechnet, sondern nur die 15-20, die in der direkten Umgebung der aktuell besten Bewertung liegen. Dadurch kann die Komplexität des Verfahrens für $S = T$ von $O(ST) \approx O(T^2)$ auf $O(T)$ gedrückt werden. Findet das Verfahren das Maximum aller $D(T, s; w)$ nicht bei S, wird die Fenstergröße erhöht und die Zeitanpassung erneut durchgeführt.

- Während einer Zeitanpassung wird die Zuordnung der Vektoren zu einem Zustand des Satz-HMM in einer Datei gespeichert. Diese kann in der nächsten Iteration ausgelesen und so die relativ aufwendige Zeitanpassung vermieden werden. Typischerweise werden die Zuordnungen einer Zeitanpassung ca. 2-3 mal verwendet, bevor eine neue Anpassung vorgenommen wird.

1.3.5 Wortuntereinheiten

In Spracherkennungssystemen für kleinen Wortschatz werden meist Wörter als Basismodelle verwendet, d. h. jedes Wort des Vokabulars wird durch ein eigenes HMM modelliert. Dieser Ansatz ist hier vorteilhaft, weil für jedes Wort des Vokabulars im allgemeinen genug Trainingsmaterial zur Verfügung steht. Dies ist bei Systemen für großen Wortschatz nicht mehr der Fall. Hier ist das vorhandene Trainingsmaterial ungleichmäßig über die Wörter des Vokabulars verteilt. Beispielsweise werden für Wörter wie "ist" oder "die" viele Äußerungen zur Verfügung stehen, während für Wörter wie "Spracherkennungssystem" nur wenig Material vorhanden sein wird. Außerdem berücksichtigt ein wortbasiertes Spracherkennungssystem keine akustischen Ähnlichkeiten zwischen Wörtern, wie sie zum Beispiel zwischen "Rasen" und "Riesen" bestehen. Daher werden für Systeme mit großem Wortschatz keine Wörter, sondern Wortuntereinheiten als Basiseinheiten verwendet. Die in Spracherkennungssystemen am häufigsten verwendete Wortuntereinheit ist das Phonem. Ein Phonem wird meist definiert als kleinste bedeutungsunterscheidende Lauteinheit, beispielsweise die Laute, die die oben genannten Wörter "Rasen" und "Riesen" trennen [Schukat-Talamazzini 95]. Je nach Sprache schwankt die Anzahl der zur Beschreibung des Vokabulars einer Sprache benötigten Phoneme zwischen 20-60, für das Deutsche wird beispielsweise eine Zahl von 48 Phonemen angegeben. Im einfachsten Fall wird nun für jedes dieser Phoneme ein HMM trainiert, wodurch man bei einer entsprechenden Menge von Trainingsmaterial sehr robuste Schätzungen für die Modelle erhält. Dies führt bei steigender Vokabulargröße schließlich dazu, daß der Spracherkenner mit Phonemmodellen den wortbasierten in der Erkennungsakkuratheit übertrifft.

1.3.5.1 Triphone

Ein derartiger phonembasierter Spracherkenner modelliert die Aussprache der Phoneme eines Wortes unabhängig vom phonetischen Kontext. In der Praxis läßt sich allerdings beobachten, daß die Aussprache eines Phonems stark vom phonetischen Kontext abhängt. Grund für diese Abhängigkeit ist die sogenannte Koartikulation. Die an der Lautbildung beteiligten Organe des Sprachtraktes wie z. B. Gaumen, Zunge und Lippen besitzen unterschiedliche Trägheit bzgl. ihrer lautbildenden Eigenschaften. Durch diese Trägheit kommt es insbesondere bei schneller Sprechweise dazu, daß ein Laut sowohl vom vorher gebildeten als auch vom danach zu bildenden Laut abhängt. Dieser Effekt läßt sich relativ einfach mit sogenannten kontextabhängigen Phonemmodellen beschreiben. Das am häufigsten verwendete Modell ist das Phonemmodell im Triphonkontext, auch Triphon genannt. Ein solches Triphon modelliert die Kontextabhängigkeit eines Phonems anhand der unmittelbaren Nachbarn. Beispielsweise würden zur Modellierung des Wortes "der" die Triphone

$$ {}_{\#}d_{eh} \quad {}_{d}e h_{r} \quad {}_{eh}r_{\#} $$

verwendet. Das Symbol "#" markiert dabei eine Wortgrenze, da hier das vorherige und das folgende Wort nicht bekannt sind. In kontinuierlicher Sprache findet Koartikulation aber auch an Wortgrenzen statt, so daß durch einen Berücksichtigung der Wortgrenzenphoneme eine weitere Verbesserung der Modellierung erreicht werden kann (siehe auch Kapitel 6). Triphone modellieren die akustische Variabilität auf Grund von Koartikulation sehr genau. Allerdings besteht für sie, ähnlich wie für Wortmodelle, das Problem der Schätzbarkeit der Parameter. Z.B. ergibt sich bei 48 Phonemen eine mögliche Zahl von über 100 000 Triphonen. Obwohl die Zahl der aufgrund von phonotaktischen Einschränkungen tatsächlich möglichen Triphone deutlich geringer ist, muß bei einem typischen Korpus immer noch mit einer erheblichen Zahl von Triphonen gerechnet werden (siehe Tab. 1.1).

Tabelle 1.1: Anzahl Triphone für verschiedene Lexika für den *Wall Street Journal* (WSJ)- und den *North American Business*-Task (NAB), 43 Phoneme + 1 Silence-Phonem.

Lexikon	Anzahl Triphone
WSJ, 5 000 Wörter	5 620
NAB, 20 000 Wörter	11 280
NAB, 64 000 Wörter	17 400

D.h. bei der Verwendung von Triphonmodellen statt von Monophonmodellen treten ähnliche Probleme auf wie bei Ganzwortmodellen. Zum einen werden viele Triphone nur sehr selten gesehen, so daß die Schätzungen der Parameter relativ unzuverlässig sind. Zum anderen kommen i.A. einige Triphone der Erkennungsvokabulars im Trainingskorpus gar nicht vor, so daß deren Parameter auch nicht geschätzt werden können. Als Lösung dieser Probleme wurden früher meist Glättungs- bzw. Backing-Off-Methoden verwendet:

- Bei der Glättung werden die Parameter der wenig gesehenen Triphonmodelle mit denen robust geschätzter allgemeinerer Modelle geglättet (z.B. Monophonmodelle), um den Schätzfehler der wenig gesehenen Modelle zu reduzieren.

- Beim Backing-Off werden Triphonmodelle, die nicht oder nicht häufig genug gesehen werden, auf Diphon- (Diphone = Phone im linken *oder* rechten Phonemkontext) oder Monophonmodelle abgebildet.

Nachteil beider Verfahren ist, daß die relativ genaue Modellierung durch Triphone durch eine weniger exakte ersetzt wird. Ein Verfahren, das diesen Nachteil vermeidet, ist das *State-Tying mit Entscheidungsbäumen*, das in Kapitel 5 beschrieben wird.

1.4 Sprachmodellierung

In der Sprachmodellierung wird versucht, für die a-priori-Wahrscheinlichkeit $P(w_1^N)$ der gesprochenen Wortfolge ein möglichst genaues Modell zu finden. Diese geht dann, wie in Abschnitt 1.1 beschrieben, in die Bayes'sche Entscheidungsregel ein. Mit dem Bayes'schen Gesetz läßt sich diese Wahrscheinlichkeit einer Wortfolge als Produkt von bedingten Wahrscheinlichkeiten von Wörtern schreiben:

$$P(w_1^N) = \prod_{n=1}^{N} P(w_n|w_1^{n-1})$$

Die Wahrscheinlichkeit des Wortes w_n hängt hier von der gesamten Worthistorie w_1^{n-1} ab. Will man diese exakte Form für die Satzwahrscheinlichkeit bestimmen, würde man z.B. für ein Vokabular von 1000 Wörtern für einen Satz der Länge n ca. 1000^n bedingte Wahrscheinlichkeiten schätzen müssen. Dies ist aufgrund von beschränktem Trainingsmaterial natürlich nicht möglich. Statt dessen werden die Worthistorien auf eine bestimmte Länge beschränkt, da man argumentieren kann, daß die Wahrscheinlichkeit eines Wortes in erster Linie von den m unmittelbaren Vorgängern abhängt:

$$P(w_1^N) = \prod_{n=1}^{N} P(w_n|w_{n-m+1}^{n-1}) \tag{1.17}$$

Der Parameter m bestimmt die Länge der Historie, bei $m = 2$ hängt die Wortwahrscheinlichkeit nur vom unmittelbaren Vorgänger ab, bei $m > 2$ von einer entsprechend längeren Historie, wobei die Exaktheit der Approximation mit steigendem m zunimmt. Für die Spezialfälle $m = 0$ und $m = 1$ vereinfacht sich die bedingte Wahrscheinlichkeit zu einer Gleichverteilung bzw. zu einer einfachen Wortwahrscheinlichkeit $P(w_n)$. Heute übliche Sprachmodelle verwenden $m = 2$ (*Bigramm*-Sprachmodell), $m = 3$ (*Trigramm*-Sprachmodell) oder $m = 4$ (*Viergramm*-Sprachmodell).
Bei großem Vokabular treten für m-Gramme ähnliche Probleme bzgl. ungesehener Ereignisse auf wie für Triphone. Für die Sprachmodellierung werden daher ebenfalls Glättungsverfahren verwendet wie z.B. *lineares* oder *absolutes Discounting*, bei denen den durch relative Häufigkeiten geschätzten Wahrscheinlichkeiten ein relativer (lineares D.) oder absoluter (absolutes D.) Betrag der Wahrscheinlichkeitsmasse abgezogen wird. Dieser wird dann geeignet auf die ungesehenen Ereignisse verteilt.
Um die Güte eines Sprachmodells zu bestimmen, kann die sogenannte *Perplexität* verwendet werden. Die Perplexität einer Wortfolge w_1^N mit einem Modell $P(w_n|w_1^{n-1})$ ist definiert als

$$PP = \left[\prod_{n=1}^{N} P(w_n | w_1^{n-1}) \right]^{-\frac{1}{N}}$$

Anschaulich beschreibt die Perplexität eines Testkorpus die mittlere Anzahl von Wörtern, die an einer Position in der Wortfolge zu Auswahl stehen bzw. wie sicher das Sprachmodell ein Wort aufgrund der Historie vorhersagen kann. Eine Verkleinerung der Perplexität auf dem Testkorpus bedeutet daher meist auch eine Verbesserung des Sprachmodells und damit i.a. eine Verbesserung der Fehlerrate des Erkennungssystems.

1.5 Suche

Unter dem Begriff "Suche" wird in der Spracherkennung die Maximierung des Ausdrucks 1.1 unter Verwendung der beiden Wissensquellen "akustisches Modell" und "Sprachmodell" verstanden. Die aufgrund dieser Maximierung optimale Wortfolge wird dann als vom Spracherkennungssystem erkannter Satz ausgegeben. Diese Optimierung wird prinzipiell durchgeführt, indem *alle* möglichen Wortfolgen mit Hilfe der beiden Wissensquellen "abgesucht", d.h. bewertet werden. Nur durch diese sogenannte *volle* Suche kann garantiert werden, daß in jedem Fall die *global* optimale Wortfolge gefunden wird. Dieser Ansatz ist allerdings nur für Spracherkennungssysteme mit kleinem Wortschatz (weniger als 100 Wörter) praktikabel, für größere Wortschätze wäre die volle Suche selbst bei effizienter Implementierung zu aufwendig. Beispielsweise müßte beim Ansatz der dynamischen Programmierung [Ney 84] für ein Vokabular von 100 000 Wörtern und einer angenommenen mittleren Wortlänge von 40 HMM-Zuständen pro Zeitpunkt $40 \cdot 100\,000 = 4\,000\,000$ HMM-Zustände ausgewertet werden. Dies entspräche bei einem gesprochenen Satz der Länge 10 Sekunden, was bei einer Framerate von 100 Frames/Sekunde genau 1 000 Frames entspricht, einem Suchraum von 4 000 000 000 HMM-Zuständen. Zur Vermeidung dieses Aufwandes gibt man die globale Optimalität auf, und beschränkt die Suche auf die vielversprechenden Satzhypothesen. Dadurch ist es möglich, auch für große Wortschätze ein akzeptables Antwortverhalten der Systeme zu erreichen, ohne daß die Fehlerrate signifikant ansteigt.
Die Bewertung einer Wortfolge w_1^N durch den Ausdruck 1.1 erfordert zum einen die Berechnung von $p(x_1^T | w_1^N)$ auf Basis des akustischen Modells und zum anderen die Berechnung von $P(w_1^N)$ auf Basis des Sprachmodells:

$$\left[w_1^N\right]_{opt} = \underset{w_1^N}{\operatorname{argmax}} \left\{ P(w_1^N) \cdot \sum_{s_1^T} p\left(x_1^T, s_1^T | w_1^N\right) \right\}$$

Aus Effizienzgründen vereinfacht man die Suchaufgabe, indem man die Summe über alle Pfade s_1^T durch das Maximum (Viterbi-Kriterium) ersetzt:

$$\left[w_1^N\right]_{opt} = \underset{w_1^N}{\operatorname{argmax}} \left\{ P(w_1^N) \cdot \max_{s_1^T} p\left(x_1^T, s_1^T | w_1^N\right) \right\}$$

Das zur Berechnung der Wahrscheinlichkeit $p(x_1^T|w_1^N)$ benötigte HMM-Satzmodell zu der Wortfolge w_1^N wird dabei durch Konkatenation der Triphon-HMM des Aussprachelexikons gebildet.
Prinzipiell können diese Maximierungen durchgeführt werden, indem zu allen möglichen Satzhypothesen w_1^N die Sprachmodell-Wahrscheinlichkeit $P(w_1^N)$ mittels Gleichung 1.17 bzw. die akustische Wahrscheinlichkeit $p(x_1^T|w_1^N)$ mittels Forward- oder Viterbi-Algorithmus berechnet werden. Dies verbietet sich natürlich aus Effizienzgründen, statt dessen wird die Optimierung parallel über alle Satzhypothesen durchgeführt, indem in der Maximum-Approximation der Suchraum als stochastisches endliches Zustandsnetzwerk aufgefaßt wird, in dem der wahrscheinlichste Pfad zu finden ist. D.h. die wahrscheinlichste Wortfolge wird durch die wahrscheinlichste Zustandsfolge approximiert. Die Suche ist sowohl auf Zustandsebene als auch auf Wortebene durchzuführen.
Zur Lösung des Suchproblems bieten sich prinzipiell zwei Suchansätze an:

- die Viterbi-Suche sowie
- die A^*-Suche.

Ein wesentlicher Unterschied zwischen beiden Suchmethoden liegt in der Auswertungsstrategie des Suchraums. Bezüglich der Zeitachse ist die Viterbi-Suche eine *Breiten*suche, d.h. die Zustandshypothesen werden *zeitsynchron* ausgewertet und expandiert. Im Gegensatz dazu arbeitet die A^*-Suche als *Tiefen*suche, die Hypothesen werden aufgrund ihrer aktuellen Bewertung prinzipiell *zeitasynchron* expandiert. Diese Bewertung besteht aus der bis zu diesem Zeitpunkt berechneten Teil-Wahrscheinlichkeit der Hypothese plus einer Schätzung der restlichen Wahrscheinlichkeit. Bei einer hinreichend genauen Schätzung dieser Restwahrscheinlichkeit kann die Auswertung der Hypothesen relativ stark eingeschränkt werden [Ortmanns 98]. Dieser Suchansatz wird in der Literatur auch als *stack decoding* bezeichnet. Einige Varianten dieses Verfahrens arbeiten allerdings zumindest teilweise zeitsynchron, so daß die Abgrenzung zu einer Viterbi-Suche mit Lookahead-Verfahren nicht in letzter Konsequenz möglich ist. Beide Suchansätze stellen die Basis derzeitiger Suchverfahren zur Erkennung kontinuierlich gesprochener Sprache dar.

Kapitel 2

Themen dieser Arbeit

Das State-Tying mit Entscheidungsbäumen [Hwang *et al.* 92] [Hwang 93] [Young *et al.* 94] und die Wortgrenzenmodellierung mit wortübergreifenden Triphonen [Alleva *et al.* 97] [Hon 92] [Odell *et al.* 94] [Odell 95] sind zwei Methoden der akustischen Modellierung, die Informationen über den phonetischen Kontext, in dem ein Lautmodell bei der Spracherkennung verwendet wird, dazu verwenden, um die Genauigkeit der Modellierung der gesprochenen Sprache zu verbessern.
In dieser Arbeit soll, ausgehend von einem System ohne diese beiden Verfahren, das State-Tying und die wortübergreifenden Triphone in das System integriert und systematisch optimiert werden. Für das State-Tying werden verschiedene Erweiterungen des Basisverfahrens anhand von Erkennungstests auf ihre Tauglichkeit überprüft. Für die Wortgrenzenmodellierung mit wortübergreifenden Triphonen werden die zwei wesentlichen Ansätze zur Implementierung des Suchalgorithmus gegenüber gestellt, nämlich die sogenannte *n-best-* und die *einphasige* Suche. Aufbauend auf diesen Erweiterungen wird ein neues Verfahren vorgestellt, das automatisch eine Menge von phonetischen Fragen generieren kann. Es wird demonstriert, daß das State-Tying unter Verwendung der von diesem Algorithmus erzeugten Fragen sowohl für rein wortinterne als auch für wortübergreifende Triphonmodellierung gute Ergebnisse liefert.
In den folgenden Abschnitten werden zu den oben angesprochenen drei Bereichen jeweils der Stand der Wissenschaft und der Beitrag dieser Arbeit dazu beschrieben.

2.1 State-Tying

Ein wesentliches Problem im Bereich der Spracherkennung ist es, eine Balancierung zwischen der Exaktheit der Modellierung des Sprachsignals und der Robustheit der trainierten Modelle zu erreichen. Dieser Tradeoff resultiert letztlich aus der Tatsache, daß im Allgemeinen nur eine begrenzte Menge von Trainingsmaterial zur Verfügung steht. Dies schränkt die mögliche Anzahl der Parameter des Erkenners ein, da eine gewisse Menge von Trainingsdaten pro Parameter verwendet werden muß, damit der Schätzfehler pro Parameter klein bleibt. Erhöht man die Anzahl der Parameter eines Modells bei gleicher Trainingsdatenmenge, erhöht sich auch der Schätzfehler, da

- die Anzahl der Parameter pro Modell steigt und/oder

- die Anzahl der Trainingsdaten pro Parameter sinkt.

Dieser Schätzfehler wirkt sich direkt auf die mit dem System zu erreichende Fehlerrate aus. Es ist eine bekannte Tatsache, daß, je mehr Trainingsmaterial zur Verfügung steht, desto mehr Parameter im System verwendet werden können, und desto geringer die Fehlerrate wird (siehe z.B. [Odell 95]). Aus diesem Grund muß sichergestellt werden, daß das Spracherkennungssystem die vorhandenen Trainingsdaten optimal ausnutzt, d.h. möglichst exakte Modelle mit möglichst wenig Parametern verwendet. Offensichtlich existieren zum Erreichen dieses Ziels zwei grundlegende Ansätze:

1. Erhöhung der Exaktheit der Modelle bei gleicher Parameterzahl

   Darunter fallen z.B. Normierungsverfahren, welche die Trainingsdaten so transformieren, daß die Variabilität der Daten sinkt, ohne daß Information verloren geht. Dadurch kann das Sprachsignal bei gleicher Parameterzahl besser modelliert werden.

2. Verringerung der Anzahl der Parameter der Modelle bei gleicher Exaktheit

   Darunter fallen alle Methoden, die Abhängigkeiten von Parametern ermitteln und aufgrund dieser Abhängigkeiten die Zahl der Parameter geeignet reduzieren. Dies sind z.B. die

   - Lineare Diskriminantenanalyse (LDA) [Haeb-Umbach & Ney 92] [Welling *et al.* 97] [Welling 98] oder das
   - Parameter-Tying [Lee 88] [Young & Woodland 93].

Thema dieser Arbeit ist das State-Tying mit Entscheidungsbäumen, ein Verfahren, das zur zweiten Gruppe gehört. Hierbei werden die Phonem-Zustände von Triphonen, deren akustische Realisierung bezüglich eines Abstandsmaßes ähnlich ist, verknüpft, so daß sie sich einen Parametersatz teilen und damit nur noch durch eine Mischverteilung modelliert werden. Die Abbildung der Triphonzustände auf die jeweilige Mischverteilung geschieht durch phonetische Entscheidungsbäume. Pro Phonem und Phonem-Segment wird ein Entscheidungsbaum verwendet. Den inneren Knoten des Entscheidungsbaums sind sogenannte *phonetische Fragen* zugeordnet, während an den Blättern Mischverteilungsindizes stehen. Die Zuordnung eines Triphonzustandes zu einer Mischverteilung geschieht, indem ausgehend von der Wurzel des Entscheidungsbaums sukzessive aufgrund der phonetischen Fragen verzweigt wird, bis ein Blatt erreicht ist. Die Mischverteilung, die durch den Index an diesem Blatt referenziert wird, wird dann zur Modellierung dieses Triphonzustandes verwendet (siehe Abbildung 2.1).
Durch das State-Tying mit phonetischen Entscheidungsbäumen wird folgendes erreicht:

- durch das Verknüpfen der Triphonzustände erhöht sich die Menge von Trainingsdaten pro HMM-Zustand, so daß deren Mischverteilungen besser geschätzt werden können,

- durch die phonetischen Entscheidungsbäume können auch ungesehenen Triphonzuständen geeignete Modelle zugeordnet werden.

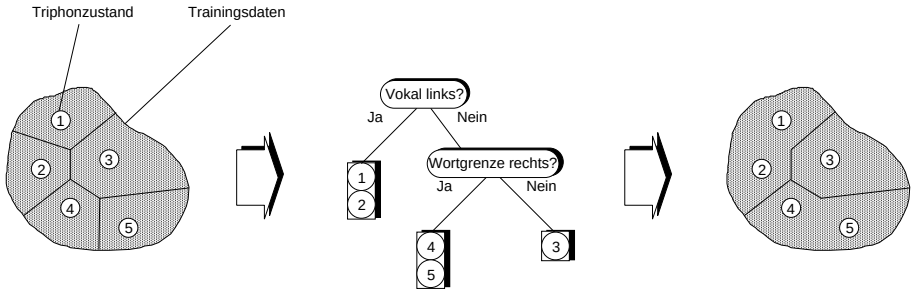


Abbildung 2.1: State-Tying mit phonetischen Entscheidungsbäumen

2.1.1 Stand der Wissenschaft

Die wichtigen Publikationen für das State-Tying mit phonetischen Entscheidungsbäumen sind:

- Lee [Lee 88] führte '88 die sogenannten generalisierten Triphone ein. Generalisierte Triphone differenzieren im Gegensatz zum State-Tying nicht nach dem Segment der HMM, sondern verknüpfen Triphone als ganzes miteinander. Das heißt, sind zwei Triphone miteinander verknüpft, dann teilen sich die ersten, die zweiten und die dritten Segmente jeweils eine gemeinsame Mischverteilung. Lee verwendete keine phonetischen Entscheidungsbäume, stattdessen wurden nicht gesehene Triphone auf Backing-Off-Modelle abgebildet. Auf dem *Resource Management* Task [Price *et al.* 88] konnte damit die Fehlerrate von 4.6% auf 4.2% gesenkt werden.

- Bahl [Bahl *et al.* 91] schlug '91 ein Verfahren vor, mit dem den Allophonen eines Phonems für beliebige Kontextlängen mit Hilfe von phonetischen Entscheidungsbäumen Modelle zugeordnet werden können. Für ein System mit diskreten HMM konnte damit die Fehlerrate von 9.2% auf 5.3% gesenkt werden.

- Hwang [Hwang *et al.* 92] führte 92 das oben beschriebenen State-Tying ein. Hierbei wurden zur Modellierung von nicht gesehenen Triphonzuständen ebenfalls Backing-Off-Modelle verwendet.

- Young [Young *et al.* 94] und Hwang [Hwang 93] kombinierten schließlich unabhängig voneinander das State-Tying mit den phonetischen Entscheidungsbäumen. Auf dem WSJ November 92 Task konnte, verglichen mit der Verwendung von Backing-Off-Modellen, die Fehlerrate um ca. 6% relativ [Hwang 93] bzw. ca. 3% relativ [Young *et al.* 94] gesenkt werden.

- Hon [Hon 92], Lazaridès [Lazarides *et al.* 96] und Paul [Paul 97] schlugen verschiedene Erweiterungen des Verfahrens vor, wobei allerdings entweder keine Fehlerraten für diese Erweiterungen angegeben wurden oder die Ergebnisse kein klares Bild ergaben, ob die jeweilige Erweiterung für großen Wortschatz zu einer Verbesserung der Wortfehlerrate führt.

2.2 Wortgrenzenmodellierung

In kontinuierlich gesprochener Sprache tritt wegen des weitgehenden Fehlens der Pausen zwischen den Wörtern der Effekt der Koartikulation auch an Wortgrenzen auf. Das bedeutet, die akustische Realisierung des Lautes an einem Wortende hängt von dem jeweiligen Laut am Wortanfang des nachfolgenden Wortes ab und umgekehrt. Sei beispielsweise der folgende Text mit phonetischer Transkription gegeben:

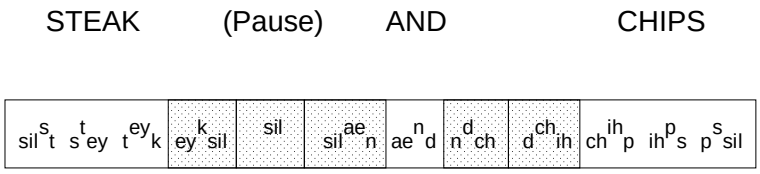


Abbildung 2.2: Wortgrenzenbehandlung für den englischen Satz "Steak and Chips"

Zwischen den ersten beiden Worten findet eine relativ lange Sprechpause statt, so daß keine Koartikulation an den Wortgrenzen stattfindet. Zwischen dem zweiten und den dritten Wort befindet sich keine wahrnehmbare Sprechpause, so daß hier die Wortgrenzenphoneme *d* und *ch* durch den jeweiligen Kontext beeinflußt werden.
Wie die Verwendung von wortinternen Triphonen zur Modellierung der Koartikulation im Wortinnern zeigt, sollte es für die automatische Spracherkennung vorteilhaft sein, auch die Koartikulation an Wortgrenzen geeignet zu modellieren. Um kontextabhängige Modelle auch an Wortgrenzen verwenden zu können, müssen in einem automatischen Spracherkennungssystem Änderungen sowohl im Training als auch in der Erkennung vorgenommen werden.

- Im Training muß für jeden Übergang zwischen zwei Wörtern die Länge der dazwischenliegenden Pause geschätzt werden. Auf Grund dieser Länge wird dann entschieden, ob an den Wortgrenzen Koartikulation modelliert wird oder nicht. Diese Entscheidung wird im einfachsten Fall unabhängig von dem konkreten Phonempaar am Wortübergang lediglich anhand der Länge der Pause getroffen.

- Um die Wortgrenzenmodellierung mit wortübergreifenden Triphonen in der Suche zu verwenden, gibt es grundsätzlich zwei Möglichkeiten,

  - die *n-best*-Suche und
  - die einphasige Suche.

  Bei der *n*-best-Suche wird in einem ersten Suchdurchgang ohne wortübergreifende Triphone die Liste der n besten Sätze ermittelt. Für jeden Satz dieser Liste wird dann ein Satzmodell mit wortübergreifenden Triphonen mittels des Viterbi-Algorithmus neu bewertet, der aufgrund dieser Bewertung beste Satz wird dann als erkannter Satz ausgewählt. Diese Neubewertung der Satzliste ist algorithmisch relativ einfach, da aufgrund der nun bekannten Wortfolge der jeweilige phonetische Kontext an den Wortgrenzen direkt in das HMM-Satzmodell einfließen kann. Das Problem bei dieser Art der Suche ist der stark eingeschränkte Suchraum, da die Anzahl der

möglichen Sätze exponentiell mit der Satzlänge wächst, so daß die n-best-Suche für Anwendungen mit sehr großen Satzlängen suboptimale Ergebnisse liefert.

Diese Nachteile können vermieden werden, indem die wortübergreifenden Triphone direkt in den Suchprozeß integriert werden. Hier muß, für den Fall der Baumsuche, der lexikalische Baum an den Wortenden aufgefächert werden. Jeder Ast am Ende des Baumes wird durch n Äste ersetzt, wobei n der Anzahl der möglichen Phoneme an einem Wortanfang plus Pause entspricht. Jeder dieser Äste modelliert für dieses Wortende eine mögliche Koartikulation aufgrund des Anfangsphonems p des folgenden Wortes bzw. einer folgenden Pause, bei der keine Koartikulation angenommen wird. Die an diesem Wortende gestartete neue Worthypothese muß dies berücksichtigen, indem nur der Teil des lexikalischen Baums berücksichtigt wird, der Wörter mit dem Phonem p am Wortanfang enthält.

2.2.1 Stand der Wissenschaft

Folgende Arbeiten sind für die Wortgrenzenmodellierung wichtig:

- Klovstad [Klovstad & Mondshein] verwendete im CASPERS-System lexikalische Bäume, bei denen die Enden der enthaltenen Wörter aufgrund von linguistischen Regeln modifiziert wurden, um Wortgrenzeneffekte zu modellieren.

- Lee [Lee 88], Paul [Paul 89] und Weintraub [Weintraub *et al.* 89] führten '89 unabhängig voneinander die Wortgrenzenmodellierung mit generalisierten wortübergreifenden Triphonen ein. Sie berichten von Verbesserungen der Fehlerrate von ca. 15-25% relativ.

- Hon [Hon 92] kombinierte '89 die Wortgrenzenmodellierung mit phonetischen Entscheidungsbäumen, wodurch die Modellierung von nicht gesehenen Triphonen insbesondere an Wortgrenzen verbessert wurde. Die durch die Wortgrenzenmodellierung ohne Entscheidungenbäume erreichen Verbesserungen der Fehlerrate liegen hier zwischen 20 und 30%.

- Odell [Odell 95] führte schließlich die einphasige Suche für wortübergreifende Triphonmodelle mit baumorganisiertem Lexikon ein. Er erreicht durch Wortgrenzenmodellierung eine Reduktion der Fehlerrate von 13 bzw. 30% für verschiedene Korpora.

- Beyerlein [Beyerlein *et al.* 97] berichtet für ein dem RWTH-System ähnliches Spracherkennungssystem unter Verwendung von generalisierten Bottom-Up-Triphonmodellen (siehe [Aubert *et al.* 96]) eine relative Verbesserung von ca. 6% durch Wortgrenzenmodellierung. Durch die zusätzliche Verwendung von Entscheidungsbäumen verbessert sich die Fehlerrate noch einmal um ca. 6% relativ.

2.3 Automatische Fragengenerierung

Bei der Verwendung von State-Tying mit phonetischen Entscheidungsbäumen werden die möglichen Aufteilungen der Triphonzustände an einem Knoten des Entscheidungsbaums eingeschränkt, indem nur bestimmte Phonemgruppierungen aufgrund einer Klassifikation

der Phoneme zugelassen werden. Diese phonetischen Klassen sind durch bestimmte artikulatorische Gemeinsamkeiten charakterisiert wie z.B. dem Ort der Lautbildung. Diese zur Konstruktion des Entscheidungsbaums verwendeten phonetischen Klassen werden im Kontext des State-Tying mit phonetischen Entscheidungsbäumen auch *phonetische Fragen* genannt. Eine Menge von phonetischen Fragen findet sich z.B. in Anhang C dieser Arbeit.
Typischerweise definiert ein phonetischer Experte für einen neuen Korpus diese Fragen. Dies ist vor allem dann erforderlich, wenn dieser in einer bis dahin noch nicht verwendeten Sprache aufgenommen ist. Daraus resultieren u.U. folgende Probleme:

- Oft ist kein phonetischer Experte vorhanden, der die phonetischen Fragen definieren kann.
- Die Definition der Fragen über die Verwandschaft der Phoneme bzgl. artikulatorischer Eigenschaften ist nicht zwangsläufig optimal für die Konstruktion eines phonetischen Entscheidungsbaums für die Spracherkennung.
- Bei der Verwendung eines neuen Korpus muß Zeit für die Definition der Fragen investiert werden.

2.3.1 Stand der Wissenschaft

In diesem Bereich existierten zur Zeit der Anfertigung dieser Arbeit noch keine Publikationen, in denen die Erzeugung von automatischen Fragen für phonetische Entscheidungsbäume beschrieben ist. Während dieser Zeit ist dem Autor lediglich ein Vortrag von D. McAllaster, *Dragon Systems* [McAllaster *et al.* 97] bekannt geworden, in denen eine ähnliche Methode beschrieben ist (siehe auch Kapitel 7).

Kapitel 3

Zielsetzung

In diesem Kapitel sollen nun zu den im vorherigen Kapitel angesprochenen Themen die offenen Fragen genannt und die Ziele bzgl. dieser Fragen, die in dieser Arbeit erreicht werden sollen, definiert werden.

3.1 State-Tying

3.1.1 Offene Fragen

Die bis zu diesem Zeitpunkt veröffentlichten Ergebnisse für Erweiterungen des State-Tying mit phonetischen Entscheidungsbäumen lassen keinen eindeutigen Schluß zu, ob eventuelle Modifikationen des Basisverfahrens wie

- geschlechtsabhängige Modelle in den Baumknoten,
- die Verwendung eines einzigen Baums und Fragen nach dem zentralen Phonem und dem Segmentindex,
- Beschränkung der Phonemmenge etc.

tatsächlich Verbesserungen des Basisverfahrens darstellen. Da die Modifikationen z.T. einen erheblichen Mehraufwand bei der Implementierung des Verfahrens darstellen, besteht natürlich die Frage, inwieweit der erhöhte Implementierungsaufwand durch die Verbesserung der Fehlerrate zu rechtfertigen ist.

3.1.2 Ziele dieser Arbeit

In dieser Arbeit werden zum State-Tying mit phonetischen Entscheidungsbäumen verschiedene z.T. in Veröffentlichungen beschriebene Erweiterungen des Basisverfahrens auf ihre Tauglichkeit untersucht. Dazu wird, ausgehend von einem Spracherkennungssystem ohne State-Tying, sowohl ein rein datengetriebenes State-Tying ohne Entscheidungsbäume als auch das State-Tying mit Entscheidungsbäumen für dieses System entwickelt und optimiert.
Aufbauend auf diesen Ergebnissen werden dann die verschiedenen Erweiterung für das State-Tying mit phonetischen Entscheidungsbäumen untersucht. Dies sind

- unterschiedlicher Umfang der Ausgangsmenge von Triphonzuständen bei der Erzeugung des Entscheidungsbaums,
- die Verwendung eines einzigen Baums und Fragen nach zentralem Phonem und Segnmentindex statt einem separaten Baum für jedes Phonemsegment,
- Verschmelzen von Blättern nach Abschluß des Aufteilens der Blätter des Entscheidungsbaums,
- geschlechtsabhängige Modelle in den Baumknoten,
- die Verwendung einer vollen statt einer diagonalen Kovarianzmatrix.

Für diese Erweiterungen werden Fehlerraten auf dem *Wall Street Journal*-Spracherkennungskorpus für ein 5 000-Wort-Lexikon angegeben.

3.2 Wortgrenzenmodellierung

3.2.1 Offene Fragen

Obwohl die Wortgrenzenmodellierung mit wortübergreifenden Triphonen in verschiedenen Veröffentlichungen behandelt worden ist, fehlen bis jetzt systematische Untersuchungen zu wichtigen Fragestellungen in diesem Zusammenhang, u.a.

- Wie lang soll der Schwellwert für die Pauselänge zwischen Wörtern gewählt werden, der festlegt, bis zu welcher Pauselänge Koartikulation modelliert wird?
- Wie beeinflußt die erste Suchphase das Ergebnis der *n-best*-Suche?
- Welche Länge der *n-best*-Liste ist optimal bzgl. Fehlerrate?
- Kann durch eine Interpolation zwischen den rein wortinternen und den wortübergreifenden Triphonmodellen die Fehlerrate verbessert werden? Welche Fehlerraten lassen sich, bei gleicher Modellierung, mit der einphasigen Suche erreichen?

3.2.2 Ziele dieser Arbeit

Die Ziele dieser Arbeit bzgl. der oben genannten Fragen lassen sich in zwei Bereiche gliedern:

- *n-best*-Suche

  Im Rahmen dieser Arbeit wird die *n-best*-Suche für die Wortgrenzenmodellierung miti wortübergreifenden Triphonemodellen implementiert und optimiert. Dazu werden Fehlerraten auf zwei verschiedenen Testkorpora (*Wall Street Journal* und *Verbmobil*) ermittelt. Weiterhin wurde der Effekt einer linearen Interpolation der Satzbewertungen mit rein wortinternen und wortübergreifenden Triphonmodellen untersucht.

- *einphasige* Suche

  Das zweite für diese Arbeit implementierte Suchverfahren für die Wortgrenzenmodellierung mit wortübergreifenden Triphonmodellen ist das *einphasige* Suchverfahren. Dieses wird ebenfalls auf den Testkorpora *Wall Street Journal* und *Verbmobil* optimiert und der Effekt der linearen Interpolation auf die Fehlerrate untersucht.

3.3 Automatische Fragengenerierung

3.3.1 Ziele dieser Arbeit

Auf der Basis der vorgenommenen Erweiterungen des Standardsystems wird der Algorithmus zur automatischen Fragengenerierung entwickelt und in das System integriert. Dabei wird sowohl ein zufallsbasiertes als auch datengetriebene Verfahren eingesetzt. Weiterhin wird eine Modifikation definiert und getestet, die die Robustheit der generierten Fragen erhöhen soll. Die Methode wird auf dem *Wall Street Journal*-Korpus für ein 5 000-Wort-Lexikon und auf dem *Verbmobil*-Korpus für ein 5 000-Wort-Lexikon (siehe auch Kapitel 4) sowohl für rein wortinterne als auch für wortübergreifende Triphone evaluiert.

Kapitel 4

Verwendete Korpora und Testbedingungen

Um die in dieser Arbeit beschriebenen Methoden zu evaluieren, wurden Spracherkennungstests auf verschiedenen Standardkorpora für Spracherkennung mit großem Wortschatz durchgeführt. Die verwendeten Korpora waren

- *Wall Street Journal* und
- *Verbmobil*.

Im folgenden soll nun kurz eine Beschreibung dieser Korpora erfolgen.

4.1 Wall Street Journal

Der *Wall Street Journal*-Korpus ist ein von der ARPA zusammengestellter Spracherkennungskorpus für großes Vokabular. Die gesprochenen Texte stammen aus Artikeln der amerikanischen Wirtschaftszeitung *Wall Street Journal*. Diese wurden von verschiedenen nativen amerikanischen Sprechern unter relativ störungsfreien Bedingungen aufgenommen. Dazu wurden verschiedene Mikrofone verwendet, die in dieser Arbeit beschriebenen Tests beschränken sich auf die Sprachproben, die mit einem Headset (Sennheiser) aufgenommen wurden.
Die Trainingsdaten lassen sich weiterhin wie folgt charakterisieren. Die durch die Sprecher vorzulesenen Texte wurden in zwei gleich große Teile geteilt. Für beide Teile wurde dann die Art und Weise, wie Zeichensetzungen zu lesen waren, unterschiedlich definiert. In der einen Hälfte des Datenmaterials wurden dazu die Zeichensetzungen durch die entsprechende Verbalisierung ersetzt, also z.B. "," durch "Comma" (*verbalisierte* Zeichensetzung). In der anderen Hälfte des Textes wurden die Zeichensetzungen komplett entfernt (*nicht verbalisierte* Zeichensetzung). Die Sprecher wurden dann angewiesen, die so vorgefilterten Texte exakt Wort für Wort zu lesen. Dadurch erhielt man den als WSJ0 bezeichneten Trainingskorpus, der aus den Teilen

- Longitudinal Speaker Dependent, *LSD*,
- Long Term Speaker Independent, *SI12* und

- Short Term Speaker Independent, *SI84*.

zusammengesetzt ist, die Teilkorpora bestehen jeweils zur Hälfte aus Texten mit verbalisierter bzw. mit nicht verbalisierter Zeichensetzung. Für das Training der Modelle, die in den in dieser Arbeit beschriebenen Tests verwendet wurden, wurde nur der *SI84*-Teilkorpus benutzt. Dieser besteht aus 7 193 Sätzen von 84 Sprechern, von denen jeweils 42 männlich und 42 weiblich sind. Die Länge des Korpus beträgt ca. 12.2 Stunden.
Die verwendeten Testdaten sind entnommen aus der ARPA-Evaluierung, die im November 1992 für ein 5 000 Wörter umfassendes Lexikon durchgeführt wurden. Diese bestehen aus zwei Teilkopora:

- *dev92*

  Dieser Teilkorpus wurde als Vorbereitung für die eigentliche Evaluierung zur Konfigurierung des Spracherkennungssystems, wie z.B. der Einstellung der Gewichtung des Sprachmodells, verwendet. Er besteht aus 410 Sätzen bzw. 6779 Wörtern von 10 Sprechern (6 männliche und 4 weibliche). Als Lexikon wurde das von *Dragon Systems* 1992 für die Evaluierung zur Verfügung gestellte Lexkon verwendet. Es enthält für die 5 000-Wort-Aufgabe insgesamt 4986 Wörter.

- *evl92*

  Dieser Teilkorpus wurde für die eigentliche Evaluierung der Systeme verwendet. Er besteht aus 330 Sätzen bzw. 5353 gesprochenen Wörtern von 8 Sprechern (4 männliche und 4 weibliche). Als Lexikon wurde ebenfalls das Dragon-Lexikon für die 5 000 Wort-Aufgabe verwendet.

Dieser Korpus soll im folgenden mit der Abkürzung *WSJ-5k* referenziert werden.

4.2 Verbmobil

Der Verbmobil-Korpus wurde im Zusammenhang mit dem vom BMBF initiierten *Verbmobil*-Projekt [Verbmobil Web Site] zusammengestellt. Dieses Projekt zielt darauf ab, ein System zu entwickeln, das in der Lage ist, Sprache zu erkennen, diese dann zu übersetzen und schließlich den übersetzten Text per Sprachsynthese wieder auszugeben. Dieser "elektronische Dolmetscher" soll dann z.B. zur Kommunikation zwischen Geschäftspartnern, die keine gemeinsame Sprache sprechen, eingesetzt werden können.
Im Teilprojekt "Signalnahe Spracherkennung" sind z.Z. drei Gruppen mit ihren Spracherkennungsystemen für jeweils unterschiedliche Zielsetzungen vertreten:

- Universität Karlsruhe: Multilingualität
- Daimler Benz: Robustheit
- RWTH Aachen: Großes Vokabular

Diese Spracherkennungssysteme werden ebenfalls regelmäßig evaluiert, obwohl hier die Vergleichbarkeit wegen der verschiedenen Zielsetzungen nicht ohne weiteres möglich ist.

Die zur Evaluierung dieser Systeme verwendeten Korpora bestehen aus spontansprachlichen Dialogen über Terminabsprachen, die von Sprechern verschiedenen Geschlechts und Lokalität (Dialekte) aufgenommen wurden. Die einzelnen Dialogabschnitte haben eine Länge von wenigen Sekunden bis zu einer 3/4 Minute. Die Aufnahmequalität der Sprachdaten ist hoch (HiFi-Qualität). Die Menge an Trainingsmaterial pro Sprecher ist dabei relativ unterschiedlich, sie reicht von einem bis zu 40 Sätzen pro Sprecher.
Der zur Evaluierung der in dieser Arbeit vorgestellten Methoden verwendete Korpus entspricht dem Korpus, der 1996 zur offiziellen Evaluierung der Spracherkennungssysteme verwendet wurde:

- Der Trainingskorpus besteht aus insgesamt 27.8 Stunden Sprachdaten aus 610 Dialogen. Das Trainingslexikon umfaßt 26 672 Einträge, wovon allerdings nur ein Teil im Korpus vorkommt.
- Zur Evaluierung wurden 35 Dialoge bestehend aus 305 Sätzen bzw. 5421 Wörtern verwendet. Diese hatten eine Gesamtdauer von ca. 40 Minuten.

Dieser Korpus soll im folgenden mit der Abkürzung *VM-5k* referenziert werden.
Abschließend werden die Eigenschaften der beiden Korpora noch einmal in Tabellenform gezeigt (siehe Tabellen 4.1, 4.2). Da die Zahl der Sprecher beim Verbmobil-Korpus aufgrund der Struktur des Korpus nicht ermittelt werden konnte, wird stattdessen die Zahl der Dialoge angegeben.

Tabelle 4.1: Eigenschaften der verwendeten Trainingskorpora.

Korpus	Sprecher male/female	Anzahl Sätze	Länge [h]
WSJ-5k	42/42	7 193	12.2
VM-5k	610 Dialoge	11 234	27.8

Tabelle 4.2: Eigenschaften der verwendeten Testkorpora.

Korpus	Sprecher male/female	Anzahl Sätze	Länge [h]
WSJ-5k	10/8	740	1.5
VM-5k	35 Dialoge	305	0.67

Kapitel 5

State-Tying mit Entscheidungsbäumen

5.1 Einführung

Das Verknüpfen der akustischen Modelle bzgl. bestimmter Parameter, sogenanntes Parameter-Tying, ist auf verschiedenen Ebenen des Spracherkennungssystems möglich, z.B. auf Ebene der Triphone, der Zustände oder der Varianzen. Verknüpfte Modelle verwenden für die jeweils verknüpften Parameter dieselben Werte, so daß sich die Gesamtzahl der freien Parameter des Erkennungssystems reduziert und so die verbleibenden Parameter robuster geschätzt werden können. In dieser Arbeit wird das State-Tying betrachtet, bei dem die Parameter der Emissionsverteilungen auf Zustandsebene verknüpft werden. Dieses Verfahren wurde zuerst von [Young *et al.* 94] bzw. [Hwang 93] als Verbesserung des Phonem-Tying verwendet, bei dem lediglich ganze Phonemmodelle verknüpft wurden.
Für das State-Tying werden zwei Verfahren betrachtet, um die zu verknüpfenden Zustände zu ermitteln, ein rein datengetriebenes Bottom-Up-Verfahren, das Cluster aus Triphonzuständen bildet [Young & Woodland 93], und ein Top-Down-Verfahren, das mit Hilfe von phonetischen Fragen (Expertenwissen) Entscheidungsbäume generiert [Bahl *et al.* 91] [Young *et al.* 94] [Hwang 93]. Dieses zweite Verfahren wird Gegenstand eingehender Untersuchungen sein, da die anderen in dieser Arbeit betrachteten Methoden darauf basieren.

5.2 State-Tying

Die akustische Modellierung des in dieser Arbeit verwendeten Spracherkennungssystems basiert auf durch HMM modellierten kontextabhängigen Phonemmodellen im Triphonkontext, Triphone genannt (siehe auch Kapitel 1). Mit diesen Triphonen werden die akustischen Realisierungen der Worte des Trainings- und Erkennungsvokabulars modelliert. D.h. die Erkennungsleistung des Systems hängt in erster Linie davon ab, wie gut die diesen Triphonen zugrundeliegende akustische Modellierung ist. Zum einen soll das akustische Modell die Realität möglichst genau wiederspiegeln. Zum anderen müssen für die Parameter des akustischen Modells genug Trainingsdaten vorhanden sein, damit diese Parameter robust geschätzt werden können. Bei den meisten heute verwendeten Spracherkennungskorpora treten nun zwei Probleme auf:

1. Viele Triphone des Trainingskorpus werden nur sehr selten in dem Korpus gesehen, daher sind für diese Triphone auch nur sehr wenig Trainingsdaten vorhanden.

2. Oft sind Trainings- und Erkennungslexikon nicht identisch, daher kann es vorkommen, daß Triphone aus dem Testkorpus im Trainingskorpus gar nicht vorkommen.

Zum ersten Problem soll die folgende Tabelle betrachtet werden. Dargestellt ist ein Histogramm über die Anzahl der Vorkommen von Triphonen im WSJ0-Trainingskorpus (siehe Tabelle 5.1):

Tabelle 5.1: Histogramm über die Häufigkeiten von Triphonen im Training für den WSJ0-Trainingskorpus.

Häufigkeit im Trainingstext	Anzahl der Triphone
1-9	3737
10-19	1072
20-29	575
30-39	358
40-49	259
50-59	199
60-69	188
70-79	145
80-89	130
90-99	89
≥ 100	1082
Summe	7834

Offensichtlich werden die Hälfte aller in diesem Trainingskorpus vorkommenden Triphone weniger als zehnmal gesehen. Folglich kann man bei einem Spracherkennungssystem, das die Wörter des Lexikons ausschließlich mit echten Triphonmodellen modelliert, erwarten, daß die meisten Triphonmodelle zu wenig Trainingsdaten für eine robuste Parameterschätzung erhalten.
Das zweite Problem läßt sich ebenfalls an diesem Korpus darstellen. Ermittelt man alle im Erkennungskorpus *WSJ Nov. 92* vorkommenden Triphone, so liegen ca. 4% dieser Triphone nicht in der Triphonmenge des Trainingskorpus (siehe auch Tabelle 5.5). D.h., kommen diese Triphone während der Erkennung vor, kann ihnen kein akustisches Modell zugeordnet werden.
Ein einfaches Verfahren, dieses Problem zu beseitigen, ist das sogenannte *Monophon-Backing-Off* [Ney 90]. Dazu teilt man die Menge der Triphone in zwei disjunkte Mengen auf. Für die eine Menge werden wie gewöhnlich akustische Modelle aus den Trainingsdaten geschätzt, während man die Trainingsdaten der anderen Menge dazu verwendet, um statt der Triphone akustische Modelle für die allgemeineren Monophone zu schätzen. Diese Monophonmodelle werden dann verwendet, um in der Erkennung nicht gesehene Triphone zu approximieren. Da die Zahl der Monophone wesentlich kleiner als die Zahl der Triphone ist, benötigt man auch wesentlich weniger Trainingsmaterial, um diese zu schätzen.

Man verwendet zur Aufteilung der Triphone meist ein einfaches Kriterium, das auf der Häufigkeit der Triphone im Trainingskorpus beruht. Ein Schwellwert τ ordnet die Triphone, die öfter als τ-mal im Trainingskorpus vorkommen, der Menge der "echten" Triphone zu, während die Trainingsdaten der restlichen Triphone zu Schätzung der Monophone verwendet werden. Nachteil dieser Methode ist, daß relativ wenige Triphone in die erste Menge aufgenommen werden können, da für die meisten Triphone nicht genügend Trainingsdaten vorhanden sind, um eigene Emissionsverteilungen für diese Triphone zu trainieren (siehe Tabelle 5.1). Konsequenz ist, daß alle anderen Triphone plus der Triphone, die nur im Erkennungslexikon vorkommen, durch Monophone modelliert werden, die die Abhängigkeit der akustischen Realisierung eines Phonems im Kontext nicht erfassen können.
Um diese Situation zu verbessern, kann das sog. State-Tying verwendet werden [Bahl *et al.* 91] [Hwang 93] [Young *et al.* 94]. Beim State-Tying werden die Zustände der HMM, deren Emissionsverteilungen bzgl. eines Abstandsmaßes ähnlich sind, verknüpft. D.h., die verknüpften Zustände teilen sich dieselbe Emissionsverteilung. Da nun die Beobachtungen aller verknüpften Zustände zur Schätzung der Parameter dieser Verteilung verwendet werden können, kann diese wesentlich robuster geschätzt werden. Die Bestimmung einer solchen Verknüpfung geschieht in drei Schritten (siehe Abbildung 5.1):

1. Für jeden der zu verknüpfenden Triphonzustände wird eine einfache Emissionsverteilung geschätzt.

2. Die Zustände werden dann bzgl. eines Abstandsmaßes verknüpft, wobei darauf zu achten ist, daß alle verknüpften Zustände eine Mindestzahl an Beobachtungen erhalten.

3. Anschließend werden die verknüpften Zustände mit einer höheren akustischen Auflösung reestimiert.

Auf diese drei Schritte soll nun im weiteren näher eingegangen werden.

5.2.1 Schätzung einfacher Emissionsverteilungen

Im Falle des RWTH-Systems werden dazu Gaußverteilungen mit diagonaler Kovarianzmatrix verwendet, die zwar die Mischverteilungen der HMM-Zustände des Erkennungsssystems relativ ungenau approximiert, dafür aber sehr robust geschätzt werden kann (wenige Parameter) und auch rechentechnisch Vorteile bietet. Alternativ dazu könnte man auch eine Gaußverteilung mit voller Kovarianzmatrix oder eine Mischverteilung mit u.U. reduzierter Komponentenzahl verwenden. Die Verwendung einer vollen Kovarianzmatrix wurde im Rahmen dieser Arbeit untersucht, wobei sich gezeigt hat, daß diese Modellierung zu keiner Verbesserung führt. Fall 2, also die Verwendung von Mischverteilungen schon während der Konstruktion des Entscheidungsbaums, ist algorithmisch relativ komplex und wurde daher in dieser Arbeit nicht weiter betrachtet (siehe Kapitel 8).
Die analytische Form einer Gaußverteilung soll nun als Basis für die weiteren Betrachtungen kurz erläutert werden:

$$p(x|\mu,\Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}}\det(\Sigma)^{\frac{1}{2}}}\exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))$$

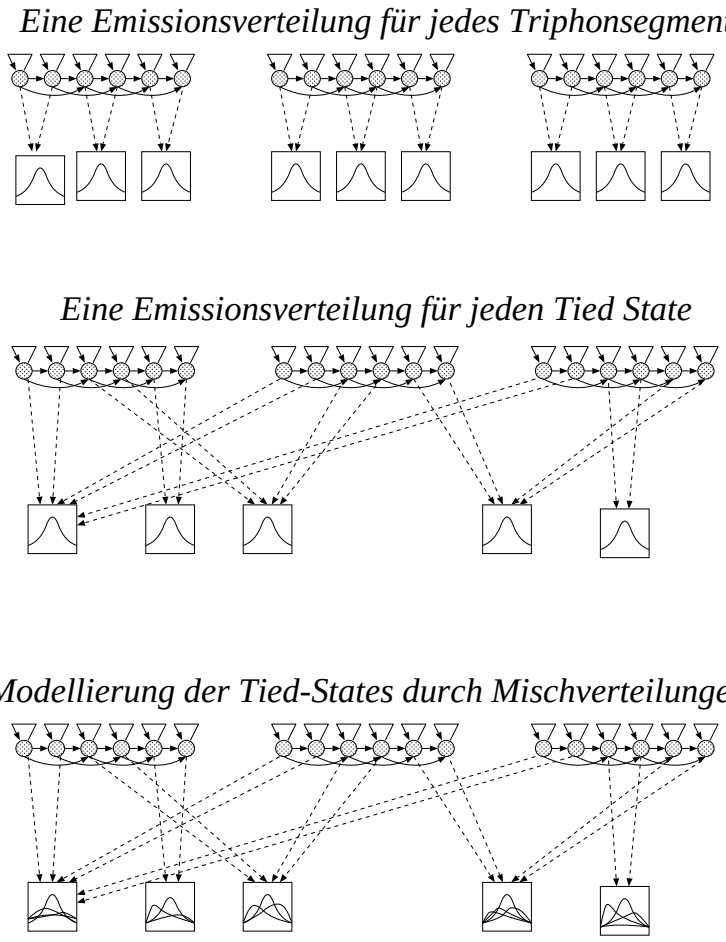


Abbildung 5.1: Ablauf des State-Tying.

μ entspricht bei einer Gaußverteilung dem Erwartungswert $E(x)$ des Parameters x, die Matrix Σ entspricht der Kovarianz $Cov(x)$, D ist die Dimension des akustischen Vektors. Diese Parameter der Emissionsverteilung können anhand einer vorher berechneten Segmentierung der Trainingsdaten geschätzt werden. Wie in Kapitel 1 schon erwähnt, wird dazu für Σ die Vereinfachung gemacht, daß alle Komponenten der Matrix σ_{ij} mit $i \neq j = 0$ angenommen werden, d.h. die Komponenten des akustischen Vektors werden als unkorrelliert angenommen. Dadurch vereinfacht sich die Schätzung der Parameter für einen Zustand s bei Verwendung des Maximum-Likelihood-Kriteriums zu:

$$\mu_{sd} = \frac{1}{N_s} \sum_{n=1}^{N_s} x_{snd} \tag{5.1}$$

$$\sigma_{sd}^2 = \frac{1}{N_s} \sum_{n=1}^{N_s} (x_{snd} - \mu_{sd})^2 \tag{5.2}$$

$$= \left(\frac{1}{N_s} \sum_{n=1}^{N_s} x_{snd}^2 \right) - \mu_{sd}^2 \tag{5.3}$$

Da nach diesen Formeln die Berechnung der Parameter mit der Anzahl, der Summe und

der Summe der Quadrate der Beobachtungen $x_1^{N_s}$ erfolgen kann, ist es möglich, diese Summen vor der Bestimmung des State-Tyings vorab einmal zu bestimmen, und dann mit diesen Summen unterschiedliche Tyings, beispielweise mit verschiedenen Anzahlen von verknüpften Zuständen, zu berechnen. Voraussetzung ist allerdings, daß sowohl die akustischen Vektoren selbst als auch die Zuordnung dieser Vektoren zu den HMM-Zuständen des akustischen Modells unverändert bleiben.

5.2.2 Verknüpfung der Zustände

Auf Basis der im ersten Schritt geschätzten Parameter der Gauß-Verteilungen der Triphonzustände werden die Verknüpfungen der Zustände berechnet. Eine Verknüpfung zweier HMM-Zustände soll dabei bedeuten, daß die Emissionsverteilungen der verknüpften Zustände durch eine einzige Mischverteilung modelliert wird. Durch diese Verknüpfung stehen für die Schätzung der Parameter der Mischverteilung eine größere Menge von Trainingsdaten zur Verfügung, wodurch diese robuster geschätzt werden können. Dies bedeutet auch, daß die Verknüpfungen so gewählt werden müssen, daß ausschließlich akustisch ähnliche HMM-Zustände verknüpft werden. Diese akustische Ähnlichkeit wird für das State-Tying durch ein zu definierendes Abstandsmaß beschrieben. Bzgl. eines solchen akustischen Abstandsmaßes verknüpfte Zustände bilden einen sogenannten *Cluster* im akustischen Merkmalsraum.
Ein Maß dafür, wie gut eine Verteilung eine Menge von Daten beschreibt, ist die sogenannte *Likelihood-Funktion*. Sie ist allgemein definiert als:

$$L(\Theta|x_1^N) = \prod_{n=1}^{N} p(x_n|\Theta)$$

wobei Θ allgemein die Parameter der Verteilung darstellt. Je besser die Verteilung $p(x)$ die Daten x_1^N beschreibt, desto größer wird auch die Likelihood-Funktion. In der Praxis hat es sich als sinnvoll herausgestellt, die Likelihood-Funktion in negativer logarithmierter Form zu verwenden:

$$LL(\Theta|x_1^N) = -\sum_{n=1}^{N} \log p(x_n|\Theta)$$

Für die Gauß-Verteilung mit diagonaler Kovarianzmatrix ergibt sich:

$$\begin{aligned}
LL(\Theta|x_1^N) &= -\frac{1}{2}\sum_{n=1}^{N}\left[\sum_{d=1}^{D}\left(\frac{x_{nd}-\mu_d}{\sigma_d}\right)^2 + \sum_{d=1}^{D}\log(2\pi\sigma_d^2)\right] \\
&= -\frac{1}{2}\left[\sum_{d=1}^{D}\left[\frac{1}{\sigma_d^2}\sum_{n=1}^{N}(x_{nd}-\mu_d)^2\right] + N\log(2\pi\sigma_d^2)\right] \\
&= -\frac{1}{2}\left[\sum_{d=1}^{D}\left[\frac{N}{\sum_{n=1}^{N}(x_{nd}-\mu_d)^2}\sum_{n=1}^{N}(x_{nd}-\mu_d)^2\right] + N\log(2\pi\sigma_d^2)\right] \\
&= -\frac{1}{2}\left[ND + N\sum_{d=1}^{D}\log(2\pi\sigma_d^2)\right]
\end{aligned}$$

Leitet man diese Funktion nach den Parametern der Gauß-Verteilung ab, erhält man als Schätzfunktionen die oben erwähnten Formeln 5.2 und 5.3.
Für das State-Tying wird die Likelihood-Funktion insbesondere als *Optimalitätskriterium* für die Verknüpfung von Zuständen verwendet, wie im nächsten Abschnitt dargestellt wird.
Gegeben sei eine Menge von HMM-Zuständen $s_1 ... s_N$. Die Log-Likelihood der Beobachtungen $x_1^{N_s}$, die diesen Zuständen zugeordnet sind, werden bezüglich der Gaußverteilungen der Zustände $N(\mu_s, \Sigma_s)$ mit der oben genannten Formel auf den Daten $x_1^{N_s}$ berechnet. Werden diese Zustände zu einem Cluster C verknüpft, ergibt sich für die Parameter der Gauß-Verteilung der verknüpften Zustände

$$\begin{aligned} \mu_{Cd} &= \frac{1}{N_C} \sum_{s \in C} \sum_{n=1}^{N_s} x_{snd} \\ \sigma_{Cd}^2 &= \left(\frac{1}{N_C} \sum_{s \in C} \sum_{n=1}^{N_s} x_{snd}^2 \right) - \mu_{Cd}^2 \end{aligned}$$

N_C ist die Anzahl der akustischen Vektoren für den Cluster C, N_s die Anzahl der akustischen Vektoren für den Zustand s. Aufgrund dieser Parameterschätzung kann nun für die verknüpften Zustände ebenfalls eine Log-Likelihood LL_C für die akustischen Vektoren eines Clusters C berechnet werden. Mit Hilfe dieser Log-Likelihood ist es nun möglich, zu einer gegebenen Zustandsmenge S und einer vorgegebenen Anzahl von verknüpften Zuständen M eine bzgl. der Log-Likelihood optimale Verknüpfung zu definieren als

$$[C_1, ..., C_M]_{opt} = \underset{C_1,...,C_M}{\operatorname{argmin}} \sum_{i=1}^{M} LL_{C_i} \tag{5.4}$$

wobei gilt:

$$\bigcup_{i=1}^{M} C_i = S \tag{5.5}$$

$$C_i \cap C_j = \emptyset \qquad \forall\, i, j \in \{1, \ldots, M\}, i \neq j \tag{5.6}$$

D.h. die Log-Likelihood-Funktion wird über alle möglichen Unterteilungen der Zustände in M Cluster minimiert. Da die Komplexität dieses Optimierungsproblems exponentiell ist, ersetzt man "möglichen" durch "sinnvollen", d.h. man schränkt den Suchraum der zu testenden Verknüpfungen durch sog. *Clusterverfahren* geeignet ein.

5.2.3 Clusterverfahren

Clusterverfahren werden im allgemeinen dazu verwendet, in einem Merkmalsraum Mengen von Beobachtungen zu identifizieren, deren Abstand zueinander signifikant geringer ist als zu anderen Mengen von Beobachtungen. Man nennt diese Mengen von Beobachtungen

dann auch *Cluster*. Diese Cluster können in zwei- oder dreidimensionalen Räumen optisch als Zusammenballungen von Beobachtungen charakterisiert werden.
Aufgrund der oben erwähnten Problematik, daß die Anzahl aller möglichen Unterteilungen einer Menge exponentielle Komplexität hat, verwenden die meisten Clusterverfahren lokal optimale Entscheidungen, so daß die globale Optimalität bzgl. der Zielfunktion i.A. verloren geht. Dafür besitzen diese Verfahren polynomielle Komplexität, so daß sie auch für große Datenmengen einsetzbar sind. Im allgemeinen können diese Verfahren bzgl. der Vorgehensweise des Algorithmus eingeteilt werden in

- Bottom-Up-Verfahren und
- Top-Down-Verfahren.

Beide Ansätze verwenden für die Entscheidung, welche Daten zu einem Cluster zusammengefaßt werden sollen, ein Abstandsmaß. Dieses Abstandsmaß quantifiziert die "Ähnlichkeit" zwischen den Beobachtungen, die im Merkmalsraum geclustert werden sollen. In dieser Arbeit wird ein Abstandsmaß verwendet, das auf der Differenz der Log-Likelihood-Funktion für Gauß-Verteilungen von zwei getrennten Clustern C_1 und C_2 und der Vereinigung von C_1 und C_2, im folgenden C_0 genannt, beruht. Diese läßt sich schreiben als

$$\Delta LL = LL_{C_1} + LL_{C_2} - LL_{C_0} \tag{5.7}$$

mit

$$C_0 = C_1 \cup C_2, \qquad C_1 \cap C_2 = \emptyset$$

Setzt man die obige Formel für die Log-Likelihood ein, ergibt sich

$$\begin{aligned}
\Delta LL &= -\frac{1}{2} N_{C_1} (D + \sum_{d=1}^{D} \log(2\Pi\sigma_{C_1,d}) \\
&\quad -\frac{1}{2} N_{C_2} (D + \sum_{d=1}^{D} \log(2\Pi\sigma_{C_2,d}) \\
&\quad +\frac{1}{2} N_{C_0} (D + \sum_{d=1}^{D} \log(2\Pi\sigma_{C_0,d}) \\
&= \frac{1}{2} \left[N_{C_0} \sum_{d=1}^{D} \log \sigma_{C_0,d} - \left(N_{C_1} \sum_{d=1}^{D} \log \sigma_{C_1,d} + N_{C_2} \sum_{d=1}^{D} \log \sigma_{C_2,d} \right) \right]
\end{aligned}$$

Andere Abstandsmaße sind allgemein in [Breiman *et al.* 84] bzw. speziell auf das Problem des State-Tying in der Spracherkennung bezogen in [Kramer 96] betrachtet worden, wobei sich keine signifikanten Unterschiede zwischen den verschiedenen Maßen bzgl. der erzielten Fehlerrate ergeben hat.
Bei *Bottom-Up-Verfahren* werden nun immer die beiden Cluster *zusammengefaßt*, für die die Log-Likelihood-Differenz zwischen C_1 und C_2 und dem zusammengefaßten C_0

möglichst *klein* ist, was eine große Ähnlichkeit zwischen den Ausgangsmodellen und dem zusammengefaßten Modell impliziert. Bei *Top-Down-Verfahren* wird entsprechend der Cluster *aufgespalten*, für den sich eine möglichst *große* Differenz der Log-Likelihood ergibt, d.h. für die sich die Modelle für C_1 und C_2 möglichst stark unterscheiden.

5.2.3.1 Bottom-Up-Verfahren

Ausgangspunkt für das Bottom-Up-Verfahren für das State-Tying [Young & Woodland 93] ist die Menge von nicht verknüpften Triphonzuständen $s_1, ..., s_N$. Das Verknüpfen der Zustände erfolgt dann in zwei Phasen:

- In der ersten Phase werden die Triphonzustände sukzessive paarweise zusammengefaßt, wobei in einem Schritt immer das Paar verknüpft wird, das bzgl. des Abstandsmaßes den geringsten Abstand besitzt, d.h. im Fall des Log-Likelihood-Abstandsmaßes ΔLL die geringste Verschlechterung der Log-Likelihood-Funktion bewirkt. Die entstehenden Cluster werden so lange weiter verknüpft, bis die Abstände aller Paare eine Schwelle τ_{dist} überschreiten.

- In der zweiten Phase werden dann die Zustandscluster mit ihrem nächsten Nachbarn verknüpft, für die die Menge an Trainingsdaten unter einer Schwelle τ_{minobs} liegt. Dadurch soll sichergestellt werden, daß zur Schätzung der Emissionsverteilungen der verknüpften Zustände genug Trainingsdaten zur Verfügung stehen.

Das oben beschriebene Verfahren ist in Abbildung 5.2 als Algorithmus dargestellt. Der Algorithmus besitzt bei einer naiven Implementierung für eine feste Zahl M von Endclustern kubische Komplexität, da in jedem Clusterschritt $\frac{n(n-1)}{2}$ Abstandsberechnungen notwendig sind. Speichert man die Abstände in einer $N \times N$-Matrix und aktualisiert nach jedem Clusterschritt nur die Abstände zu dem neu entstandenen Cluster, erhält man einen Algorithmus mit quadratischer Komplexität, allerdings zum Preis eines erhöhten Speicherbedarfs aufgrund der $N \times N$-Matrix, der ebenfalls quadratisch ist.

Tabelle 5.2: Bottom-Up-Clusteralgorithmus.

Erzeuge für jeden Triphonzustand ein Cluster
Bestimme das Clusterpaar (C_1, C_2) mit dem kleinsten Abstand
Verschmelze C_1 und C_2 zu C_0
Weiter, bis der Abstand für alle Paare (C_1, C_2) oberhalb einer Schwelle τ_{dist} liegt
Für alle Cluster C_1 mit Anzahl der Beobachtungen unterhalb einer Schwelle τ_{minobs}
Suche nächsten Nachbarn C_2 zu C_1
Verschmelze C_1 und C_2

5.2.3.2 Top-Down-Verfahren

Das Top-Down-Verfahren für das State-Tying startet mit einem Cluster, der alle Triphonzustände enthält. Dieser wird sukzessive aufgeteilt, indem in jedem Schritt für jeden Cluster die Aufteilung gewählt wird, für die das Abstandsmaß ΔLL besonders groß wird, d.h. für die die Log-Likelihood-Funktion möglichst stark wächst. Diese Aufteilungen werden so lange fortgeführt, bis entweder der Abstand für alle möglichen Aufteilungen kleiner wird als eine Schwelle τ_{dist} oder einer der entstehenden Cluster für alle möglichen Aufteilungen zu wenig Beobachtungen bezüglich einer Schwelle τ_{minobs} hat. D.h. für den Top-Down-Algorithmus entfällt die Notwendigkeit für eine zweite Phase zur Sicherstellung einer minimalen Anzahl von Beobachtungen pro verknüpftem Zustand. Problematisch ist dagegen bei diesem Verfahren, für einen Cluster die optimale Aufteilung zu finden. Eine naive Inplementierung, die über alle möglichen Aufteilungen optimiert, hätte wie schon erwähnt exponentielle Komplexität. Für eine typische Zahl von 10 000 Triphonzuständen wären für eine Aufteilung in zwei Mengen schon allein 10^{3000} Möglichkeiten zu überprüfen. In [Chou 91] wird diese Aufteilung mit Hilfe des LBG-Algorithmus berechnet, der lineare Komplexität besitzt, wobei gezeigt werden kann, daß die durch diesen Algorithmus gefundene Aufteilung optimal ist bzgl. Gleichung 5.4. Dieses Verfahren geht allerdings davon aus, daß die einzelnen Objekte, beim State-Tying also die Triphonzustände, frei zwischen beiden Clustern aufteilbar sind. In dieser Arbeit wird dagegen der Suchraum mit Hilfe von phonetischen Mengen, sog. *phonetischen Fragen*, eingeschränkt (siehe Abschnitt 5.3). D.h. die möglichen Aufteilungen der Triphonzustände sind durch die Menge der vorhandenen Fragen vorgegeben. Da diese Fragenmenge bezogen auf die mögliche Zahl der Aufteilungen klein ist, ist es hier möglich, alle zulässigen Aufteilungen zu bewerten.
Das oben beschriebene Verfahren ist als Algorithmus in Abbildung 5.3 dargestellt. Offensichtlich läßt sich der Ablauf eines Clustervorgangs mit diesem Verfahren als Baum darstellen, wobei der Ausgangscluster mit allen Triphonzuständen die Wurzel bildet, während die resultierenden verknüpften Triphonzustände an den Blättern stehen. Dies wird uns in Abschnitt 5.3 zu den phonetischen Entscheidungsbäumen führen, die diesen Ansatz des Top-Down-Clusterns mit den sogenannten phonetischen Fragen verbindet.

Tabelle 5.3: Top-Down-Clusteralgorithmus.

Bilde einen Cluster mit allen Triphonzuständen
Für alle Cluster C_0
Bestimme die Aufteilung in C_1 und C_2, für die nach dem Spalten • der Log-Likelihood-Gewinn maximal wird • die Anzahl der Beobachtungen für beide Cluster oberhalb einer Schwelle τ_{minobs} liegt
Falls der log-Likelihood-Gewinn über einer Schwelle τ_{dist} liegt, spalte den Cluster C_0 mit der Aufteilung auf
Weiter, bis kein Cluster mehr gespalten wird

5.2.4 Reestimierung der akustischen Parameter

Nach der Bestimmung der Verknüpfungen werden die Emissionsverteilungen der resultierenden verknüpften Zustände in einem akustischen Training neu geschätzt. Die Anzahl der Mischverteilungen richtet sich dabei nach der Anzahl der Cluster, die während der Verknüpfung der Zustände ermittelt worden sind. Weitere Einzelheiten zum akustischen Training finden sich in Kapitel 1.3.4 oder in [Ney 90].
Abbildung 5.2 stellt den Ablauf des State-Tying mit Bottom-Up- oder Top-Down-Algorithmus und die anschließende Reestimierung noch einmal dar. Mit $p(x, y)$ soll dabei eine bivariate Verteilungsdichtefunktion als Emissionsverteilung angenommen werden, wodurch eine bildliche Darstellung der Verfahren möglich wird.

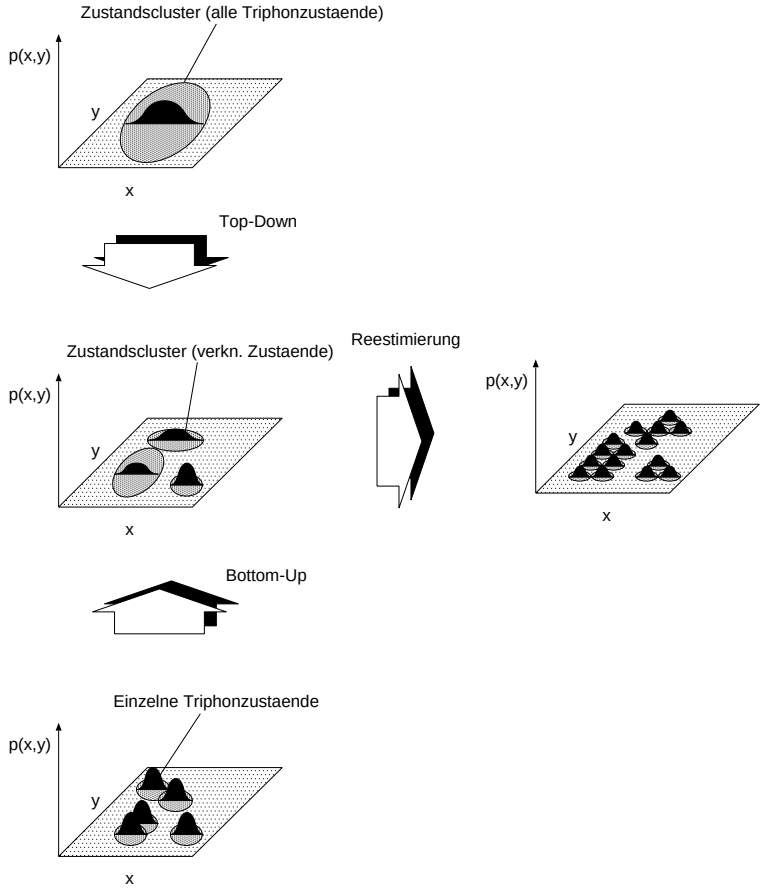


Abbildung 5.2: State-Tying mit Bottom-Up- und Top-Down-Algorithmus.

5.3 State-Tying mit phonetischen Entscheidungsbäumen

In Systemen für großen Wortschatz, die die Aussprache der Wörter im Trainings- und Erkennungslexikon mit Triphonen modellieren, sind im Allgemeinen das Trainings- und Erkennungsvokabular nicht deckungsgleich. D.h., oft kommen viele Wörter des Erkennungslexikons im Trainingskorpus gar nicht vor. Meist besteht dann das Problem, daß

auch einige der Triphone im Erkennungslexikon nicht im Training gesehen werden, so daß zunächst einmal keine akustischen Modelle für diese Triphone geschätzt werden können. Ein einfaches Verfahren, das diese Problematik vermeidet, ist das oben beschriebene *Monophon-Backing-Off* [Ney 90].
Da Monophone keine Koartikulationseffekte modellieren können, verschlechtern sie, verglichen mit entsprechend gut geschätzten Triphonen, das akustische Modell. Da aber für die alternativ zu schätzenden Triphone zu wenig oder gar keine Beobachtungen zur Verfügung stehen, führt die Verwendung von Monophon-Backing-Off bei entsprechender Wahl des Häufigkeits-Schwellwertes τ zu besseren Fehlerraten verglichen mit einem System, das ausschließlich Triphonmodelle verwendet.
Aufgrund der Schwäche des Monophon-Backing-Off bzgl. der Koartikulationsmodellierung wurden als Verbesserung dieser Modellierung verschiedene Verfahren entwickelt, die im Training nicht gesehenen Triphonen bessere akustische Modelle zuordnen sollen [Aubert *et al.* 96] [Moore *et al.* 94]. Die bei weitem wichtigste Methode in diesem Zusammenhang ist dabei die Kontextmodellierung mit phonetischen Entscheidungsbäumen [Young *et al.* 94] [Hwang 93]. Dazu wird zu jedem Phonemzustand ein Entscheidungsbaum erzeugt, der anhand von phonetischen Fragen einem beliebigen Triphonzustand eine passende Verteilung zuordnen kann. Dieses Verfahren soll im folgenden genauer betrachtet werden.

5.3.1 Phonetische Entscheidungsbäume

Ein phonetischer Entscheidungsbaum ist ein binärer Baum, an dessen inneren Knoten sogenannte phonetische Fragen stehen, während an den äußeren Knoten oder "Blättern" die Indizes der Mischverteilungen stehen. Abbildung 5.3 zeigt einen derartigen Entscheidungsbaum für das Phonem *th*, erstes Segment.

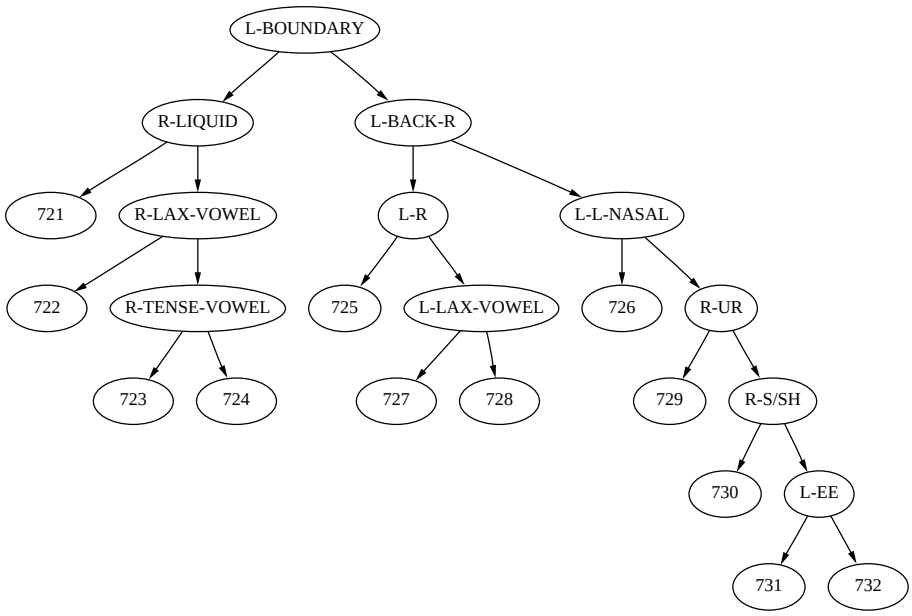


Abbildung 5.3: Entscheidungsbaum für das Phonem *th*, 1. Segment

Die phonetischen Fragen sind dabei von der Form "Ist der linke Kontext ein Vokal?" oder

"Ist der rechte Kontext ein Diphthong?". Eine Verzweigung nach links bedeutet eine Bejahung der Frage, eine Verzweigung nach rechts eine Verneinung. Um nun einem Triphonzustand mit diesem Entscheidungsbaum eine Emissionsverteilung zuzuweisen, beginnt man an der Wurzel. Mit der dort stehenden Frage wird der Triphonzustand, je nachdem ob die Antwort auf diese Frage "Ja" oder "Nein" ist, dem linken oder rechten Unterbaum der Wurzel zugeordnet. Die an der Wurzel des jeweiligen Unterbaums stehende Frage wird nun verwendet, um den Triphonzustand zu klassifizieren usw. Dies wird so oft wiederholt, bis ein Blatt erreicht wird. Der an diesem Blatt stehende Mischverteilungsindex wird zur Modellierung des Triphonzustandes verwendet (siehe Abbildung 5.4).

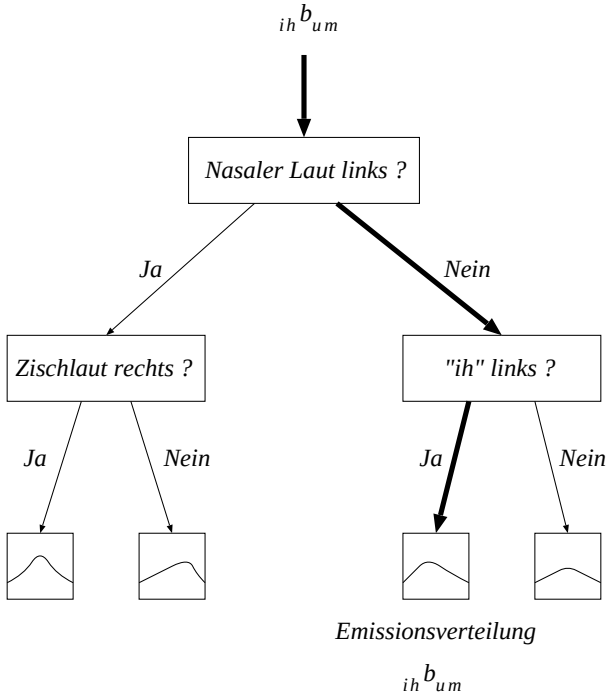


Abbildung 5.4: Zuordnung eines Triphonzustandes

Für jedes zentrale Phonem und jeden Segmentindex wird *ein* getrennter Entscheidungsbaum konstruiert. Soll zu einem Triphonzustand die optimale Mischverteilung gefunden werden, wird anhand des zentralen Phonems und des Segmentindexes der entsprechende Baum gewählt, der dann die Mischverteilung mit dem oben beschriebenen Verfahren liefert.

5.3.2 Phonetische Fragen

Die für einen solchen Entscheidungsbaum verwendeten phonetischen Fragen können folgendermaßen definiert werden: Eine phonetische Frage besteht aus

- einer Menge von Phonemen M
- und der Art des Kontextes c (links / rechts)

Eine Frage mit $M = \{ah, aa\}$ und $c =$ "links" bedeutet dann: "Ist das linke Kontextphonem des Triphonzustandes aus der Menge ah, aa?". Stünde diese Frage an einem inneren Knoten des Entscheidungsbaums, würden alle dort zu klassifizierenden Triphonzustände, deren linkes Kontextphonem ein *ah* oder ein *aa* ist, dem linken Unterbaum zugewiesen, alle anderen Triphonzustände dem rechten Unterbaum. Komplexere Fragen, die nach beiden Kontexten gleichzeitig fragen und dadurch, wenn man mehrere solcher Fragen kombiniert, alle möglichen Aufteilungen der Triphonzustände an einem Baumknoten ermöglichen, werden in dieser Arbeit nicht betrachtet. Entsprechende Untersuchungen finden sich z.B. in [Hwang 93]. Statt dessen werden Fragenkombinationen durch die Struktur des Entscheidungsbaums realisiert.

Eine weitere Vereinfachung ergibt sich dadurch, daß man für die an einem Knoten möglichen Fragen nur eine Untermenge aller möglichen Phonemmengen zuläßt. Diese Mengen werden dabei so gewählt, daß sie phonetischen Klassen entsprechen, wie sie von der Phonologie (z.B. [Ladefoged 82]) definiert werden. Eine phonetische Klasse ist eine Menge von Phonemen, die eine bestimmte, von der Phonologie als "unterscheidend" angenommene Eigenschaft haben. Diese wird in der Phonologie als "kleinste bedeutungsunterscheidende Lauteinheit" definiert. Die Menge von Phonemen wird somit durch sogenannte *Minimalpaare* definiert. Minimalpaare sind Worte, die unterschiedliche Bedeutung haben *und* sich nur in einem einzigen Lautsegment unterscheiden. Ein Beispiel für ein solches Minimalpaar ist "Rasen" und "Riesen".

Je nach Grad der Unterscheidung können so 20-50 Phonemklassen gebildet werden, die dann als mögliche phonetische Fragen beim Aufbau des Entscheidungsbaums verwendet werden. Motiviert wird diese Vorgehensweise dadurch, daß der Entscheidungsbaum in der Lage sein soll, im Training nicht gesehenen Triphonzuständen Emissionsverteilungen zuzuordnen. Könnten an den inneren Knoten beliebige phonetische Fragen verwendet werden, wäre der Freiheitsgrad bei der Aufteilung der Triphonzustände des Trainingskorpus zwar größer. Dafür würden die Entscheidungen über die Aufteilung der Zustände, die an den Knoten getroffen werden, spezifisch für die Trainingsdaten sein, da die zu verwendende Frage rein datengetrieben auf dem Trainingskorpus ermittelt werden würde. Bei der Verwendung von phonetischen Klassen als Fragen geht statt dessen auch Expertenwissen über die Bildung der Laute ein, was gerade bei der Modellierung von Auswirkungen des phonetischen Kontextes auf die Bildung eines Lautes relevant ist. D.h. ein mit phonetischen Klassen als Fragen konstruierter Baum sollte besser generalisieren, also im Training nicht gesehenen Triphonzuständen Mischverteilungen zuordnen, die den tatsächlichen Emissionsverteilungen ähnlicher sind.

Problematisch bei der Verwendung von phonetischen Klassen als Fragen ist, daß zur Definition der phonetischen Klassen Expertenwissen über die Lautbildung benötigt wird. D.h. will man für einen neuen Korpus Entscheidungsbäume verwenden, muß man zunächst von Hand phonetische Fragen definieren. Dies ist relativ zeitaufwendig und, vor allem für Nichtexperten, auch fehleranfällig. In Kapitel 7 wird daher ein automatisches Verfahren beschrieben, mit dem man solche phonetischen Fragen, die auch für nicht gesehene Triphonzustände eine vernünftige Zuordnung von Emissionsverteilungen erlauben, erzeugen kann.

5.3.3 Konstruktion des Baums

Der Entscheidungsbaum für ein Phonem und ein Segment wird mit dem in Abschnitt 5.2.3.2 beschriebenen Top-Down-Verfahren konstruiert. Das im Fall des für das Spracherkennungssystem im Training zu optimierende Kriterium, die Log-Likelihood über die Menge der Trainingsdaten, impliziert, als Kriterium für die optimale Aufteilung eines Knoten die Log-Likelihood-Differenz nach Formel 5.7 zu verwenden. Die Menge der möglichen Aufteilungen eines Knotens s wird durch die Menge der zur Verfügung stehenden Fragen bestimmt. Für Triphone kann jede Frage für zwei verschiedene Kontexte gestellt werden. Damit ergeben sich für n Fragen maximal $2n$ Aufteilungen pro Knoten. Über diese Aufteilungen wird für jedes Blatt des Baums optimiert, die Frage mit der größten Log-Likelihood-Verbesserung wird dann zur Aufteilung des jeweiligen Blatts verwendet wenn

- die Verbesserung ΔLL über einem Schwellwert τ_{dist} liegt und
- die Anzahl an Beobachtungen für beide resultierenden Knoten über einem Schwellwert τ_{minobs} liegt.

τ_{minobs} soll verhindern, daß Blätter entstehen, die im späteren akustischen Training nicht mehr robust genug geschätzt werden können. τ_{dist} begrenzt die Tiefe des Baums und damit die Menge der durch die Aufteilungen entstehenden Blätter auf eine vernünftige Anzahl, da die Verbesserung der Log-Likelihood durch das Aufspalten der Blätter mit zunehmender Größe des Baums annähernd monoton abnimmt.

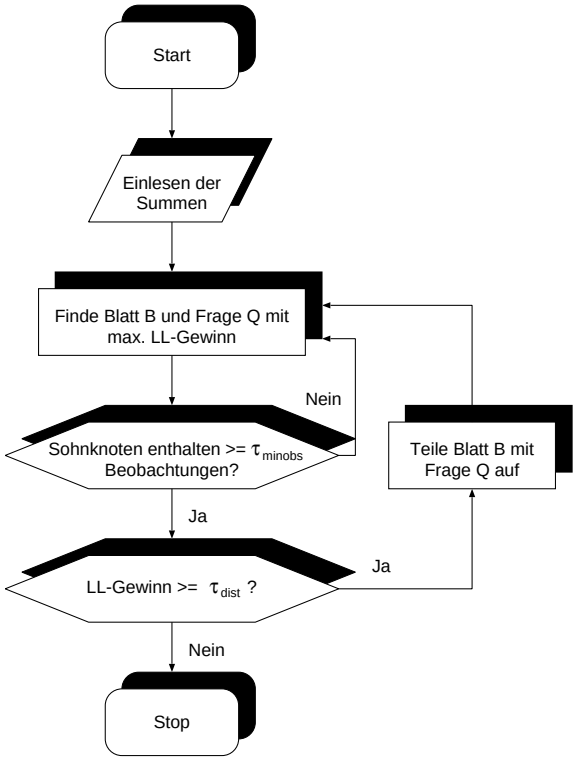


Abbildung 5.5: Konstruktion eines phonetischen Entscheidungsbaums

Abbildung 5.5 zeigt noch einmal den grundsätzlichen Ablauf der Konstruktion eines phonetischen Entscheidungsbaums. Zunächst werden die Summen und Quadratsummen (siehe Gleichung 5.1) zur Berechnung der Parameter der Gaußverteilungen, welche die Beobachtungen innerhalb eines Knotens modelliert, eingelesen. Danach wird das Blatt B und die Frage Q bestimmt, die die maximale Log-Likelihood-Verbesserung über alle Blätter und phonetischen Fragen ergibt. Falls diese Aufspaltung dazu führt, daß die Anzahl der Beobachtungen für einen der beiden Tochterknoten unter die Schwelle τ_{minobs} fällt, wird diese Aufteilung verworfen und eine neue gesucht. Andernfalls wird für diese Aufteilung noch überprüft, ob der Log-Likelihood-Gewinn oberhalb der Schwelle τ_{dist} liegt. Ist dies ebenfalls der Fall, wird das Blatt mit dieser Frage aufgespalten und danach eine neue Aufteilung gesucht. Findet der Algorithmus keine Aufteilung, welche die genannten Kriterien erfüllt, wird die Aufteilung der Blätter abgebrochen. Mit diesem Verfahren wird für jedes zentrale Phonem und jedes Segment ein phonetischer Entscheidungsbaum generiert. Die Erzeugung läuft sequentiell ab, d.h. ein Entscheidungsbaum wird komplett aufgebaut, bevor der nächste Baum begonnen wird.
Im aktuellen am Institut verwendeten System wird inzwischen eine modifizierte Vorgehensweise gewählt. Statt pro Phonemsegment einen Baum zu erzeugen, wird nur *ein einziger* Entscheidungsbaum verwendet [Paul 97] [Beulen *et al.* 97], dessen Erzeugung in drei Phasen abläuft. Zu Beginn der Erzeugung befinden sich *alle* Triphonzustände an der Wurzel des Baums. In der ersten Phase werden nur Fragen an das *zentrale Phonem* des Triphozustandes gestellt, bis die Triphonzustände komplett nach zentralem Phonem getrennt sind. In der zweiten Phase werden nur Fragen nach dem *Segment* des Zustandes gestellt, bis die Triphonzustände auch nach Segment getrennt sind. In der dritten Phase schließlich werden Fragen nach dem Kontext des Zustandes gestellt. Diese Vorgehensweise hat zwei Vorteile:

1. Es können leicht andere Baumstrukturen verwendet werden, beispielsweise Bäume, bei denen Fragen nach zentralem Phonem, Segment und Phonemkontext in beliebiger Reihenfolge verwendet werden (siehe Abschnitt 5.3.5)

2. Indem quasi alle Bäume parallel erzeugt werden, kann die Anzahl der entstehenden Blätter relativ einfach gesteuert werden, indem einfach die n besten Aufteilungen ausgeführt werden. Man erhält dadurch $n+1$ Blätter. Dies ist ein Vorteil gegenüber den in der Literatur beschriebenen Verfahren, bei denen die Zahl der Blatter nur indirekt durch die Schwellwerte τ_{dist} und τ_{minobs} gesteuert werden kann.

Dem gegenüber steht ein höherer Aufwand sowohl bei der Erzeugung des Baums als auch bei der Klassifikation eines Triphonzustandes, da

- beim Erzeugen des Baums zusätzlich k Aufteilungen nötig sind, wobei k der Zahl der Phonemsegmente entspricht, und

- bei der Klassifikation eines Triphonsegments dieser zusätzlich erzeugte Baum durchlaufen werden muß.

Da dieser Aufwand aber nur einmal pro Baumgenerierung bzw. einmal pro Initialisierung des Erkenners anfällt, ist dieser Mehraufwand vernachlässigbar (siehe auch Abbildung 5.6).

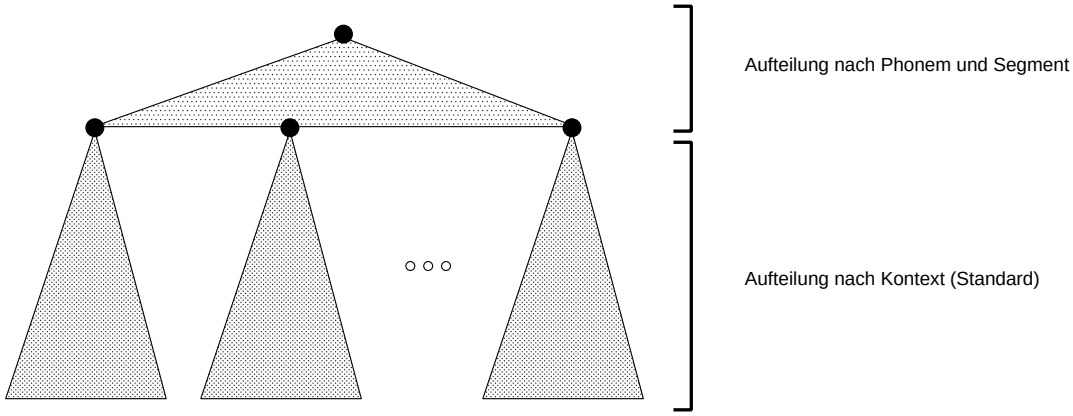


Abbildung 5.6: Modifizierter Aufbau des Entscheidungsbaums

Die für die Berechnung der jeweils besten Aufteilung für einen Baum mit N Blättern nötigen N Berechnung des Log-Likelihood-Gewinns für die möglichen Aufteilungen würde die Komplexität für die Baumerzeugung von $O(N)$ auf $O(N^2)$ steigen lassen. Durch Speicherung der möglichen Aufteilungen in jedem Schritt läßt sich diese Komplexität aber wieder auf $O(N)$ reduzieren.

5.3.4 Vor- und Nachteile von Entscheidungsbäumen

Der entscheidende Vorteil eines Entscheidungsbaums verglichen mit rein datengetriebenen Verfahren besteht in seiner Verallgemeinerungsfähigkeit. D.h. ein Entscheidungsbaum ist in der Lage, jedem beliebigen Triphonzustand ein akustisches Modell zuzuordnen. Dadurch kann die Verwendung von Backing-Off-Modellen, die meist eine relativ ungenaue Annäherung an die tatsächliche Verteilung der Zustände darstellen, vermieden werden (siehe oben). Insbesondere bei der Verwendung von wortübergreifenden Triphonen (siehe auch Kapitel 6), bei denen die Zahl der im Trainingskorpus nicht gesehenen Triphone größer ist als bei den Standardmodellen, ist die Verwendung von phonetischen Entscheidungsbäumen in Verbindung mit State-Tying sinnvoll.

Nachteilig bei phonetischen Entscheidungsbäumen ist zum einen der geringere Freiheitsgrad des Algorithmus, die Triphonzustände in den Knoten aufzuteilen, verglichen mit einem rein datengetriebenen Verfahren. Diese, durch die begrenzte Anzahl der phonetischen Fragen begründete Tatsache, ist im Hinblick auf die Verallgemeinerungsfähigkeit des Baums natürlich wünschenswert. Allerdings bedeutet dies auch, daß bei einem Korpus, in dem für alle Triphonzustände genügend Trainingsmaterial zur Verfügung steht, und bei dem keine nicht gesehenen Triphone im Erkennungsvokabular vorkommen, ein Entscheidungsbaum verglichen mit datengetriebenen Verfahren in der Regel zu schlechteren Fehlerraten führt (siehe z.B. [Hon 92]). Ein weiteres Problem ist, wie oben schon angesprochen, die Definition von sinnvollen phonetischen Fragen. Wie in Kapitel 7 gezeigt wird, können durch geeignete rein datengetriebene Algorithmen solche phonetischen Fragen automatisch erzeugt werden.

5.3.5 Erweiterungen des Basisverfahrens

Obwohl bereits das oben beschriebene Basisverfahren des State-Tying mit phonetischen Entscheidungsbäumen gute Ergebnisse auf verschiedenen Testkorpora liefert, sollte in dieser Arbeit durch verschiedene z.T. in der Literatur beschriebene Erweiterungen dieses Verfahren weiter verbessert werden. Dabei war das Kriterium zur Aufteilung der Knoten selbst nicht Ziel der Optimierung, da in verschiedenen Publikationen schon die Äquivalenz verschiedener Kriterien bzgl. der Qualität des Entscheidungsbaums beschrieben wird (siehe z.B. [Breiman *et al.* 84] [Kramer 96]). Statt dessen wurde versucht, durch mehr Freiheitsgrade bei der Struktur des Baums bzw. durch verbesserte Modellierung der Beobachtungen an den Baumknoten die Modellierung durch den Entscheidungsbaum zu verbessern. Im einzelnen wurden folgende Modifikationen getestet:

- Verwendung von geschlechtsabhängigen Modellen in den Knoten [Odell 95],
- Glättung der Modellparameter an den Knoten mit allgemeineren Modellen [Hon 92],
- zusätzliches Verschmelzen von Blättern nach dem Aufteilen der Knoten des Entscheidungsbaums [Odell 95] [Young *et al.* 94] [Lazarides *et al.* 96],
- Verwendung einer vollen statt einer diagonalen Kovarianzmatrix für die Gaußverteilungen an den Baumknoten.

Für die in der Literatur vorgeschlagenen ersten drei Methoden gab es bisher nur allgemeine Aussagen über deren Leistungsfähigkeit, nicht aber über den konkreten Gewinn bzgl. Erkennungsgenauigkeit. Die Verwendung einer vollen statt einer diagonalen Kovarianzmatrix wird als neues Verfahren vorgeschlagen, um die Approximation der Mischverteilungen durch die Gaußverteilungen an den Baumknoten zu verbessern.
Diese Erweiterungen sollen im folgenden beschrieben werden.

5.3.5.1 Verwendung von geschlechtsabhängigen Modellen

Hintergrund für die Verwendung von geschlechtsabhängigen Modellen für das State-Tying ist, daß akustische Modelle für Männer und Frauen im Allgemeinen recht unterschiedlich sind, da sich männliche und weibliche Stimmen insbesondere in ihrer Grundfrequenz stark unterscheiden. Dies kann entweder mit Mischverteilungen modelliert werden, wobei in [Aubert *et al.* 94] gezeigt wird, daß sowohl mit geschlechtsabhängigen als auch mit geschlechtsunabhängigen Modellen dieselben Ergebnisse erzielt werden. Zum anderen kann durch eine sogenannte *Vokaltraktlängennormalisierung* (VTLN) [Welling 98] die Vokaltraktlänge aus dem Sprachsignal herausgerechnet werden, wodurch auch die Geschlechtsabhängigkeit des Sprachsignals weitgehend verschwindet. Da bei der Erzeugung des Baums zur Modellierung der akustischen Beobachtungen an den Baumknoten aber nur Einzelverteilungen und nicht normierte Sprachdaten verwendet werden, könnte es von Vorteil sein, statt einer Gaußverteilung pro Knoten zwei geschlechtsspezifische Gaußverteilungen zu verwenden [Odell 95]. Bei der Erzeugung des Baums werden dann zwei Mengen geschlechtsabhängiger Triphonzustände in den Knoten mit je einer Gaußverteilung modelliert. Die phonetischen Fragen werden bei einer Aufteilung eines Knotens auf beide Mengen gleichermaßen angewendet, der Log-Likelihood-Gewinn pro Aufteilung ergibt sich aus der Summe der Gewinne für beide Zustandsmengen.

5.3.5.2 Verwendung eines einzigen Baums

Beim Basisverfahren werden, wie oben beschrieben, die Triphonzustände nach zentralem Phonem und Segment aufgeteilt und dann für jede dieser Teilmengen ein Entscheidungsbaum berechnet. Nun kann man argumentieren, daß diese Voreinteilung u.U. nicht optimal ist, da z.B. zwei bestimmte Phoneme oder Segment 1 und Segment 2 eines Phonems akustisch sehr ähnlich sind. Daher könnte es vorteilhaft sein, dem Algorithmus zur Konstruktion des Entscheidungsbaums zu überlassen, ob er die Triphonzustände nach zentralem Phonem, Segment oder Kontextphonem trennt [Paul 97]. Eine einfache Möglichkeit, dieses zu erreichen, ist, neben den Fragen nach dem Kontextphonem auch Fragen nach dem zentralen Phonem ("Ist das zentrale Phonem ein *ah*?") oder dem Segment ("Hat das Segment die Nummer 2?") zuzulassen. Dies bedingt natürlich, daß jetzt nicht mehr ein Baum für jedes Phonem und jedes Segment verwendet werden (Abbildung 5.7, "Phonem und Zustand gleich"), sondern nur ein einziger Baum, der dann auf alle Triphonzustände angewendet wird (Abbildung 5.7, "keine Einschränkung", Abbildung 5.6).

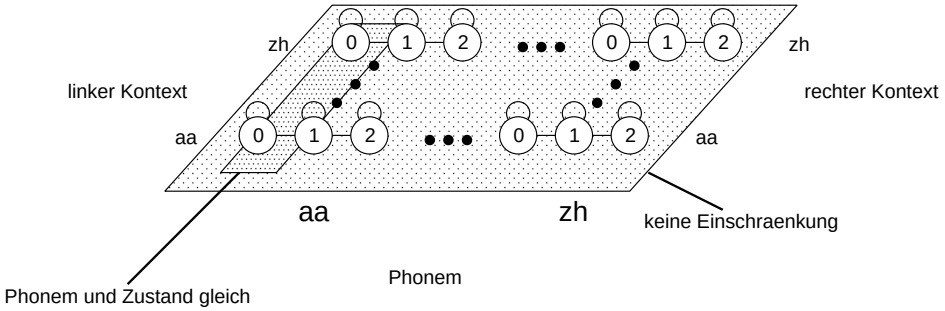


Abbildung 5.7: Verschiedene Voreinteilungen der Triphonzustände bei der Baumkonstruktion

Dieses Verfahren wird, wie oben beschrieben, auch zur Erzeugung eines Baums mit der üblichen Voreinteilung der Triphonzustände (zentrales Phonem und Segment) verwendet.

5.3.5.3 Zusammenfassen von Knoten

Obwohl durch die bei der Konstruktion des Entscheidungsbaums verwendete Menge von Fragen schon eine relativ große Freiheit bei der Aufteilung der Knoten gegeben ist, führt die eingeschränkte Menge der möglichen Aufteilungen doch dazu, daß die Modellierung der Daten durch die an den Blättern stehenden Gaußverteilungen nicht optimal bzgl. Log-Likelihood ist. Außerdem führt das Top-Down-Konstruktionsprinzip des Baums dazu, daß Blätter entstehen, deren Modelle akustisch relativ ähnlich sind. Diese potentiellen Probleme lassen sich beseitigen, indem nicht nur Aufteilungen, sondern auch Zusammenfassungen von Knoten erlaubt werden. Dabei wird die Auswahl, welche Knoten zusammengefaßt werden, ebenfalls durch das Log-Likelihood-Kriterium getroffen, indem die Knoten verschmolzen werden, bei der die Verschlechterung der Log-Likelihood möglichst gering ist. Des weiteren kann das verschmelzen entweder erfolgen, nachdem das Aufteilen der Knoten komplett abgeschlossen ist, oder es wird gleichzeitig mit dem Aufteilen durchgeführt. In dieser Arbeit soll allerdings nur auf das erste Verfahren eingegangen werden, für weitergehende Betrachtungen zu dem anderen Verfahren kann z. B. auf [Bransch 95] verwiesen werden.

Das hier untersuchte Verfahren läuft in zwei Phasen ab: in der ersten Phase wird mit dem Top-Down-Verfahren ein Entscheidungsbaum mit einer bestimmten Menge von Blättern erzeugt. In der zweiten Phase werden diese Blätter mit Hilfe des Bottom-Up-Verfahrens solange zusammengefaßt, bis eine bestimmte reduzierte Anzahl von Blättern erreicht ist (siehe auch Abbildung 5.8).

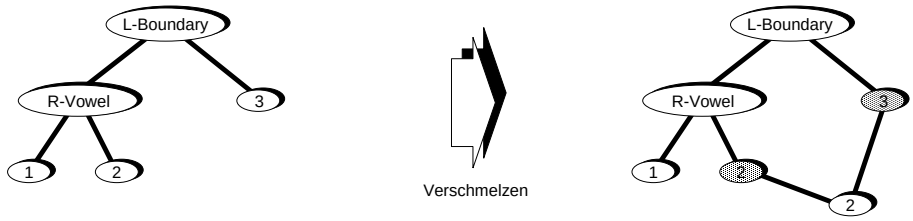


Abbildung 5.8: Verschmelzen von Blättern

5.3.5.4 Reduzierte Triphonliste

In [Hon 92] wird beschrieben, daß es vorteilhaft sein kann, nur eine Auswahl der im Trainingskorpus vorkommenden Triphone zur Konstruktion des Entscheidungsbaums zu verwenden. Es wurden nur die Triphone des Trainingskorpus verwendet, die auch im Erkennungsvokabular vorkommen. Dies soll verhindern, daß im Entscheidungsbaum Strukturen für nicht im Erkennungsvokabular vorkommende Triphone vorgesehen werden.

In dieser Arbeit wurden zwei Arten von reduzierten Triphonlisten verwendet. Zum einen wurde wie in [Hon 92] die Triphonmenge aufgrund der im Erkennungskorpus vorkommenden Triphone beschränkt. Zum anderen wurde die Triphonmenge auf die Triphone beschränkt, die mindestens mit einer Häufigkeit τ im Trainingskorpus vorkommen. Z.B. werden für $\tau = 20$ nur die Triphone zur Konstruktion des Entscheidungsbaums verwendet, die 20 mal oder öfter im Trainingskorpus gesehen werden.

5.3.5.5 Volle Kovarianzmatrix

Die zu Modellierung der Beobachtungen an einem Knoten verwendete Gaußverteilung besitzt für das Basisverfahren lediglich eine diagonale Kovarianzmatrix. Dies ist eine relativ schlechte Annäherung an die Mischverteilungen, die in der Erkennung zu Modellierung der Emissionsverteilungen verwendet werden. Das kann natürlich auch dazu führen, daß die bei der Konstruktion des Baums durchgeführten Aufteilungen der Knoten nicht optimal für die in der Erkennung verwendeten akustischen Modelle sind. Eine bessere Approximation einer Mischverteilung kann durch die Verwendung einer vollen Kovarianzmatrix erreicht werden (siehe Abbildung 5.9). Allerdings müssen dann pro Knoten statt $2D$ Parameter $D + D(D+1)/2$ Parameter geschätzt werden. Dies macht die Verwendung von Glättungsverfahren für die Parameter der Kovarianzmatrix erforderlich, wie sich in den durchgeführten Tests zeigen wird (siehe Anhänge A und B).

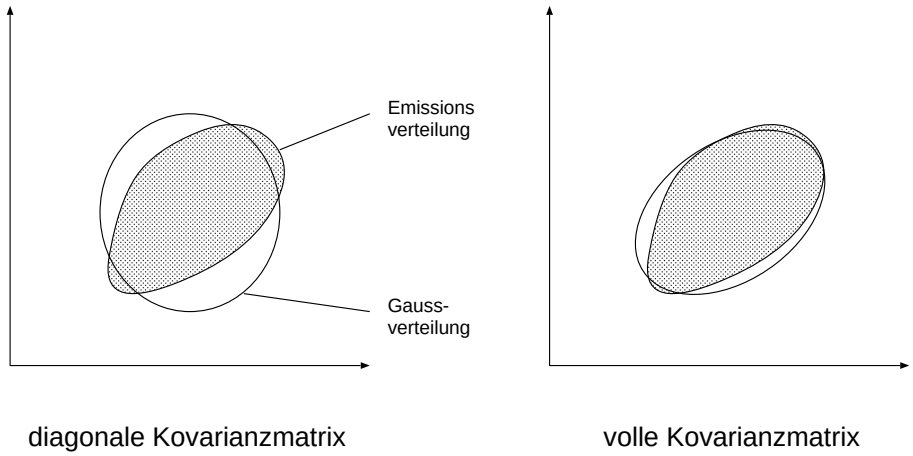


Abbildung 5.9: Approximation einer Emissionsverteilung durch eine Gaußverteilung mit diagonaler oder voller Kovarianzmatrix

5.4 Ergebnisse

Für die oben betrachteten Verfahren und Erweiterungen wurden Erkennungstests auf dem Spracherkennungskorpus *Wall Street Journal* (WSJ-5k) durchgeführt. Als akustische Merkmale wurden die in Abschnitt 1.2 beschriebenen Filterbankmerkmale verwendet. Die für die Emissionsverteilungen verwendeten Mischverteilungskomponenten waren Laplace-Verteilungen mit gepooltem Standardabweichungsvektor. Diese führen für großes Vokabular zu vergleichbaren Fehlerraten wie Gauß-Verteilungen [Welling 98]:

Tabelle 5.4: Vergleich der Fehlerraten des RWTH–Systems mit LDA für Gauß– und Laplace–Verteilungen auf dem WSJ-5k-Korpus mit Bigramm–Sprachmodell und geschlechtsabhängigen Referenzen.

Verteilungen	Dichten	Zustände (k)	DEL – INS [%]	WER [%]
Laplace	75k+86k	3k	1.3 – 0.7	7.1
Gauß	68k+82k	8k	1.3 – 0.8	6.9

Die Fehlerrate für Gauß-Verteilungen ist nur geringfügig besser als die für Laplace-Verteilungen, wobei dies auch auf den größeren Suchraum beim Test mit Gauß-Verteilungen ("Zustände") zurückgeführt werden könnte. Die Diskrepanzen mit den im folgenden genannten Fehlerraten ergeben sich aus der Tatsache, daß für den obigen Test cepstrale Merkmale verwendet wurden, während für die folgenden Tests (noch) Filterbank-Merkmale eingesetzt werden mußten.
Als Sprachmodell wurde ein von D. Paul, MIT, zur Verfügung gestelltes Bigramm-Sprachmodell mit einer Perplexität $PP = 110$ verwendet.
Die weiterhin in den Tabellen enthaltenen Informationen sind:

- Anzahl der Zustände vor/nach dem State-Tying

  Diese Zahlen geben an, mit welcher Zahl vom HMM-Zuständen das State-Tying durchgeführt worden ist bzw. welche Zahl von Zustandsclustern nach dem State-

Tying vorhanden sind. Aus diesen beiden Zahlen läßt sich somit die Reduktion der Zahl der zu schätzenden Mischverteilungen des Systems direkt ablesen.

- mittlere LL vor/nach dem State-Tying

  Diese Zahlen geben an, wie hoch die mittlere Log-Likelihood pro Beobachtungsvektor auf den Trainingsdaten vor bzw. nach dem Clustervorgang beim Erzeugen des Entscheidungsbaums ist. D.h. vor dem Clustervorgang werden die Trainingsdaten des Korpus (ohne Pause) durch 129 Gaußverteilung modelliert (43 Phoneme und drei Segmente pro Phonem), nach dem Clustervorgang werden diese Daten durch die n Gaußverteilungen an den Blättern des Entscheidungsbaums modelliert.

- Dichten

  Diese Zahl gibt die Gesamtzahl der Mischverteilungskomponenten des akustischen Modells an. Die Parameter einer Mischverteilungskomponente sind der Mittelwertsvektor mit 35 Komponenten plus einem Mischverteilungsgewicht.

- Einfügungen und Auslassungen (DEL-INS)

  Die Zahl der Einfügungen bzw Auslassungen gibt an, wieviele Fehler bei der Erkennung auf Einfügung eines Wortes bzw. Auslassung eines Wortes zurückzuführen sind. Sind diese Zahlen stark unterschiedlich, deutet das auf einen schlecht eingestellten Skalierungsfaktor für das Sprachmodell hin. Angestrebt werden sollte ein Verhältnis von 1:1 bis 2:1 zugunsten der Auslassungen.

- WER

  Die *Wortfehlerrate* (WER) gibt die prozentuale Anzahl der Wortfehler bezogen auf die Zahl der Wörter des Erkennungskorpus an. Als Fehler werden Einfügungen, Auslassungen und Verwechslungen gezählt. Verwechslungen entstehen durch das Austauschen eines Wortes des gesprochenen Satzes durch ein anderes, z.B. wegen akustischer Ähnlichkeit ("sein" statt "ein"). Diese Zählung der Fehler eines erkannten Satzes wird auch *Levenshtein-Distanz* zwischen gesprochenem und erkanntem Satz genannt. Sie kann effizient durch dynamische Programmierung bestimmt werden.

Im folgenden sollen nun die Erkennungsergebnisse auf dem oben genannten Erkennungskorpus vorgestellt werden. Die für die Beurteilung des State-Tying mit Entscheidungsbäumen wichtige Anzahl der im Training nicht gesehenen Triphone, die im Erkennungs*korpus* auftreten, findet man in Tabelle 5.5. Ein Anteil von 3.7% bedeutet dabei, daß für 3.7% der akustischen Vektoren des Erkennungskorpus keine Triphonmodelle im Training geschätzt werden können.

5.4.1 Tabellen

Die Ergebnisse auf dem WSJ-5k-Korpus sind in den Tabellen 5.6 bis 5.14 dargestellt.
Durch das datengetriebene State-Tying mit Bottom-Up-Clustering wird die Wortfehlerrate von 8.8% auf 8.1% verglichen mit nicht verknüpften Modellen gesenkt. Durch die Verwendung des Verfahrens mit Entscheidungsbäumen kann diese Fehlerrate noch einmal

Tabelle 5.5: Anteil der ungesehenen Triphone bei der Erkennung.

Test-umgebung	Triphone in der Erkennung	davon im Training nicht gesehen	relativer Anteil [%]
WSJ Nov '92 (5k)	50915	1861	3.7

Tabelle 5.6: Wortfehlerraten [%] für State-Tying mit phonetischen Entscheidungsbäumen auf WSJ-5k.

Tying	Zustände vor	Zustände nach	Dichten (k)	DEL-INS [%]	WER [%]
kein Tying	2338	2338	246	–	8.8
datengetrieben	5698	2005	168	1.2-1.0	8.1
Entscheidungsbaum	23502	2001	192	1.2-0.7	7.6

auf 7.6% gesenkt werden. Dieses Ergebnis wurde unter Verwendung der vollen Triphonliste erzielt (siehe Tabelle 5.6).
Tabelle 5.7 enthält dazu die Vergleichsergebnisse anderer Gruppen. Die dort verwendeten Korpora sind allerdings nicht identisch mit dem von uns verwendeten Korpus, lediglich die Vokabulargröße und die Aufgabenstellung (klare Sprache, geschlossenes Vokabular) sind dieselben (siehe [Young *et al.* 94] [Beyerlein *et al.* 97] [Hon 92]). Die Fehlerrate für ein System ohne phonetische Entscheidungsbäume im Falle *Philips Forschungslab.* ist nur geschätzt, da in [Aubert *et al.* 96] lediglich Ergebnisse für ein System ohne State-Tying bzw. mit Bottom-Up-Tying und in [Beyerlein *et al.* 97] nur Ergebnisse für ein System mit Bottom-Up-Tying bzw. Tying mit phonetischen Entscheidungsbäumen angegeben sind (für verschiedene Korpora). Rechnet man mit den Ergebnissen von [Aubert *et al.* 96] die Bottom-Up-Tying-Ergebnisse von [Beyerlein *et al.* 97] auf ein System ohne State-Tying zurück, erhält man das in Tabelle 5.7 angegebenen Resultat.
Wie man sieht, liegen die durch andere Gruppen erzielten Verbesserungen durch State-Tying mit phonetischen Entscheidungsbäumen im Rahmen dessen, was auch in dieser Arbeit erzielt wurde.

Tabelle 5.7: Vergleichsergebnisse anderer Gruppen für State-Tying mit phonetischen Entscheidungsbäumen.

Gruppe	Korpus	WER [%] Kein Tying	WER [%] Tying
Cambridge University	RM (Feb. 91)	3.8	3.7
	RM (Sep. 92)	8.1	7.3
Philips Forschungslab.	WSJ (92 + 93)	9.5 (est.)	9.0
CMU	TIRM	6.0	6.2
	VI-2000	6.5	5.4
diese Arbeit	WSJ-5k	8.8	7.6

Tabelle 5.8: Wortfehlerraten [%] für State-Tying mit phonetischen Entscheidungsbäumen mit verschiedenen Anzahlen an Mischverteilungen auf WSJ-5k.

Zustände		mittlere LL		Dichten	DEL-INS	WER
vor	nach	vor	nach	(k)	[%]	[%]
23502	1501	219.73	212.10	192	1.3-0.7	7.9
23502	2001	219.73	211.85	160	1.2-0.7	7.6
23502	2501	219.73	211.67	216	1.3-0.9	8.1

Tabelle 5.9: Wortfehlerraten [%] für State-Tying mit phonetischen Entscheidungsbäumen mit verschiedenem τ_{minobs} auf WSJ-5k.

τ_{minobs}	Zustände		mittlere LL		Dichten	DEL-INS	WER
	vor	nach	vor	nach	(k)	[%]	[%]
100	23502	2001	219.73	211.82	188	1.3-0.7	8.1
200	23502	2001	219.73	211.82	190	1.2-0.7	7.9
500	23502	2001	219.73	211.85	192	1.2-0.7	7.6
1000	23502	2001	219.73	212.02	208	1.3-0.7	7.9

Tabelle 5.8 enthält die Ergebnisse für State-Tying mit phonetischen Entscheidungsbäumen mit verschiedenen Zahlen von Blättern, die der Zahl der im Spracherkennungssystem verwendeten Mischverteilungen entsprechen. Ein Optimum findet sich hier bei ca. 2000 Blättern.
Tabelle 5.9 zeigt die Ergebnisse für verschiedene Werte des Parameters τ_{minobs}, der die minimale Zahl von Beobachtungen pro Blatt des Entscheidungsbaums und damit pro Mischverteilung des Spracherkennungssystems festlegt. Hier findet sich ein Optimum bei $\tau_{minobs} = 500$. Dieser Wert wurde für die folgenden Tests standardmäßig verwendet.
Durch die in 5.3.5.4 beschriebene Reduktion der Triphonliste steigt die Wortfehlerrate wieder leicht auf 7.8% (siehe Tabelle 5.10). Dabei ist ein direkter Zusammenhang zwischen der Zahl der verwendeten Triphone und der erzielten Fehlerrate zu beobachten. Je mehr Triphone zur Konstruktion des Entscheidungsbaums verwendet werden, desto geringer ist die Fehlerrate auf den Testdaten. Ein gegenteiliger Effekt ist bzgl. der Log-Likelihood auf den Trainingsdaten zu beobachten. Die Verbesserung der Log-Likelihood durch die Aufteilung der Blätter des Entscheidungsbaums ist besser, je weniger Triphone verwendet

Tabelle 5.10: Wortfehlerraten [%] für State-Tying mit phonetischen Entscheidungsbäumen mit verschiedenen Triphonlisten auf WSJ-5k.

Anzahl	Zustände		mittlere LL		Dichten	DEL-INS	WER
Beobachtungen	vor	nach	vor	nach	(k)	[%]	[%]
1	23502	2001	219.73	211.85	192	1.2-0.7	7.6
20	9075	2001	219.72	211.76	194	1.3-0.7	7.9
50	5499	2001	219.72	211.65	196	1.2-0.8	8.3
Schnitt	15660	2001	219.71	211.82	194	1.2-0.7	7.8

Tabelle 5.11: Wortfehlerraten [%] für State-Tying mit phonetischen Entscheidungsbäumen und der Verwendung *eines* Baums auf WSJ-5k.

Tying	Zustände		mittlere LL		Dichten	DEL-INS	WER
	vor	nach	vor	nach	(k)	[%]	[%]
129 Bäume	23502	2001	219.73	211.85	192	1.2-0.7	7.6
ein Baum	23502	2001	219.72	211.89	192	1.3-0.7	7.8

Tabelle 5.12: Wortfehlerraten [%] für State-Tying mit phonetischen Entscheidungsbäumen und Verschmelzen der Blätter auf WSJ-5k.

Verschmelzen	Zustände		mittlere LL		Dichten	DEL-INS	WER
	vor	nach	vor	nach	(k)	[%]	[%]
nein	23502	2001	219.73	211.85	192	1.2-0.7	7.6
ja (3000)	23502	2001	219.73	211.79	200	1.3-0.7	7.8
ja (2000)	23502	1714	219.73	211.93	176	1.3-0.7	8.0
ja (2000)	23502	1412	219.73	212.08	154	1.4-0.6	8.0
ja (2000)	23502	1130	219.73	212.29	130	1.5-0.6	8.3
ja (2000)	23502	880	219.73	212.55	102	1.5-0.5	8.4

werden.

Diese zunächst gegensätzlichen Beobachtungen lassen sich damit erklären, daß bei einer Vergrößerung der Menge der Triphone bei der Konstruktion des Entscheidungsbaums zwar die Modellierung der gesamten Triphonmenge schlechter wird, da eine größere Menge von Beobachtungen durch dieselbe Zahl von Parametern modelliert wird. Andererseits sind bei der Konstruktion des Entscheidungsbaums nun wesentlich mehr Triphone verwendet worden, so daß die Menge der vom Entscheidungsbaum *nicht* gesehenen Triphone stark reduziert ist *und* mehr Trainingsmaterial (Triphone) zur Verfügung steht. Die Bedeutung einer möglichst großen Menge von Trainingsmaterial wird deutlich an der Verschlechterung der Fehlerrate auf den Testdaten beim Test "Schnitt", bei dem nur die Triphone zum Erzeugen des Entscheidungsbaums verwendet wurden, die auch in der Erkennung vorkommen. D.h. die beim Test "eine Beobachtung" noch hinzukommenden Triphone werden bei der Erkennung gar nicht gesehen, sondern tragen lediglich zum Trainingsmaterial des Enscheidungsbaums bei.

Tabelle 5.13: Wortfehlerraten [%] für State-Tying mit phonetischen Entscheidungsbäumen und geschlechtsabhängiger Koppelung auf WSJ-5k.

Kopplung	Zustände		mittlere LL		Dichten	DEL-INS	WER
	vor	nach	vor	nach	(k)	[%]	[%]
nein	23502	2001	219.73	211.85	192	1.2-0.7	7.6
ja	5562	2001	219.62	211.55	198	1.1-0.8	8.0
ja	17571	2001	219.62	211.73	210	1.3-0.7	8.1
ja (geglättet)	23502	2001	219.62	211.74	195	1.2-0.6	7.7

Tabelle 5.14: Wortfehlerraten [%] für State-Tying mit phonetischen Entscheidungsbäumen und voller Kovarianzmatrix auf WSJ-5k.

Kovarianz	Zustände		mittlere LL		Dichten	DEL-INS	WER
	vor	nach	vor	nach	(k)	[%]	[%]
diagonal	23502	219.73	211.85	2001	192	1.2-0.7	7.6
voll	9075	219.51	205.85	2001	196	1.2-0.9	8.0
voll, geglättet	9075	219.51	205.86	2001	212	1.2-0.7	7.9
voll, geglättet	23502	219.50	205.23	2001	199	1.2-0.6	7.7

Durch die Verwendung eines einzigen Baums und Fragen nach dem zentralen Phonem und Segment konnte die Fehlerrate nicht weiter gesenkt werden (siehe Tabelle 5.11). Auch die Verbesserung der Log-Likelihood ist bei dieser Methode schlechter als bei der Standardmethode. Offensichtlich sind die aufgrund des Aufteilungsverfahrens *lokal* optimalen Entscheidungen bei der Aufteilung der Knoten in den ersten Schichten des Baums *global* nicht optimal. Man erreicht mit dieser Methode 7.8%. Trotzdem wurde diese Struktur wegen der höheren Flexibilität für das derzeitige Basissystem des Lehrstuhls beibehalten, allerdings in der Form eines dreistufigen Aufbaus des Baums wie in Abschnitt 5.3.3 oben beschrieben.

Durch die Verschmelzung von Blättern nach dem Aufbau des Baums kann auf diesem Korpus ebenfalls keine Verbesserung erzielt werden (siehe Tabelle 5.12). Bei dem Test "ja (3000)" wurde zunächst ein Baum mit 3000 Blättern erzeugt, diese wurden dann durch das Verschmelzen auf 2000 Blätter reduziert. Dadurch wird eine im Vergleich zum Basisverfahren größere Verbesserung der Log-Likelihood erreicht, da beim Zusammenfassen der Blätter der Algorithmus praktisch frei entscheiden kann, welche Modelle zusammengefaßt werden. Die durch diese Methode erreichte Fehlerrate beträgt allerdings nur 7.8%, d.h. durch das unbeschränkte Zusammenfassen scheint die Verallgemeinerungsfähigkeit des Baums zu leiden.

Bei den Tests "ja (2000)" wurden Bäume mit jeweils 2000 Blättern erzeugt, diese wurden dann auf eine geringe Zahl von Blättern reduziert. Hier verschlechtert sich die Log-Likelihood auf den Trainingsdaten, da nun weniger Parameter für die Modellierung zur Verfügung stehen. Je nach Umfang der anschließenden Reduktion der Blätter werden Fehlerraten zwischen 8.0% und 8.4% erzielt. Bei einer Reduktion der Parameter des Systems um ca. 50% steigt durch das Verschmelzen der Blätter die Fehlerrate um ca. 10%. Dies kann u.U. bei Echtzeitanwendungen eine Rolle spielen, da hier eine Verringerung der Parameterzahl meist auch eine Beschleunigung der Suche zu Folge hat.

Durch die Verwendung von geschlechtsabhängigen Modellen in den Knoten kann die Log-Likelihood auf den Trainingsdaten deutlich verbessert werden. Dies ist nicht verwunderlich, da bei gleicher Modellzahl doppelt soviele Parameter zur Modellierung der Trainingsdaten verwendet werden. Allerdings verschlechtert sich die Fehlerrate für diese Methode auf 8.0%. Dies kann allerdings durch Glätten der Varianz der Gaußverteilungen, die zur Modellierung der Beobachtungen an den Baumknoten verwendet werden, verbessert werden (siehe Anhang B). Bei einer Glättung der Modelle mit den jeweiligen Modellen des Vaterknotens erzielt man 7.7%, was praktisch identisch mit der Fehlerrate des Basisverfahrens ist (siehe Tabelle 5.13).

Durch die Verwendung einer vollen Kovarianzmatrix ohne Glättung für die Gaußveteilungen an den Baumknoten erreicht man auf einer reduzierten Triphonliste eine Fehlerrate von 8.0%, die praktisch identisch mit der entsprechenden Fehlerrate des Basisverfahrens ist. Die Reduktion der Triphonliste war zunächst notwendig, da zu Beginn der Tests dieser Methode die eingesetzte Hardware nicht in der Lage war, die Parameter einer vollen Triphonliste zu verarbeiten. Durch anschließende Glättung der Kovarianzmatrizen kann diese auf 7.9% gesenkt werden, für die volle Triphonliste erzielt man schließlich 7.7%. Obwohl also durch die volle Kovarianzmatrix eine signifikante Verbesserung der Log-Likelihood erreicht werden kann, bleibt die Fehlerrate praktisch konstant (siehe Tabelle 5.14).

5.5 Zusammenfassung

Durch State-Tying konnten auf dem untersuchten Korpus eine Reduktion der Fehlerrate um ca. 8% erreicht werden. Es zeigte sich außerdem, daß aufgrund der erhöhten Zahl an nicht gesehenen Triphonen die Fehlerrate durch die Verwendung von phonetischen Entscheidungsbäumen im Vergleich zu einem rein datengetriebenen Verfahren noch einmal um 6% gesenkt werden kann, d.h. die gesamte Verbesserung der Fehlerrate durch State-Tying mit phonetischen Entscheidungsbäumen liegt bei ca. 13% verglichen mit einem System ohne State-Tying. Die in der Literatur beschriebenen und für diese Arbeit implementierten Erweiterungen des Basisverfahrens brachten nicht die erhofften Verbesserungen. Es muß in diesem Zusammenhang darauf hingewiesen werden, daß für keines der Verfahren, abgesehen von der Verwendung eines Baums, die möglichen Verbesserungen der Fehlerrate dokumentiert waren, so daß auch keine Aussage darüber getroffen werden kann, ob diese Beobachtung korpusspezifisch oder systemspezifisch ist. Andere Publikationen zeigen allerdings ein ähnliches Bild [Lazarides *et al.* 96] [Nock *et al.* 97], so daß abschließend folgende Aussagen zum State-Tying mit Entscheidungsbäumen getroffen werden können:

- Das implementierte Basisverfahren senkt die Fehlerrate auf dem getesteten Korpus um ca. 10-15% verglichen mit einem System ohne State-Tying,
- Das Basisverfahren erweist sich im Vergleich mit den getesteten Modifikationen als optimal.

Im nächsten Kapitel soll nun, aufbauend auf dem Verfahren des State-Tying mit phonetischen Entscheidungsbäumen, die Wortgrenzenmodellierung mit wortubergreifenden Triphonmodellen untersucht werden.

Kapitel 6

Wortgrenzenmodellierung

Die Verwendung von wortübergreifenden Triphonen stellt eine für die kontinuierliche Spracherkennung wichtige Verbesserung der akustischen Modellierung dar. Durch die genauere Modellierung der Wortübergänge wird vor allem die Modellierung kurzer Wörter verbessert, die oft für Fehler bei der Spracherkennung verantwortlich sind. Allerdings führt die Verwendung von wortübergreifenden Triphonen zu einer signifikanten Erhöhung der Komplexität des Suchverfahrens, so daß die konkrete Implementierung nicht unproblematisch ist. Im folgenden sollen nun die wesentlichen Schritte bei der Einführung der Modellierung mit wortübergreifenden Triphonen in ein Spracherkennungssystem dargestellt werden.

6.1 Einführung

Im Gegensatz zur Einzelworterkennung tritt bei der Erkennung von kontinuierlicher Sprache der Effekt der Koartikulation auch an Wortgrenzen auf. Je nachdem, wie "flüssig" ein Satz gesprochen wird, tauchen explizite Wortgrenzen in Form von Pausen zwischen den Wörtern gar nicht mehr auf. Dies bedeutet für ein Spracherkennungssystem, daß

- die Start- und Endzeitpunkte der Wörter nicht mehr durch einfache Pausedetektion bestimmt werden können und
- die akustische Realisierung der Wortgrenzenphoneme nicht nur vom Wort selbst, sondern auch vom vorhergehenden bzw. folgenden Wort abhängen.

Das erste Problem wird durch die heutigen Suchalgorithmen zur Erkennung von Wortfolgen gelöst, d.h. die Algorithmen sind in der Lage, zu einer gegebenen Folge von akustischen Vektoren alle plausiblen Wortfolgen zu hypothetisieren und daraus die gemäß dem akustischen Modell und dem Sprachmodell beste Wortfolge auszuwählen.
Das zweite Problem kann durch die explizite Modellierung der Wortgrenzen mit Triphonen gelöst werden. Man geht über von der *wortinternen* Kontextmodellierung zu der *wortübergreifenden* Kontextmodellierung.

- Bei der *wortinternen* Kontextmodellierung werden nur innerhalb der Wörter eines Satzes echte Triphone zur Modellierung des akustischen Kontextes verwendet. An

den Wortgrenzen werden Rechts- bzw. Links-Diphone verwendet, die als Wortgrenzenmodelle markiert sind, indem für den jeweils an der Wortgrenze befindlichen Kontext ein allgemeines Wortgrenzen-Kontextphonem "#" verwendet wird.

Beispiel: Phonetische Transkription des Satzes "Tom is cool" mit wortinternen Triphonen

$$\text{si} \quad \#\text{t}_{\text{aw}} \quad \text{t}\text{aw}_{\text{m}} \quad \text{aw}\text{m}_{\#} \quad \text{si} \quad \#\text{ih}_{\text{z}} \quad \text{ih}\text{z}_{\#} \quad \#\text{k}_{\text{ooh}} \quad \text{k}\text{ooh}_{\text{l}} \quad \text{ooh}\text{l}_{\#} \quad \text{si}$$

- Bei der zusätzlichen Verwendung von wortübergreifenden Triphonen ändert sich die Modellierung des Satzes an den Wortgrenzen. Bei Koartikulation werden die Wortgrenzen-Diphone der rein wortinternen Modellierung durch das jeweils passende Triphon ersetzt, der phonetische Kontext an der Wortgrenze wird durch das an der Wortgrenze liegende Phonem des Nachbarwortes bestimmt. Findet keine Koartikulation statt, d.h. liegt zwischen den beiden Wörtern eine (genügend lange) Pause, wird für das Kontexttriphon an der Wortgrenze "si" (Pause) verwendet.

  Beispiel: Phonetische Transkription des Satzes "Tom is cool" mit wortinternen und wortübergreifenden Triphonen

$$\text{si} \quad \text{si}\text{t}_{\text{aw}} \quad \text{t}\text{aw}_{\text{m}} \quad \text{aw}\text{m}_{\text{si}} \quad \text{si} \quad \text{si}\text{ih}_{\text{z}} \quad \text{ih}\text{z}_{\text{k}} \quad \text{z}\text{k}_{\text{ooh}} \quad \text{k}\text{ooh}_{\text{l}} \quad \text{ooh}\text{l}_{\text{si}} \quad \text{si}$$

  Beim ersten Wortübergang von "Tom" nach "is" wird angenommen, daß zwischen den Wörtern eine genügend lange Pause liegt, so daß keine Koartikulation stattfindet. Zwischen dem zweiten Wortpaar befindet sich keine Pause, so daß dort die allgemeinen Wortgrenzenphoneme "#" durch das passende Kontextphonem an der Wortgrenze ersetzt wird.

Die Entscheidung, ob an einer Wortgrenze Koartikulation stattfindet oder nicht, wird im einfachsten Fall durch die Länge der zwischen den Wörtern befindlichen Pause bestimmt. Liegt die Länge der Pause unter einem geeignet zu wählenden Schwellwert, wird Koartikulation angenommen, ansonsten wird der allgemeinere Pause-Kontext verwendet. Dieser Ansatz wird auch in dieser Arbeit betrachtet.
Unabhängig von der Komplexität der verwendeten Verfahren zur Wortgrenzenmodellierung stellt diese gegenüber der rein wortinternen Kontextmodellierung eine naheliegende Verbesserung dar. Allerdings wird diese Verbesserung durch einen nicht unerheblichen Komplexitätszuwachs erkauft. Für das Training muß abhängig von der Pauselänge zwischen den Worten des Trainingskorpus an den Wortgrenzen das jeweils passende Triphon eingesetzt werden. Wesentlich größer ist der Komplexitätszuwachs bei der Erkennung. Hier gibt es grundsätzlich zwei Ansätze:

- *n-best*-Suche

Die *n-best*-Suche läuft in zwei Phasen ab. In der ersten Suchphase wird mit wortinternen Triphonen eine sogenannte *n-best*-Liste erzeugt. Diese Liste enthält die n besten durch die erste Suchphase hypothetisierten Wortfolgen. Diese n Wortfolgen werden dann in einem zweiten Durchgang mit wortübergreifenden Triphonen durch eine nichtlineare Zeitanpassung neu bewertet. Der aufgrund dieser Bewertung beste Satz wird dann als erkannter Satz ausgegeben. Aufgrund der vor der Neubewertung eines Satzes bekannten Wortfolge muß an jeder Wortgrenze nur noch entschieden werden, ob Koartikulation auftritt oder nicht, die jeweiligen Kontextphoneme sind bekannt.

- einphasige Suche

  Bei der einphasigen Suche wird die Entscheidung, welcher Wortübergang für eine hypothetisierte Wortfolge verwendet werden muß, in die Baumsuche integriert. Dies führt zu folgendem Problem: Erreicht der Algorithmus eine Wortgrenze, ist zu diesem Zeitpunkt nicht bekannt, welches Wort darauf folgen wird. Daher kann auch nicht entschieden werden, welches wortübergreifende Triphon zur Modellierung des Wortendes verwendet werden muß. Daher muß die Entscheidung bzgl. des ersten Phonems des Nachfolgewortes in die letzte Astgeneration des letzten Phonems des Vorgängerwortes verlagert werden. Dies soll anhand des für diese Arbeit verwendeten zeitsynchronen Suchverfahrens mit Baumlexikon erläutert werden. Abbildung 6.1 a) zeigt zwei aufeinanderfolgende Worthypothesen für ein Baumlexikon ohne wortübergreifende Triphone. Die durchgezogenen Linien markieren den Übergang von Phonem zu Phonem innerhalb eines Wortes, die gestrichelten Linien den Übergang an Wortgrenzen. Die Schreibweise `a{b+c}` bedeutet ein kontextabhängiges Phonem `a` im linken Kontext `b` und rechten Kontext `c`. Erreicht die Suche eine solche Wortgrenze, wird ein kompletter neuer Baum gestartet.

  Im Gegensatz dazu wird bei der Suche mit wortübergreifenden Triphonen (Abbildung 6.1 b)) das jeweils letzte Modell jedes Wortes derart aufgefächert, daß für jeden möglichen rechten Kontext einschließlich der Pause ein Phonemast mit entsprechendem Modell vorhanden ist. Die nach dem Wortende zu startenden neuen Bäume enthalten nur noch den Teilbaum, der aufgrund des vorweg genommenen rechten Kontextes abzusuchen ist.

  Durch die Auffächerung des Baumlexikons an den Endknoten und die notwendigen Baumkopien vergrößert sich der Suchaufwand erheblich. Eine grobe Abschätzung ist die folgende: Die Zahl der Wortenden für ein 20 000-Wort-Lexikon ist ebenfalls 20 000. Diese müssen für die n möglichen rechten Kontexte ($n-1$ Phoneme plus Pause) aufgefächert werden. Die Größe des Baumlexikons ohne Wortgrenzenmodelle beträgt ungefähr 65 000 Phonemäste. Die Größe des Baumlexikons mit Wortgrenzenmodellen beträgt daher $20\,000 \cdot n + 65\,000$ Phonemäste. Bei 43 Phonemen ergäbe das eine Größe von annähernd 1 000 000 Ästen. Berücksichtigt man weiterhin, daß bei jedem der Wortübergänge zwei dieser Bäume gestartet werden müssen (ein Baum für den linken Pause-Kontext und 43 Teilbäume für die linken Phonemkontexte), kämen bei jedem Wortübergang ca. 2 000 000 potentiell abzusuchende Äste hinzu. Dies macht es zwingend erforderlich, für die einphasige Wortgrenzensuche effiziente Verfahren zu verwenden, um diesen Suchaufwand beherrschen zu können.

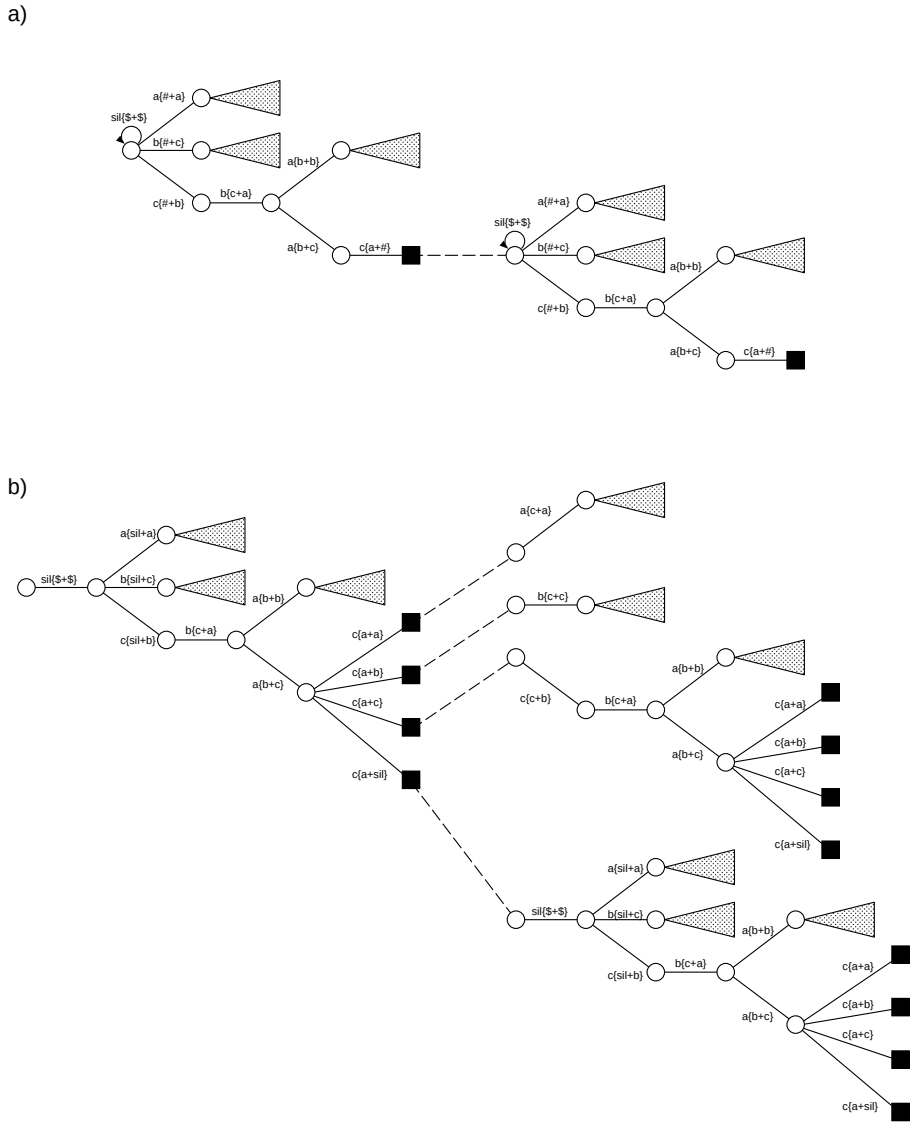


Abbildung 6.1: Baumorganisation beim Übergang von rein wortinterner Triphonmodellierung zur Modellierung mit wortinternen und wortübergreifenden Triphonen

In den weiteren Abschnitten sollen nun einige wesentliche Aspekte bei der akustischen Modellierung und der Suche für die Verwendung von wortübergreifenden Triphonen untersucht werden. Insbesondere die für die Suchverfahren zu optimierenden Parameter sollen dabei im Blickpunkt stehen.

6.2 Training

Das Problem von im Training nicht gesehenen Triphonen tritt bei der Verwendung von wortübergreifenden Triphonen in weit größerem Maße auf als bei der Verwendung von rein wortinternen Triphonen, da durch die Wortübergänge zusätzliche Triphone auftreten können, die im Wortinnern aus phonotaktischen Gründen nicht vorkommen. Tabelle 6.1 zeigt die Zahl der im Training auftretenden bzw. in der Erkennung möglichen Tri-

phone für rein wortinterne Triphone und für zusätzliche wortübergreifende Triphone für den WSJ-5k-Korpus, Tabelle 6.2 zeigt dies für den Verbmobil-Korpus. Für wortinterne Triphone ist die Zahl der im Training gesehenen Triphone höher als die Zahl der in der Erkennung möglichen Triphone, obwohl auch dort keine vollständige Überdeckung der Triphone des Erkennungslexikons durch die im Training gesehenen Triphone vorhanden ist. Bei zusätzlicher Verwendung von wortübergreifenden Triphonen beträgt die Zahl der im Trainingskorpus gesehenen Triphone nur ca. 60% der nach dem Erkennungslexikon, das für den WSJ-5k-Korpus verwendet wurde, möglichen Triphone, so daß die Zahl der im Training nicht gesehenen Triphone für die Erkennung mit 40% nach unten abgeschätzt werden kann.

Tabelle 6.1: Anzahl der im Training gesehenen bzw. im Erkennungslexikon vorkommenden Triphone für WSJ0 (Training) und WSJ-5k (Erkennung).

Korpus	wortintern	wortübergreifend
Training (tatsächlich aufgetreten)	7881	17040
Erkennung (maximal möglich)	5654	28716

Tabelle 6.2: Anzahl der im Training gesehenen bzw. im Erkennungslexikon vorkommenden Triphone für VM-5k.

Korpus	wortintern	wortübergreifend
Training (tatsächlich aufgetreten)	6127	13012
Erkennung (maximal möglich)	6131	28580

D.h. ohne die Verwendung von phonetischen Entscheidungsbäumen für das State-Tying würde eine erhebliche Menge der Triphone des Erkennungslexikons durch Monophone modelliert werden. Wie in [Hwang 93] demonstriert wird, ist aber die Verwendung von phonetischen Entscheidungsbäumen für das State-Tying besonders für Sätze mit einer hoher Zahl von nicht gesehenen Triphonen für die Erkennungsgenauigkeit entscheidend. Daher setzt die Verwendung von Wortgrenzenmodellen praktisch die Verwendung von phonetischen Entscheidungsbäumen beim State-Tying voraus.

6.2.1 Schätzung der Pauselänge zwischen den Wortgrenzen

Das akustische Training wird zur Verwendung von wortübergreifenden Triphonmodellen dahingehend modifiziert, daß an jeder Wortgrenze die Länge der zwischen den Worten liegenden Pause bestimmt wird. Abhängig davon, ob diese Länge oberhalb oder unterhalb eines Schwellwertes N_{sil} liegt, wird für die Wortgrenzentriphone entweder der phonetische Kontext des benachbarten Wortes oder der allgemeinere Pause-Kontext verwendet. Die Verteilung der Länge der Pause an Wortgrenzen für die verwendeten Trainingskorpora WSJ und Verbmobil ist in Abbildung 6.2 zu sehen. Die Histogramme für die beiden Korpora lassen sich grob in zwei Abschnitte mit einem Übergangsbereich einteilen, wobei der Verlauf der Kurve in den jeweiligen Abschnitten linear ist. Die nicht in diesem Diagramm eingetragenen Pausen der Länge Null, bei denen auf jeden Fall Koartikulation am

Wortübergang angenommen wird, machen ca. 80% der im Trainingsmaterial vorkommenden Wortübergänge aus.

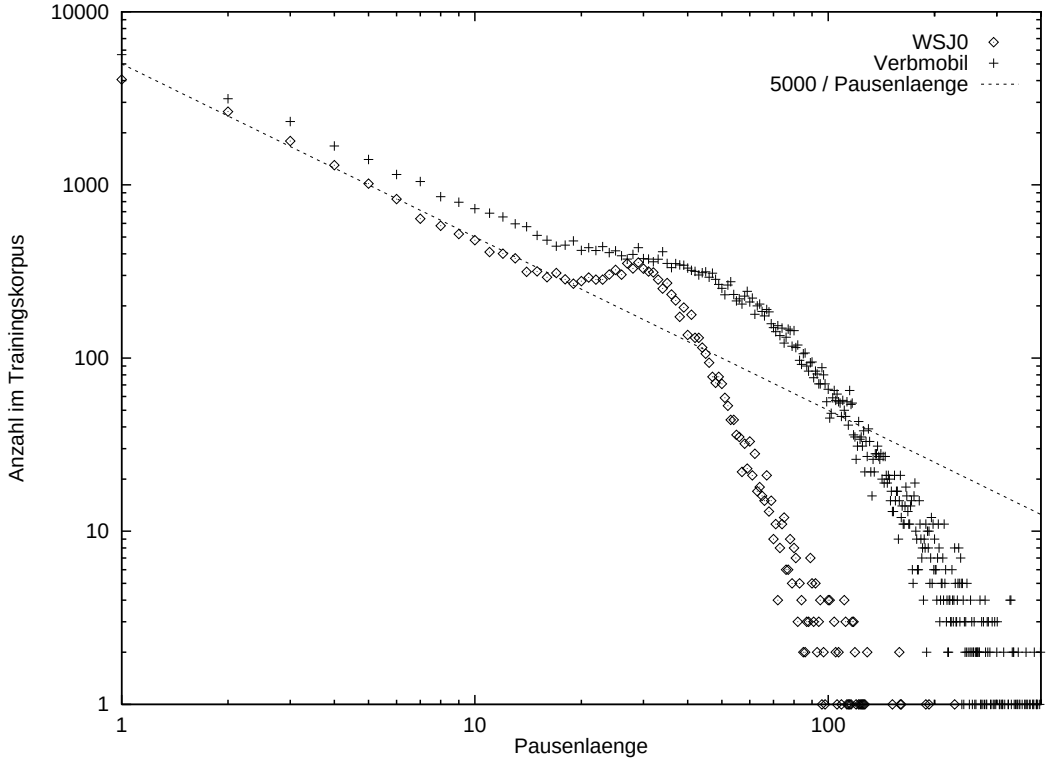


Abbildung 6.2: Verteilung der Pauselängen auf WSJ0 und VM-5k

Diese Pauselängen werden nun in jeder Iteration des Trainings neu geschätzt. Anhand dieser Pauselängen wird dann entschieden, ob an einer Wortgrenze Koartikulation angenommen wird oder nicht. Das am Institut verwendete System erzeugt dazu eine Datei, welche die geschätzten Pauselängen enthält. Diese Information wird dann in der jeweils nächsten Trainingsiteration verwendet. Da bei diesem Ansatz in der ersten Iteration noch keine geschätzten Pauselängen zu Verfügung stehen, werden zur Initialisierung die Pauselängen in der ersten Iteration auf die wahrscheinlichste gesetzt, d.h. zu Null angenommen.
Allerdings sind durch diese Näherung die Modelle nach der ersten Trainingsiteration, d.h. nach einer Erzeugung eines Entscheidungsbaums und nachfolgendem Training, noch nicht optimal auf die Trainingsdaten adaptiert. Daher werden diese beiden Schritte zweimal durchgeführt, wobei die in der ersten Iteration gewonnene Segmentierung der Daten und die Information über die Zwischenwortpause zur Generierung des zweiten Entscheidungsbaums und zur Initialisierung des zweiten Trainings verwendet werden (siehe nächster Abschnitt).

6.2.2 Iterative Bestimmung der phonetischen Entscheidungsbäume

Ein weiteres Problem bei einem initialen Training von wortübergreifenden Triphonmodellen besteht darin, daß noch kein mit wortübergreifenden Triphonmodellen generierter Entscheidungsbaum für das State-Tying zur Verfügung steht. Wie im Kapitel 5 gezeigt,

ist es für eine optimale Wortfehlerrate wichtig, möglichst alle im Training vorkommenden Triphone zur Generierung des Baums zu verwenden. Daher kann davon ausgegangen werden, daß der nur mit wortinternen Triphonen erzeugte Baum für ein Spracherkennungssystem mit wortübergreifender Triphonmodellierung eine suboptimale Erkennungsleistung zur Folge haben wird. Ein anderes Problem wiegt noch schwerer. Da der mit wortinternen Triphonen erzeugte Baum keinen Pausen-Phonemkontext kennt, müßte für diesen Kontext der allgemeine Wortgrenzenkontext "#" verwendet werden, der bei der rein wortinternen Kontextmodellierung auch dann verwendet wird, wenn Koartikulation an einer Wortgrenze vorlag. Aus diesen Gründen ist es für ein optimales System mit wortübergreifender Triphonmodellierung erforderlich, einen phonetischen Entscheidungsbaum zu generieren, der die veränderte Modellierung an den Wortgrenzen berücksichtigt. Dazu sind folgende Schritte notwendig:

- Erzeugen eines phonetischen Entscheidungsbaums (erste Generation)

  Die hierfür benötigten Statistiken über die Trainingsdaten (Beobachtungen für jedes Segment) werden aus einem Pfad (d.h. einer Segmentierung der Daten) von einem vorangegangenen Training mit Wortgrenzenmodellierung mit wortübergreifenden Triphonen berechnet. Steht kein solcher Pfad zur Verfügung, wird mit Modellen für wortinterne Kontextmodellierung eine Zeitanpassung auf einer entsprechend der Wortgrenzenmodellierung transkribierten Zustandsfolge durchgeführt (siehe unten). Dabei wird an jeder Wortgrenze der wortübergreifende Kontext eingesetzt, da noch keine bessere Information vorhanden ist und dies für ca. 80% der Fälle der richtige Kontext ist.

- Trainieren der Parameter des Baums (erste Generation)

  Als Ausgangspunkt hierfür wird ebenfalls die obige Zustandszuordnung benutzt. Man erhält einen ersten Parametersatz für die akustischen Modelle.

- Erzeugen eines phonetischen Entscheidungsbaums (zweite Generation)

  Die hiefür benötigten Statistiken über die Trainingsdaten werden aus dem letzten Zeitanpassungspfad des Trainings der ersten Generation ermittelt.

- Trainieren der Parameter des Baums (zweite Generation)

  Man erhält die endgültigen Wortgrenzenmodelle

Mit diesen akustischen Modellen kann dann eine Erkennung auf den jeweiligen Korpora durchgeführt werden.
Im Gegensatz zum oben beschriebenen akustischen Training, bei dem relativ wenige Modifikationen zur Verwendung von wortübergreifenden Triphonmodellen nötig, muß für die Erkennung ein erheblicher Aufwand getrieben werden. Wie weiter oben schon diskutiert, gibt es hier zwei grundsätzliche Möglichkeiten,

- die *n-best*-Suche und
- die einphasige Suche

Beide sollen in dieser Arbeit untersucht und verglichen werden.

6.3 Erkennung: *n-best*-Suche

Die *n-best*-Suche [Schwartz *et al.* 91] [Schwartz *et al.* 92] gehört zur Gruppe der mehrphasigen Suchverfahren. Bei diesen Verfahren wird der Suchprozeß in mehrere Phasen unterteilt, wobei die Suche der Phase i die Ergebnisse der vorausgegangenen Suchdurchgänge $1, \ldots, i-1$ verwendet. Im allgemeinen besteht das Ergebnis eines Suchdurchgangs aus einer Beschreibung des Suchraums, mit der sich die Suche in der nächsten Phase einschränken läßt. In der letzten Stufe wird dann die Entscheidung über den besten Satz bzgl. der verwendeten Modelle getroffen, der dann als erkannter Satz ausgegeben wird. Vorteil bei einer solchen Strategie ist, daß insbesondere bei der Verwendung von komplexen akustischen und/oder Sprachmodellen in den ersten Phasen eine Suche verwendet werden kann, die eine große Anzahl von Satzhypothesen mit einfachen Modellen einschränkt, wonach in den nachfolgenden Suchphasen diese eingeschränkte Menge von Hypothesen mit komplexeren Modellen, aber einfacheren Suchverfahren neu bewertet werden kann. D.h. die Modellkomplexität streigt mit jeder Phase an, während die Suchkomplexität entsprechend sinkt (siehe Abbildung 6.3)

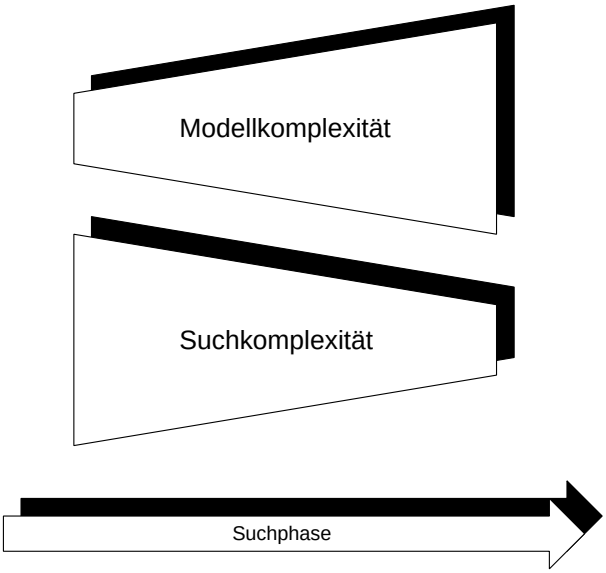


Abbildung 6.3: Modell- versus Suchkomplexität

Die *n-best*-Suche ist ein zweistufiges Verfahren, bei der in der ersten Phase die bezüglich dieser Suchphase n besten Sätze generiert werden. Diese n Sätze werden dann in der zweiten Phase mit komplexeren Modellen neu bewertet. Diese Neubewertung kann mit einer einfachen nichtlinearen Zeitanpassung geschehen, da zu jedem Satz die Wortfolge vor der Neubewertung vollständig bekannt ist. Daher ist die Integration von komplexen Wissensquellen in die zweite Suchphase der *n-best*-Suche meist auch mit relativ wenig Aufwand verbunden.

Im folgenden soll nun die Anwendung der *n-best*-Suche auf die Wortgrenzenmodellierung mit wortübergreifenden Triphonen dargestellt werden. Insbesondere soll auf die dafür notwendigen Optimierungsschritte eingegangen werden.

6.3.1 Basisverfahren

Die für diese Arbeit implementierte *n-best*-Suche basiert auf der in der Standardsuche des verwendeten Systems integrierte Wortgrapherzeugung. Wie in [Schwartz *et al.* 91] dargestellt, lassen sich auf einem Wortgraphen relativ leicht *n-best*-Satzlisten erzeugen. Die Vorgehensweise bei der *n-best*-Suche ist daher grundsätzlich die folgende:

- Erzeugung eines Wortgraphen mit der wortabhängigen Strahlsuche mit baumorganisiertem Lexikon und wortinternen Triphonen,
- Generierung der n besten Sätze auf diesem Wortgraphen,
- Neubewertung der n besten Sätze mit wortübergreifenden Triphonen.

Diese Vorgehensweise stellt, insbesondere wenn die Algorithmen zur Erzeugung eines Wortgraphen in der einphasigen Suche schon vorhanden sind, einen guten Kompromiß zwischen Implementierungsaufwand und Komplexität des Suchalgorithmus dar, zumindest bei Aufgabenstellungen, bei denen relativ kurze Sätze zu verarbeiten sind (unter zehn Sekunden). Bei sehr langen Sätzen müssen relativ viele Sätze neu bewertet werden, um eine ausreichende Abdeckung des Suchraums zu erreichen. Daher ist bei Korpora mit langen Sätzen die Alternative "einphasige Suche" zu bevorzugen.

6.3.2 Erzeugung eines Wortgraphen

Die Idee, die hinter der Verwendung eines Wortgraphen steckt, besteht darin, die in der integrierten Suche bewerteten Satzalternativen, die relativ nahe bei der Bewertung der besten Satzalternative liegen, aber aufgrund der Rekombination an Wortgrenzen nicht bis zum Satzende berücksichtigt werden, in kompakter Form darzustellen [Ney *et. al.* 94]. Ein Wortgraph ist im Sinne der Graphentheorie ein gerichteter, azyklischer, bewerteter Graph. Die Knoten des Graphen stellen Zeitpunkte dar, an denen Worthypothesen geendet sind, die Kanten repräsentieren die Worthypothesen selbst (siehe Bild 6.4). Jede Kante ist zusätzlich zu ihrem Wort mit der am Ende des Wortes aufgetretenen akustischen Bewertung (ohne Sprachmodell) versehen.

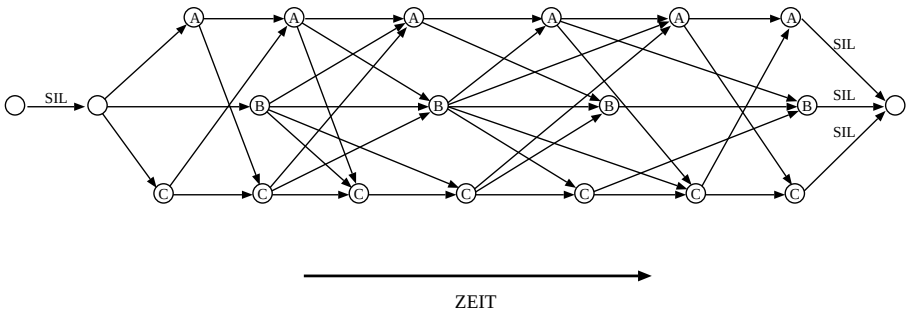


Abbildung 6.4: Wortgraph

Mit einem Wortgraphen lassen sich nun die Satzhypothesen, die während der akustischen Suche betrachtet wurden, mit ihren Bewertungen rekonstruieren. Man startet dazu bei dem (eindeutigen) Knoten zum Startzeitpunkt des Satzes und verfolgt einen Pfad zum

(ebenfalls eindeutigen) Endknoten zum Endzeitpunkt des Satzes. Die dabei akkumulierten akustischen Bewertungen zusammen mit der Sprachmodellbewertung (aufgrund der durchlaufenen Wortfolge) ergeben die letztendliche Bewertung des Satzes im Sinne von Formel 1.1.

Eine für diese Arbeit verwendete Näherung bei der Erzeugung des Wortgraphen stellt die sogenannte Wortpaarapproximation dar. Dabei wird angenommen, daß der Startzeitpunkt eines Wortes w ausschließlich von seinem direkten Vorgängerwort v abhängt (siehe Abbildung 6.5). Dies ist für ausreichend lange Wörter v in guter Näherung erfüllt, da Zeitanpassungspfade für das Wort v mit unterschiedlicher Startzeit vor dem Start des Wortes w rekombiniert werden können (Abbildung 6.5 oben). Für kurze Wörter v ist diese Approximation zunehmend weniger zutreffend (Abbildung 6.5 unten), wie in [Ortmanns 98] nachgewiesen wurde. Für die Erzeugung einer *n-best*-Liste ist diese Ungenauigkeit aber nicht relevant, da sie nur zu unerheblichen Veränderungen der Satzbewertungen führt, d.h. nur für sehr kleine n ($n < 10$) kann eine wesentliche Änderung der Liste bzgl. der enthaltenen Sätze erwartet werden.

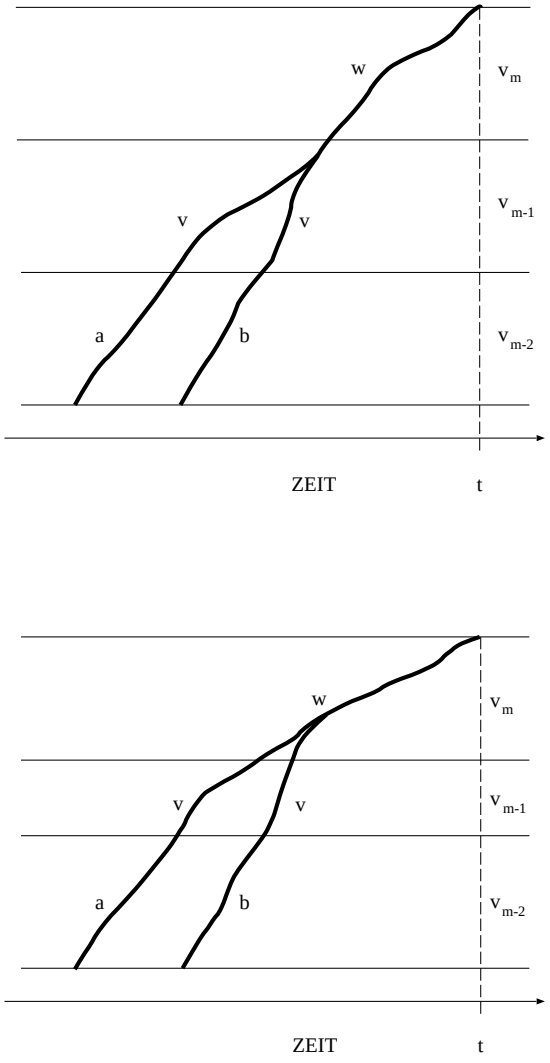


Abbildung 6.5: Wortpaarapproximation

Die Generierung eines Wortgraphen unter Berücksichtigung der Wortpaarapproximation kann dann durch folgenden Algorithmus erfolgen [Ortmanns 98]:

- Betrachte zu jedem Zeitpunkt die wahrscheinlichsten Wortpaare $v_{m-1}^{m} = (v, w)$
- Speichere für jedes Tripel (v, w, t)
  - die Wortgrenze $\tau(v, w, t)$
  - die akustische Wortbewertung $h(w, \tau(v, w, t), t)$
- Konstruiere den Wortgraphen durch Zurückverfolgen der in der Traceback-Liste enthaltenen Entscheidungen

Eine genaue Beschreibung dieses Verfahrens findet sich in [Ortmanns 98]. Der so erzeugte Wortgraph wird nun zur Generierung der n besten Sätze verwendet. Dazu soll zunächst kurz geschildert werden, wie der beste Satz aus dem Wortgraphen extrahiert werden kann. Dieses Verfahren kann leicht auf n Sätze verallgemeinert werden.
Dazu wird zeitsynchron an jedem Knoten die lokal optimale Bewertung mit Hilfe der dynamischen Programmierung berechnet. Man betrachtet zu einem Knoten k alle möglichen Vorgängerknoten und optimiert über deren Bewertungen plus der Bewertung durch das akustische Modell und das Sprachmodell für die Expansion der Hypothesen in diesen Knoten k. Dies wird Zeitschritt für Zeitschritt durchgeführt, so daß zum Endzeitpunkt T aufgrund der links-rechts-Struktur des Wortgraphen die beste Satzhypothese am Endknoten steht (siehe auch [Ortmanns 98]).
Dieses Verfahren kann auf die Generierung der n besten Sätze erweitert werden, indem in jedem Knoten nicht nur die beste, sondern die n besten Hypothesen gespeichert werden. Die Optimierung geschieht dann nicht mehr über die m Hypothesen der m Vorgängerknoten, sondern über deren mn Hypothesen. Durch diese Vorgehensweise kann sichergestellt werden, daß am Endknoten auf jeden Fall die n besten Satzhypothesen zu finden sind (siehe Abbildung 6.6).

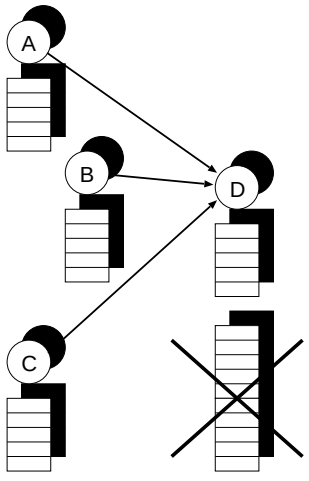


Abbildung 6.6: Propagierung der n-best-Listen durch den Wortgraphen

In diesem Beispiel werden die n-best-Listen der Knoten A, B und C der Länge 5 zum Knoten D propagiert. Hier werden zu den Bewertungen der Listen die Sprachmodellbewertung und die lokale akustische Bewertung für Knoten D addiert, danach sortiert und alle Hypothesen bis auf die 5 besten verworfen.
Im Prinzip kann durch diesen Algorithmus allerdings nicht garantiert werden, daß die Wortfolgen der am Endknoten berechneten n Satzhypothesen paarweise verschieden sind, da allein die zeitliche Verschiebung eines der Knoten des Wortgraphen u.U. dieselbe Wortfolge mit einer leicht anderen Bewertung entstehen läßt. Durch die sogenannte *Wortgrenzenoptimierung*, bei der äquivalente Knoten des Wortgraphen in einer bestimmten zeitlichen Umgebung zusammengefaßt werden, kann dieser Effekt praktisch ausgeschlossen werden. Untersuchungen an den in dieser Arbeit verwendeten Korpora haben gezeigt, daß nach der Wortgrenzenoptimierung keine doppelten Wortfolgen in den *n-best*-Listen vorkamen.
Die so erzeugte *n-best*-Liste wird dann mit den im Training geschätzten wortübergreifenden Triphonmodellen neu bewertet. Dazu wird die in der ersten Suchphase mit rein wortinternen Modellen ermittelte Information, ob zwischen zwei Wörtern eine Pause liegt oder nicht, verwendet, um das für die Zeitanpassung benötigte Satzmodell mit wortübergreifenden Triphonen zu erzeugen. Als Kriterium, ob an Wortgrenzen Koartikulation angenommen wird oder nicht, wird der im Training verwendete Schwellwert für die Pauselänge benutzt. Diese zunächst rein akustische Bewertung wird dann um die Bewertung des Sprachmodells ergänzt. Diese Gesamtbewertung ist dann das Kriterium für die Auswahl des besten Satzes.
Eine wesentliche Beschleunigung der akustischen Neubewertung kann erreicht werden, indem die lokalen akustischen Bewertungen zu einem Zeitpunkt t und einer Mischverteilung s in einem Cache zwischengespeichert werden. Da sich die Wortfolgen von Satz zu Satz in der n-best-Liste nur wenig ändern, muß nur für den ersten Satz der n-best-Liste eine komplette akustische Neubewertung vorgenommen werden. Für alle folgenden Sätze finden sich die meisten lokalen akustischen Bewertungen im Cache, so daß nur ein geringer Teil neu berechnet werden muß. Die Beschleunigung für eine Neubewertung eines Satzes beträgt dadurch ca. Faktor 5-10.

6.3.3 Ergebnisse

Die Ergebnisse für Wortgrenzenmodelle mit wortübergreifenden Triphonmodellen und die *n-best*-Suche wurden mit den in Kapitel 1 angesprochenen cepstralen Merkmalen (33 Komponenten) erzeugt. Weitere Randbedingungen bei den folgenden Tests:

- Laplace-Verteilungen mit gepooltem Varianzvektor,
- State-Tying mit phonetischem Entscheidungsbaum (2000 Blätter),
- Bigramm-Sprachmodell.

Diese Randbedingungen gelten für alle in diesem Abschnitt erwähnte Tests, falls keine anderen Angaben gemacht werden.
Zur Evaluierung der Modellierung mit wortübergreifenden Triphonen wurden Tests auf verschiedenen Spracherkennungskorpora durchgeführt Tabelle 6.3 zeigt die Ergebnisse auf

dem WSJ-5k und dem VM-5k-Korpus für ein Bigramm-Sprachmodell ($PP = 107$ für WSJ-5k, $PP = 66$ für VM-5k), dynamische Pauseschätzung und keine Interpolation (siehe Abschnitte "Schätzung der Länge der Zwischenwortpause" und "Interpolation mit wortinternen Triphonmodellen"). Damit wird auf dem WSJ-5k-Korpus mit wortübergreifenden Triphonmodellen eine Verbesserung von 7.1% auf 6.3% erreicht, auf dem VM-5k-Korpus von 21.9% auf 20.5%. Dies entspricht einer relativen Verbesserung von 8-10% (die bessere Fehlerrate des Ergebnisses mit rein wortinternen Triphonmodellen gegenüber den Tests aus dem letzten Kapitel ist auf die Verwendung der cepstralen Merkmale zurückzuführen, die zum Zeitpunkt der Tests für State-Tying noch nicht zur Verfügung standen).
Für ein Trigramm-Sprachmodell ($PP = 56$ für WSJ-5k, $PP = 51$ für VM-5k) (Tabelle 6.4) sind die erreichten Verbesserungen geringer. Auf dem WSJ-5k-Korpus reduziert sich die Fehlerrate von 4.9% auf 4.6%, auf dem VM-5k-Korpus von 20.0% auf 19.1%, was einer relativen Verbesserung von 5-7% entspricht. Für die getesteten Korpora sind die durch den Übergang von Bigramm- zum Trigramm-Sprachmodell bzw. von den rein wortinternen Triphonen zu den wortübergreifenden Triphonen erzielten Verbesserungen der Fehlerrate also nicht additiv.

Tabelle 6.3: Wortfehlerraten [%] für wortinterne und wortübergreifende Triphone und Bigramm-Sprachmodell auf WSJ-5k ($n = 20$) und VM-5k ($n = 100$).

Korpus	Wortgrenzenmodellierung	DEL-INS[%]	WER[%]
WSJ-5k	wortintern	1.3 - 0.7	7.1
	wortübergreifend	1.0 - 0.8	6.4
VM-5k	wortintern	4.4 - 3.7	21.9
	wortübergreifend	3.9 - 3.9	20.5

Tabelle 6.4: Wortfehlerraten [%] für wortinterne und wortübergreifende Triphone und Trigramm-Sprachmodell auf WSJ-5k ($n = 20$) und VM-5k ($n = 100$).

Korpus	Wortgrenzenmodellierung	DEL-INS[%]	WER[%]
WSJ-5k	wortintern	0.8 - 0.5	4.9
	wortübergreifend	0.8 - 0.5	4.6
VM-5k	wortintern	3.8 - 3.3	20.0
	wortübergreifend	3.7 - 3.7	19.1

Auffällig ist bei diesen Ergebnissen vor allem, daß sich die erzielten Verbesserungen nur in einer geringeren Zahl der Auslassungen und Vertauschungen, nicht aber in einer Verringerung der Einfügungen niederschlägt. Dies kann damit erklärt werden, daß bei der Hinzunahme von wortübergreifenden Triphonen die Genauigkeit der akustischen Modellierung an den Wortgrenzen verbessert wird. Dadurch wird auch die Wahrscheinlichkeit größer, daß längere Worte durch mehrere kürzere ersetzt werden, was vor allem die Anzahl der Einfügungen beeinflußt.
Tabelle 6.5 enthält Vergleichsergebnisse anderer Gruppen für Systeme ohne und mit Wortgrenzenmodellierung (siehe [Odell 95] [Beyerlein *et al.* 97] [Hon 92]). Die erzielten Ergebnisse sind hier für die Systeme von *Cambridge University* und *CMU* teilweise deutlich

Tabelle 6.5: Vergleichsergebnisse anderer Gruppen für Wortgrenzenmodellierung (Bigramm).

Gruppe	Korpus	WER [%] wortintern	WER [%] wortübergr.
Cambridge University	WSJ (eval 92 + eval 93)	10.0	7.6
Philips Forschungslab.	WSJ (92 + 93)	9.0	8.2
CMU	TIRM	7.5	6.0
	VI-2000	9.1	6.5
diese Arbeit	WSJ-5k	7.1	6.4
	VM-5k	21.9	20.5

besser, liegen aber für das System von *Philips Forschungslab.*, das dem System der RWTH am ähnlichsten ist, auf gleichem Niveau. Grund hierfür könnte der jeweilige technische Stand der verschiedenen Systeme sein, die Ergebnisse von *Cambridge University* stammen aus dem Jahre 1993, die Ergebnisse der *CMU* wurden vor 1992 erzielt, während die Resultate von *Philips Forschungslab.* und unsere Resultate deutlich jüngeren Datums sind. Daher könnte ein Teil der bei den älteren Systemen erzielten Verbesserungen durch andere Verfahren der neueren Systeme schon abgedeckt werden, die in den älteren Systemen nicht enthalten sind.
Ein wichtiger Unterschied ist, daß die Ergebnisse von *Cambridge University* und *Philips Forschungslab.* mit einphasigen Suchmethoden erzielt worden, eine solche Methode wird in Abschnitt 6.4 für das RWTH-System beschrieben und evaluiert.
Im folgenden sollen nun verschiedene Optimierungen der *n-best*-Suche betrachtet werden. Diese Optimierungen beziehen sich auf

- den Schwellwert für die maximale Pauselänge, bis zu dem noch eine Koartikulation modelliert wird,
- den Sprachmodellfaktor,
- die Länge der *n-best*-Liste,
- die Anzahl der Mischverteilungskomponenten des Systems,
- den Aufwand bei der ersten Phase der Suche,
- die Erkennung des "korrekten" Wortgrenzenkontextes (Koartikulation oder keine Koartikulation) während der akustischen Neubewertung und
- der Interpolation der Satzbewertungen von beiden Suchphasen.

6.3.3.1 Pauseschwellwert N_{sil}

Dieser Schwellwert legt fest, ab welcher Länge der Zwischenwortpause keine Koartikulation mehr modelliert wird. Die Länge dieser Pause wird dabei für das Basisverfahren in der ersten Suchphase bestimmt, indem die im Wortgraphen enthaltene Information

über Zwischenwortpausen und deren Längen für die zweite Suchphase verwendet werden (bei den Ergebnissen auf VM-5k wurde nur das in Abschnitt "Schätzung der Länge der Zwischenwortpause" beschriebene Verfahren verwendet). Diese Pauseschätzung wird im folgenden mit *statischer* Schätzung bezeichnet. Durch die Variation des Schwellwerts N_{sil} kann zwischen einer rein wortinternen Kontextmodellierung ($N_{sil} = 0$) und der Annahme, daß unabhängig von der Pauselänge zwischen den Wörtern immer Koartikulation vorliegt ($N_{sil} = \infty$) variiert werden. Tabelle 6.6 zeigt die Ergebnisse für verschiedene Pauselängen, die für Training und Erkennung konsistent verwendet wurden.

Tabelle 6.6: Wortfehlerrate [%] für verschiedene Koartikulationsschwellwerte für $n = 20$ auf WSJ-5k.

N_{sil}	# Dichten (m/w)	DEL-INS[%]	WER[%]
1	66k/73k	1.0 - 0.8	6.4
2	72k/75k	1.0 - 0.8	6.6
5	74k/70k	1.1 - 0.8	6.7
∞	68k/81k	1.1 - 0.9	7.0

Offensichtlich ist schon bei einem Schwellwert von 1 das Optimum erreicht, für größere Werte verschlechtert sich die Fehlerrate wieder. D.h. schon ab einer Pauselänge von 1 zwischen zwei Wörtern scheinen die Koartikulationseffekte deutlich abzunehmen.

6.3.3.2 Sprachmodellfaktor

Bei den ersten mit wortübergreifenden Triphonen durchgeführten Tests wurde festgestellt, daß sich das Verhältnis von Auslassungen zu Einfügungen stark geändert hatte. Dies deutet meist darauf hin, daß die Gewichtung des Sprachmodells u.U. neu eingestellt werden muß. Tabelle 6.7 zeigt die Auswirkungen einer Änderung der Sprachmodellgewichtung für die akustische Neubewertung.
Der optimale Wert für diesen Parameter liegt für den WSJ-5k-Korpus bei 20, verglichen mit einem Wert von 15 für die rein wortinternen Modelle. Für die durchgeführten Tests wurde daraufhin dieser Wert von 20 verwendet. Für den VM-5k-Korpus sind beide Werte im Optimum identisch, so daß keine allgemeinen Richtlinien für die Einstellung dieses Parameters für Wortgrenzenmodelle abzuleiten sind. Man kann lediglich feststellen, daß u.U. eine Veränderung der Sprachmodellgewichtung für Wortgrenzenmodelle sinnvoll sein kann.

6.3.3.3 Länge der n-best-Liste

Die Länge der *n-best*-Liste bestimmt den Suchraum, der in der zweiten Suchphase ausgewertet wird. Man sollte daher vermuten, daß sich die Verbesserungen der Modellierung durch Wortgrenzenmodelle mit steigender Länge der Liste immer deutlicher zeigen, für sehr große n sollte schließlich eine Sättigung eintreten. Um so überraschender sind die in Tabelle 6.8 gezeigten Ergebnisse.
Für den WSJ-5k-Korpus ist schon mit 2 Sätzen die Hälfte der möglichen Verbesserungen erreicht, bei 10-20 ausgewerteten Sätzen befindet sich ein Minimum, für größere n steigt

Tabelle 6.7: Wortfehlerrate [%] für verschiedene Sprachmodellgewichtungen für $n = 20$ auf WSJ-5k.

Gewichtung	DEL-INS[%]	WER[%]
13	1.0 - 0.8	7.0
14	1.0 - 0.8	6.8
15	1.1 - 0.8	6.7
16	1.1 - 0.9	6.6
17	1.1 - 0.9	6.6
18	1.1 - 0.9	6.5
19	1.1 - 0.9	6.4
20	1.0 - 0.8	6.4
21	1.1 - 0.9	6.5
22	1.1 - 0.9	6.5
23	1.1 - 0.9	6.5
24	1.1 - 0.9	6.5
25	1.1 - 0.9	6.6

Tabelle 6.8: Wortfehlerrate [%] für verschiedene Längen der n-best-Liste auf WSJ-5k.

n	DEL-INS[%]	WER[%]
2	1.1 - 0.8	6.7
5	1.0 - 0.7	6.4
10	1.0 - 0.8	6.4
20	1.0 - 0.8	6.4
50	1.0 - 0.8	6.5
100	1.0 - 0.8	6.5
200	1.0 - 0.8	6.5

die Fehlerrate wieder an. Für den VM-5k-Korpus kann dieses Verhalten in Tabelle 6.14 ebenfalls beobachtet werden.

Der Grund für dieses Verhalten der *n-best*-Suche ist offensichtlich zunächst einmal ein Anstieg der Einfügungen und der Vertauschungen, wobei beide Fehlerquellen ungefähr in gleichem Maße zur Verschlechterung beitragen. Dies kann damit erklärt werden, daß bei der Verwendung von wortübergreifenden Triphonmodellen die Wortgrenzen nun besser modelliert werden. Da vor allem auch keine Unterscheidung von wortinternen und wortübergreifenden Triphonmodellen getroffen wird, verschlechtert sich andererseits die Modellierung der Triphone im Wortinnern, da nun zur Schätzung dieser Triphone zusätzlich Beobachtungen von den Wortgrenzen herangezogen werden. Die Folge ist, daß nun zwei oder mehr kurze Wörter statt einem längeren Wort u.U. eine bessere akustische Bewertung erhalten können als mit rein wortinternen Triphonen. Die dadurch herbeigeführte Aufspaltung eines Wortes taucht in der Fehlerstatistik als zusätzliche Einfügung plus Vertauschung auf.

Im Abschnitt 6.3.3.7 wird ein Verfahren vorgestellt, mit dem dieser Effekt verringert werden kann.

6.3.3.4 Anzahl der Mischverteilungskomponenten

Der Übergang von rein wortinterner Modellierung zu einer Modellierung mit wortübergreifenden Triphonen bedeutet sicherlich auch eine Erhöhung der Komplexität der die Emissionsverteilungen beschreibenden Mischverteilungen. Da nun die Triphonmodelle nicht nur die akustischen Ereignisse innerhalb von Wörtern, sondern auch an Wortgrenzen beschreiben müssen, könnte eine Erhöhung der akustischen Auflösung durch Verwendung von mehr Mischverteilungen verglichen mit rein wortinternen Modellen eine Verbesserung ergeben. Tabelle 6.9 enthält Ergebnisse für 1 000, 2 000 (Standardwert) und 3 000 Mischverteilungen.

Tabelle 6.9: Wortfehlerrate [%] für verschiedene Anzahlen von Mischverteilungen auf WSJ-5k.

# Modelle	# Dichten (m/w)	DEL-INS[%]	WER[%]
1 000	56k/47k	1.3 - 0.7	6.7
2 000	66k/73k	1.0 - 0.8	6.4
3 000	104k/103k	1.0 - 0.8	6.7

Das Optimum ist hier bei derselben Zahl von Mischverteilungen zu finden wie für das Standardsystem.

6.3.3.5 Aufwand beim ersten Suchdurchlauf

Eine interessante Fragestellung bei der Verwendung der *n-best*-Suche ist, inwiefern der Suchaufwand beim ersten Suchdurchgang mit rein wortinternen Modellen die Fehlerrate beeinflußt. Denn neben der Länge der *n-best*-Liste, die den Umfang des Suchraums der akustischen Neubewertung im zweiten Suchdurchgang festlegt, ist auch die Exaktheit der ersten Suchphase von Bedeutung. Davon hängt letztlich ab, wie gut die durch die erste Suchphase durchgeführte "Vorauswahl" der n besten Sätze ist. Tabelle 6.10 zeigt die Ergebnisse für $n = 10$ und verschieden große Suchräume.
Der Vergleich mit der Fehlerrate für rein wortinterne Modelle, was der Fehlerrate für den ersten Suchdurchgang entspricht, zeigt ein überraschendes Ergebnis. Die Verbesserungen der Fehlerrate durch eine Erhöhung des Suchaufwandes sind für die zweite Suchphase höher als für die erste Suchphase. Z.B. verbessert sich die Fehlerrate bei einer Anhebung des Suchaufwandes von ca. 1200 auf ca. 8700 Zuständen pro Zeitschritt für die erste Suchphase von 7.6% auf 7.0%, was einer relativen Verbesserung von 8% entspricht. Die Fehlerrate für die zweite Suchphase verbessert sich allerdings von 7.0% auf 6.2%, was einer relativen Verbesserung von 11% entspricht. Eine Betrachtung der *n-best*-Fehlerrate, bei der der bezüglich der Levenshtein-Distanz beste Satz der *n-best*-Liste zur Fehlerzählung verwendet wird, zeigt einen entsprechenden Effekt (siehe Tabelle 6.11).
Die Differenzen zwischen den Anzahlen von Fehlern mit ca. 3500 und ca. 8700 Zuständen liegen bei 10-20, was in Tabelle 6.10 ebenfalls beobachtet werden kann. Eine bei gleichlanger *n-best*-Liste verbesserte *n-best*-Fehlerrate, was einer besseren Vorauswahl der Sätze

Tabelle 6.10: Suchaufwand und Wortfehlerrate [%] für wortinterne und wortübergreifende Kontextmodellierung auf WSJ-5k.

	wortinterne Triphone		wortübergr. Triphone	
Zustände	DEL-INS[%]	WER[%]	DEL-INS[%]	WER[%]
1 277	1.3 - 1.0	7.6	1.1 - 1.0	7.0
2 161	1.3 - 0.8	7.2	1.0 - 0.8	6.5
3 532	1.3 - 0.8	7.1	1.0 - 0.8	6.4
5 634	1.3 - 0.7	7.0	1.0 - 0.7	6.3
8 711	1.3 - 0.7	7.0	1.0 - 0.7	6.2
13 008	1.3 - 0.7	7.0	1.0 - 0.7	6.2
18 581	1.3 - 0.7	7.0	1.0 - 0.7	6.2

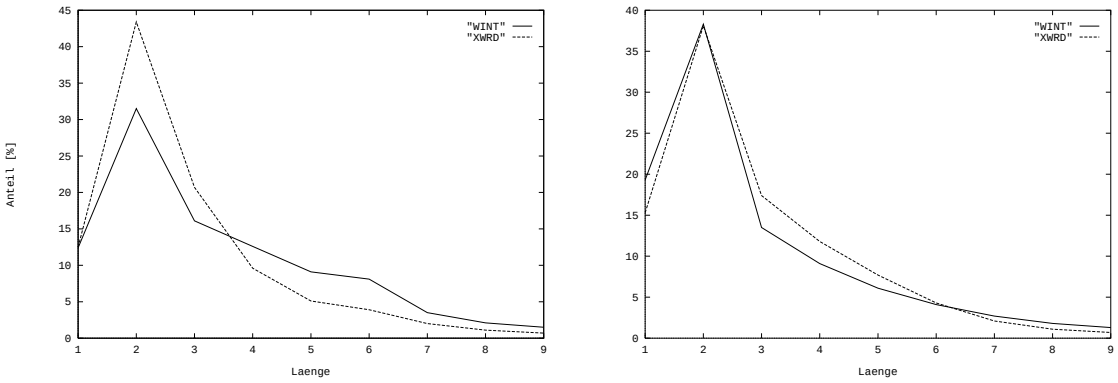
Tabelle 6.11: n-best-Levenshtein-Distanz zum gesprochenen Satz für WSJ-5k.

Suchraum	3532 Zustände pro Satz		8711 Zustände pro Satz	
Wortgraphdichte	37.3		57.7	
n-best	# Fehler	[%]	# Fehler	[%]
1	861	7.1	850	7.0
2	673	5.5	662	5.5
5	490	4.0	478	3.9
10	402	3.3	386	3.2
20	335	2.8	315	2.6
50	251	2.1	234	1.9
100	213	1.8	197	1.6
200	185	1.5	163	1.3
Wortgraph	109	0.9	64	0.5

durch die erste Suchphase entspricht, verbessert also auch die Ergebnisse der zweiten Suchphase um eine entsprechende Zahl von Fehlern.

6.3.3.6 Schätzung der Länge der Zwischenwortpause

Die durch die erste Suchphase geschätzten Pauselängen sind, da sie mit rein wortinternen Modellen ermittelt wurden, u.U. nicht zuverlässig genug, um in der zweiten Suchphase als Basis für die Entscheidung, ob Koartikulation an Wortgrenzen vorliegt oder nicht, verwendet zu werden. Abbildung 6.7 zeigt eine Statistik über die Längen der HMM-Segmente an Wortgrenzen für rein wortinterne und für wortübergreifende Triphone für den WSJ0-Trainingskorpus (siehe Kapitel 4).
Das linke Diagramm zeigt die Verteilung der Längen bzgl. des linken Segments am Wortanfang, das rechte Diagramm die Verteilung der Längen bzgl. des rechten HMM-Segments am Wortende. Tabelle 6.12 enthält die Anteile für Längen ≥ 10.
Bzgl. des *linken* Triphonsegments an einem Wortanfang besitzt die Verteilung der Längen für wortübergreifende Triphonmodelle eine geringere Varianz als die für die unspezifischen Modelle. Obwohl für die *rechten* Triphonsegmente an einem Wortende beide Verteilun-



Linkes Segment an Wortgrenze

Rechtes Segment an Wortgrenze

Abbildung 6.7: Länge der HMM-Segmente an Wortgrenzen für WSJ-5k

Tabelle 6.12: Anteil der HMM-Segmentlänge an Wortgrenzen für Längen ≥ 10 für WSJ-5k.

Wortgrenze	Modelle	Anteil[%]
links	wortintern	3.0
	wortübergreifend	1.0
rechts	wortintern	3.9
	wortübergreifend	1.6

gen praktisch identisch sind, sollte die Schätzung der Länge einer Zwischenwortpause mit wortübergreifenden Triphonmodellen besser als mit rein wortinternen Modellen gelingen. Als Folge sollte dem Algorithmus, der die akustische Neubewertung des Satzes vornimmt, überlassen werden, ob er an einer Wortgrenze Koartikulation annimmt oder nicht. Daher wurde der Zeitanpassungsalgorithmus entsprechend modifiziert (siehe Anhang). Tabelle 6.13 zeigt die Fehlerraten mit dem hier beschriebenen modifizierten Verfahren, im folgenden auch *dynamische* Pauseschätzung genannt.

Tabelle 6.13: Wortfehlerrate [%] für n-best-Rescoring für dynamische Pauseschätzung auf WSJ-5k.

n-best	DEL-INS[%]	WER[%]
2	1.2 - 0.8	6.7
5	1.0 - 0.7	6.3
10	1.0 - 0.8	6.3
20	1.0 - 0.8	6.3
50	1.0 - 0.8	6.4
100	1.0 - 0.8	6.4
200	1.0 - 0.8	6.4

Der Gewinn durch die exaktere Pauseschätzung bzgl. Fehlerrate ist gering. Im Optimum bzgl. n beträgt die Differenz lediglich 7 Fehler. Für große n ist der Gewinn durch die bes-

sere Modellierung größer, hier gewinnt man ca. 20 Fehler gegenüber der Pauseschätzung aus der ersten Suchphase. Obwohl diese Verbesserungen nicht als signifikant einzustufen sind, wurde dieses Verfahren als Standardmethode für die weiteren Tests in das System übernommen, da i.A. bei ähnlichem Aufwand bzgl. Zeit- und Platzkomplexität eines Algorithmus die theoretisch "exaktere" Methode zu verwenden ist.
Dieses Verfahren wurde ebenfalls auf dem VM-5k-Korpus getestet. Tabelle 6.14 enthält die dazu gehörigen Fehlerraten.

Tabelle 6.14: Wortfehlerrate [%] für n-best-Rescoring für dynamische Pauseschätzung auf VM-5k.

n-best	DEL-INS[%]	WER[%]
5	4.1 - 3.8	21.1
10	4.1 - 4.0	20.8
20	3.9 - 4.0	20.6
50	3.9 - 4.0	20.6
100	3.9 - 3.9	20.5
200	3.7 - 3.9	20.2
500	3.7 - 4.2	20.5

Hier läßt sich wie auf dem WSJ-5k-Korpus ein Minimum der Fehlerrate für ein bestimmtes n, hier $n = 200$ beobachten. Für größere n-best-Listen steigt die Fehlerrate wieder an. Diese Beobachtung entspricht der auf dem WSJ-Korpus, so daß das weiter oben zu diesem Thema gesagte auch hier Gültigkeit hat.

6.3.3.7 Interpolation mit wortinternen Triphonmodellen

In dem oben beschriebenen Basisverfahren wird nicht unterschieden, ob ein Triphon an einer Wortgrenze auftritt oder nicht. Dies ist bzgl. der Robustheit der Modelle, sprich der Anzahl der Trainingsdaten pro Modell, sicher zu begrüßen. Allerdings bringt diese Art der Modellierung auch Probleme mit sich. Vor allem an Wortenden treten bei vielen Wörtern Varianten auf, z.B. Verschlucken von Phonemen. Dies kann, wenn die Triphone an Wortgrenzen nicht von den wortinternen Triphonen getrennt werden, zu Problemen führen. Ein einfaches Verfahren, das gerade für die *n-best*-Suche einfach zu realisieren ist, ist die Interpolation zwischen den Satzbewertungen von erster und zweiter Suchphase.
Dabei wird die Entscheidung, welcher der Sätze der *n-best*-Liste als erkannter ausgegeben wird, nicht allein aufgrund der Neubewertung in der zweiten Suchphase mit Wortgrenzenmodellen getroffen. Statt dessen werden die Bewertungen *beider* Suchphasen mit einem konstanten Faktor interpoliert. Diese interpolierte Bewertung wird dann zur Entscheidung über den besten Satz verwendet. Da die beiden akustischen Modelle, das rein wortinterne und das Modell mit Wortgrenzenmodellierung, die akustischen Ereignisse innerhalb eines Wortes und an Wortgrenzen verschieden gut modellieren, könnte eine Interpolation zwischen den Bewertungen durch beide Modelle eine verbesserte Gesamtbewertung zur Folge haben. Die Interpolation wird durch einen Faktor λ mit folgender Formel für die Bewertung der ersten Suchphase Q_{within} und die der zweiten Suchphase Q_{across} berechnet:

$$Q_{int} = (1 - \lambda)Q_{within} + \lambda Q_{across}$$

D.h. für $\lambda = 0$ gehen nur die rein wortinternen Modelle in die Bewertung ein, während für $\lambda = 1$ nur die wortübergreifenden Triphonmodelle verwendet werden. Tabelle 6.15 zeigt die Ergebnisse für verschiedene Interpolationsfaktoren λ.

Tabelle 6.15: n-best-Rescoring für Score-Interpolation zwischen Modellen mit rein wortinternen und mit wortübergreifenden Triphonen für $n = 20$ und `LMScale = 18` auf WSJ-5k.

λ	WSJ-5k		VM-5k	
	DEL-INS[%]	WER[%]	DEL-INS[%]	WER[%]
0.0	1.4 - 0.7	7.1	4.5 - 3.7	21.9
0.1	1.3 - 0.7	6.7	4.3 - 3.8	21.4
0.2	1.2 - 0.6	6.4	4.2 - 3.8	21.1
0.3	1.2 - 0.6	6.3	4.0 - 3.7	21.0
0.4	1.1 - 0.6	6.2	4.0 - 3.6	20.3
0.5	1.1 - 0.6	6.2	3.9 - 3.6	20.1
0.6	1.1 - 0.7	6.2	3.9 - 3.7	20.0
0.7	1.0 - 0.7	6.1	3.9 - 3.7	20.1
0.8	1.0 - 0.7	6.2	3.9 - 3.9	20.3
0.9	1.0 - 0.7	6.3	3.9 - 3.9	20.3
1.0	118 - 0.7	6.3	3.9 - 3.9	20.5

Durch die Interpolation kann bei geeigneten Werten für λ eine geringfügige Verbesserung der Fehlerrate erreicht werden. Für den WSJ-5k-Korpus erreicht man bei $n = 20$ und $\lambda = 0.7$ eine Verbesserung von 6.3% auf 6.1%, für den VM-5k-Korpus von 20.5% auf 20.1%, also ca. 2-3% relativ. Eine interessante Eigenschaft der Interpolation zeigt Tabelle 6.16.

Tabelle 6.16: Score-Interpolation zwischen Modellen mit rein wortinternen und mit wortübergreifenden Triphonen für $\lambda = 0.7$ und `LMScale = 18` auf WSJ-5k.

n-best	DEL-INS[%]	WER[%]
2	1.2 - 0.7	6.7
5	1.0 - 0.7	6.2
10	1.0 - 0.7	6.2
20	1.0 - 0.7	6.1
50	1.0 - 0.7	6.1
100	1.0 - 0.7	6.1
200	1.0 - 0.7	6.1

Bei Verwendung des Interpolationsfaktors $\lambda = 0.7$ zeigt sich, daß für steigende n der in den bisherigen Versuchen beobachtete Effekt einer ansteigenden Fehlerrate nicht mehr auftritt. Statt dessen konvergiert die Fehlerrate für große n zu einem Minimum. Beim WSJ-5k-Korpus erhält man dieses Minimum bei 6.1% und 734 Fehlern.

Tabelle 6.17: Wortfehlerate [%] für die Einbeziehung des gesprochenen Satzes bei der *n-best*-Suche für $n = 200$, `LMScale = 18` und $\lambda = 0.7$ für WSJ-5k.

Einbeziehung des gesprochenen Satzes	DEL-INS[%]	WER[%]
nein	1.0 - 0.7	6.1
ja	1.0 - 0.6	5.8

6.4 Erkennung: Einphasige Suche

Der Vorteil der *n-best*-Suche besteht vor allem darin, daß das Verfahren prinzipiell einfach ist, und daß sich infolgedessen der Implementierungsaufwand in Grenzen hält. Andererseits führt der zweiphasige Aufbau der Suche dazu, daß aufgrund des durch das Prinzip der *n-best*-Suche stark eingeschränkten Suchraums i.A. nicht der Satz mit der besten Bewertung bzgl. der wortübergreifenden Triphone gefunden wird. Tabelle 6.17 zeigt, daß bei der Einbeziehung des gesprochenen Satzes in die *n-best*-Liste die Fehlerrate verbessert weiter werden kann. Weiterhin zeigt Tabelle 6.11, daß selbst bei einer *n-best*-Liste der Länge 200 die akkumulierte Levenshtein-Distanz fast um den Faktor 3 größer ist als auf dem entsprechenden Wortgraphen. Dies läßt vermuten, daß bei einer einphasigen Suche, bei der die wortübergreifenden Triphonmodelle direkt in der wortabhängigen Baumsuche verwendet werden und dadurch eine wesentlich bessere Abdeckung des Suchraums erreicht wird, die Fehlerrate weiter verbessert werden kann. Allerdings zeigen die Versuche mit verschiedenen n für die *n-best*-Suche (siehe Tabelle 6.8), wo für steigende n ab einer bestimmten Grenze die Fehlerrate wieder ansteigt, daß dies nicht ohne weiteres zu erreichen ist, wie im weiteren dargestellt werden wird. Eine Lösungsmöglichkeit besteht in der im vorigen Abschnitt beschriebenen Interpolation, die für die *n-best*-Suche eine Verringerung dieses Effektes erreicht hat. Wie sich im folgenden zeigen wird, ist dieses einfache Verfahren auch für die einphasige Suche mit wortübergreifenden Triphonmodellen einsetzbar.

6.4.1 Basisverfahren

Der zu Beginn des Kapitels 6 kurz vorgestellte Ansatz, mit dem die wortabhängige Baumsuche für wortübergreifende Triphone erweitert werden kann, soll hier noch einmal ausführlich dargestellt werden.
Das prinzipielle Problem bei der Verwendung von wortübergreifenden Triphonen für die wortabhängige Baumsuche besteht darin, daß je nach Wortpaar vw sowohl die Modelle am Wortende von v als auch die Modelle am Wortanfang von w spezifisch für dieses Wortpaar sind. D.h. beginnt das Wort w mit einem Phonem α, müssen die Triphonmodelle am Wortende von v als rechten Kontext das Phonem α verwenden. Genauso müssen die Modelle am Anfang von w das jeweilige Endphonem von v als linkes Kontextphonem verwenden. Dies setzt z.T. erhebliche Modifikationen am Baumlexikon voraus, auf die nun genauer eingegangen werden soll.
Die Berücksichtigung des *linken* Phonemkontextes am Wortanfang kann dadurch erreicht werden, daß beim Start eines neuen Baums die Identität des Vorgängerphonems mit berücksichtigt wird. Dies ist bei der Verwendung einer wortabhängigen Bigrammsuche

ohne Aussprachevarianten prinzipiell durch die Verwendung der Baumkopien bzgl. des Vorgängerwortes schon gegeben. Bei der zusätzlichen Verwendung von Aussprachevarianten, bei denen die Wortenden trotz gleichem lexikalischen Wort variieren können, muß dieses Vorgängerphonem explizit mitpropagiert werden. Die Abhängigkeit des Baumlexikons von diesem Vorgängerphonem kann nun durch die Verwendung eines *generischen* Baumlexikons, bei dem die erste Phonemgeneration abhängig vom linken Kontextphonem modifiziert wird, realisiert werden (siehe Abbildung 6.8).

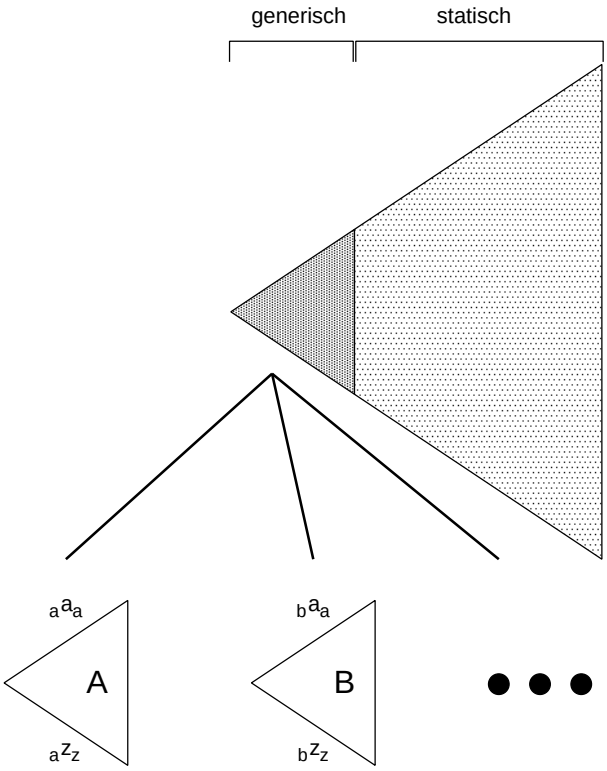


Abbildung 6.8: Generische erste Phonemgeneration

Die erste Generation des lexikalischen Baums ist generisch (dunkelgrau), d.h. für diesen Teil des Baums können die jeweils passend zum letzten Phonem des Vorgängerwortes modifizierten ersten Generationen eingesetzt werden (untere kleinere Bäume). Diese sind mit dem Vorgängerphonem (a, b, ...) indiziert, und alle Triphone dieser ersten Generation besitzen als linken Kontext genau dieses Phonem (a, b, ...).
Alternativ könnten für jeden Ast der ersten Phonemgeneration m Äste mit gleichem Unterbaum in das Baumlexikon eingefügt werden, wobei m der Anzahl der möglichen linken Kontextphoneme entspricht. Dies würde weniger Modifikationen des Suchalgorithmus erfordern, allerdings würde die Größe des Baumlexikons um den Faktor m steigen, was bei großem Vokabular nicht mehr realisierbar wäre. Bei der Verwendung einer generischen ersten Phonemgeneration steigt dagegen die Lexikongröße nur um eine additive Größe von $(m-1)p$, wobei p der Anzahl der Äste der ersten Phonemgeneration entspricht.
Bei der Behandlung der Wortenden besteht die Schwierigkeit, daß die Identität des Nachfolgewortes und damit des rechten Kontextphonems zu einem Wortende nicht bekannt ist. Dies kann aber dadurch erreicht werden, indem die Äste der Wortenden bezüglich der möglichen rechten Kontextphoneme aufgefächert werden (siehe Abbildung 6.9).

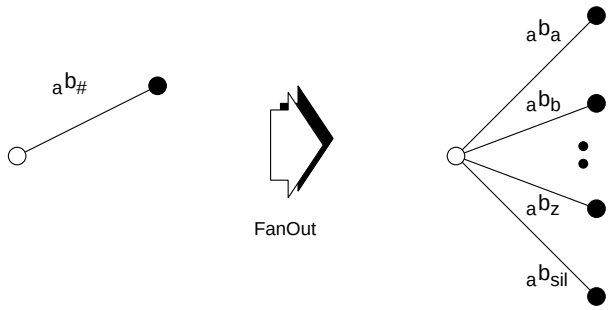


Abbildung 6.9: Fan-Out des lexikalischen Baums an Wortenden

Jeder dieser Äste beschreibt nun die akustische Realisierung des Wortendes mit dem jeweiligen rechten Kontextphonem bzw. dem Pausekontext. Die damit erreichte Konditionierung des Wortendes bezüglich des rechten Kontextes geht allerdings einher mit einer starken Vergrößerung des Baumlexikons. Bei einem Lexikon mit 20 000 Einträgen und m möglichen rechten Kontexten vergrößert sich das Baumlexikon um $20\,000 \cdot m$ Äste, wodurch nicht nur der Aufwand für die Darstellung des Baumlexikons im Speicher ansteigt, sondern auch der Suchraum in der letzten Phonemgeneration wächst um einen Faktor m. Allerdings kann aufgrund des nun an einem Wortende bekannten rechten Kontextes für das folgende Wort im Falle eines "echten" Phonems nur ein Teilbaum gestartet werden, nämlich der Teilbaum, dessen Phonem in der ersten Astgeneration dem rechten Kontextphonem des aufgefächerten Wortendes entspricht. Für den rechten Kontext "Pause" muß allerdings immer noch ein kompletter Folgebaum gestartet werden.
Durch eine Methode ähnlich der für die erste Baumgeneration kann auch für den Fan-Out eine erhebliche Reduktion des Speicherplatzbedarfs erreicht werden. Da die in einem konkreten Fan-Out-Ast enthaltenen akustischen Modelle für den betrachteten Phonemkontext nur vom vorletzten und letzten Phonem eines Wortes abhängt, sind maximal n^2 verschiedene Fan-Out-Äste denkbar. Diese können in einer separaten Liste abgelegt werden, der eigentliche lexikalische Baum enthält in der letzten Phonemgeneration lediglich eine Referenz auf den passenden Fan-Out-Ast in dieser Liste. Die Zahl der notwendigen Äste in der letzten Phonemgeneration beträgt z.B. für ein 20 000-Wort-Lexikon und 43 Phoneme statt 860 000 Ästen mit dieser Methode nur noch maximal 80 000 Äste, also ca. 10% (siehe Abbildung 6.10). Diese Verbesserung wurde allerdings für die in dieser Arbeit durchgeführten Tests noch nicht verwendet.
Die Rekombination an den Wortgrenzen geschieht nun ebenfalls nicht nur in Abhängigkeit vom Vorgängerwort, sondern auch in Abhängigkeit vom unmittelbaren linken Phonemkontext und vom Phonem der ersten Astgeneration. Dabei werden für die Bäume, deren linker Kontext nicht "Pause" ist, die Wortenden mit passendem Wortindex, passendem Phonem in der letzten Astgeneration und passendem rechten Kontextphonem rekombiniert (siehe Abbildung 6.11).
Für die Bäume, deren linker Kontext "Pause" ist, wird die Rekombination analog zur Rekombination mit rein wortinternen Triphonen durchgeführt.
In der Basisimplementierung des Verfahrens werden keine weiteren Einschränkungen des Algorithmus vorgenommen. Insbesondere wird das Einfügen von Pausen zwischen den Wörtern zugelassen unabhängig von der gewählten Alternative bei der Auffächerung.
Im Gegensatz dazu wird beim *n-best*-Verfahren im Falle der Alternative "keine Koarti-

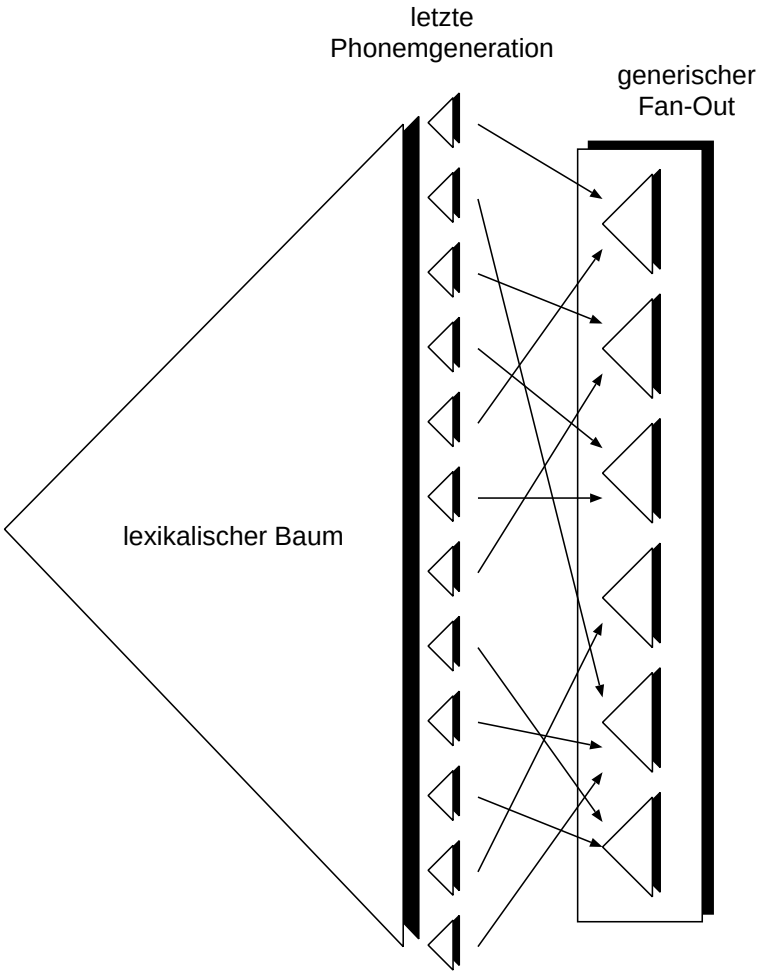


Abbildung 6.10: Verwendung von generischen Fan-Out-Ästen in der letzten Phonemgeneration

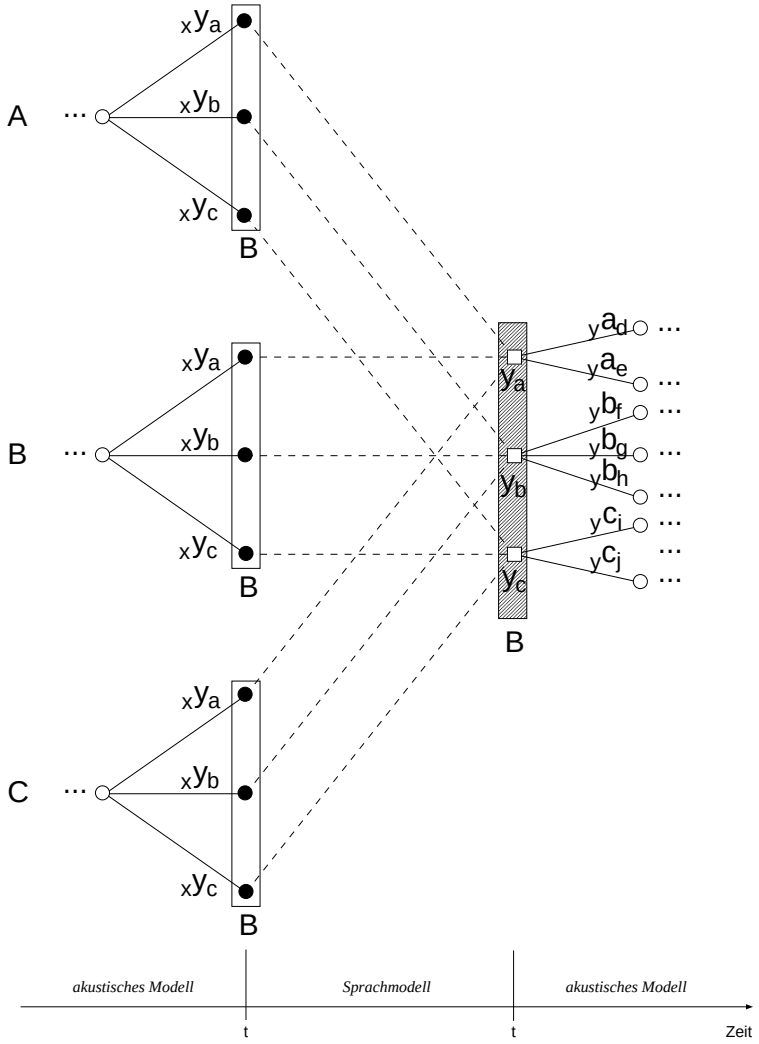


Abbildung 6.11: Wortgrenzenrekombination für wortübergreifende Triphone

kulation" die Verwendung von mindestens t Pausezuständen bei einem Pauseschwellwert von t erzwungen. Dies wurde in der Basisimplementierung zunächst vermieden, um eine möglichst einfache Struktur des Suchalgorithmus zu erhalten. Untersuchungen, die die Erkenntnisse in Bezug auf die *n-best*-Suche einbeziehen, werden in Abschnitt 6.4.2.1 vorgestellt.

6.4.2 Ergebnisse

Für die in diesem Abschnitt dargestellten Ergebnisse gelten dieselben Randbedingungen wie für die Ergebnisse der *n-best*-Suche (siehe Abschnitt 6.3.3).
Tabelle 6.18 zeigt die Fehlerrate auf WSJ-5k und VM-5k für die einphasige Suche, verglichen mit dem Ergebnis der *n-best*-Suche auf diesen Korpora (siehe auch [Elting 99]).
Offensichtlich sind die Ergebnisse der einphasigen Suche mit wortübergreifenden Triphonen in der oben beschriebenen Variante nur leicht besser als die der einphasigen Suche mit rein wortinternen Modellen. Dies kann zum einen auf die im Vergleich zur *n-best*-Suche ungenaue Modellierung der Wortgrenzen in Bezug auf die zu verwendende Zwischenwort-

Tabelle 6.18: Wortfehlerrate [%] für die einphasige Suche auf WSJ-5k und VM-5k.

Korpus	Methode	DEL-INS[%]	WER[%]
WSJ-5k	wortintern	1.3 - 0.7	7.1
	n-best	1.0 - 0.8	6.3
	einphasig	1.1 - 0.8	6.8
VM-5k	wortintern	4.4 - 3.7	21.9
	n-best	3.9 - 3.9	20.5
	einphasig	3.4 - 5.2	21.7

pause zurückgeführt werden. Andererseits bestätigt sich dadurch das Bild, das man in der *n-best*-Suche bei steigendem n erhalten hat. Dort stiegen die Fehlerraten ab einem optimalen n wieder an, hier erhält man für $n = \infty$ ein nur leicht gegenüber den rein wortinternen Modellen verbessertes Ergebnis. Bei der *n-best*-Suche ließ sich dieser Effekt weitgehend durch die Interpolation mit den rein wortinternen Modellen unterdrücken. Dies wird für die einphasige Suche im nächsten Abschnitt untersucht.

6.4.2.1 Übertragung der Erkenntnisse aus der n-best-Suche

In diesem Abschnitt sollen die aus der *n-best*-Suche gewonnenen Erkenntnisse auf die einphasige Suche übertragen werden, soweit dies möglich und sinnvoll ist. Folgende Optimierungen bieten sich daher an:

- Erzwingung der korrekten Pauselängen zwischen den Wörtern
- Interpolation der wortübergreifenden Triphonmodelle mit rein wortinternen Modellen

6.4.2.2 Erzwingung der korrekten Pauselänge

Die Erzwingung der korrekten Pauselängen zwischen den Wörtern soll den Suchalgorithmus derart einschränken, daß für einen Koartikulationsschwellwert N_{sil}

- zwischen Wörtern, zwischen denen keine Koartikulation angenommen wird, minimal N_{sil} Pausezustände angenommen werden,
- zwischen Wörtern, zwischen denen Koartikulation angenommen wird, maximal $N_{sil} - 1$ Pausezustände angenommen werden und

Diese Einschränkungen wurden folgendermaßen berücksichtigt:

1. Für Wörter, zwischen denen keine Koartikulation angenommen wird, wird der Fan-Out-Ast um N_{sil} Pausezustände verlängert, zwischen denen nur Übergänge in den nächsten Zustand erlaubt sind (siehe Abbildung 6.12).

2. Um zwischen Wörtern, zwischen denen Koartikulation angenommen wird, eine maximale Pause von $N_{sil} - 1$ zu erzwingen, wird für die Fan-Out-Hypothesen, die eine Koartikulation modellieren, die Zwischenwort-Pauselänge ermittelt. Falls diese größer oder gleich N_{sil} ist, wird diese Hypothese geprunt.

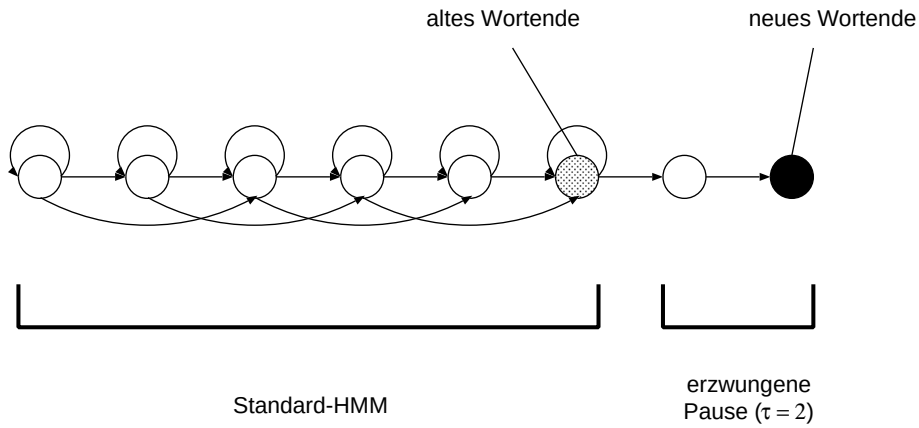


Abbildung 6.12: Erzwungene Pause der Länge N_{sil} für Nicht-Koartikulation

Diese Modifikation wurde in zwei Schritten eingeführt. Im ersten Schritt wurden lediglich die Pausezustände an den Wortenden hinzugefügt. Damit entsprach die Modellierung der Wortgrenzen im wesentlichen der Modellierung der n-best-Suche mit Bestimmung des Übergangs in der zweiten Suchphase (siehe Tabelle 6.14). Die Ergebnisse für diesen Schritt sind in Tabelle 6.19 aufgelistet. Die Fehlerrate verbessert sich für den VM-5k-Korpus von 21.7% auf 21.4% für einen Pauseschwellwert von 1, wobei die Verbesserungen im wesentlichen auf eine Reduzierung der Einfügungen zurückzuführen sind. Dies entspricht den Erwartungen, da eine Erzwingung einer Mindestpausenlänge die Verwendung des Nicht-Koartikulationsübergangs für eine Aufspaltung von Wörtern (z.B. "geh hin" statt "gehen") erschwert. Für den WSJ-5k-Korpus ist dieser positive Effekt nicht zu beobachten. Hier ist die Fehlerrate gleich der Fehlerrate ohne die Verwendung einer minimalen oder maximalen Pausedauer.

Tabelle 6.19: Wortfehlerrate [%] für Erzwingung der minimalen bzw. maximalen Zwischenwortpause auf WSJ-5k und VM-5k ($N_{sil} = 1$).

Korpus	Art der Einschränkung	DEL-INS[%]	WER[%]
WSJ-5k	keine	1.1 - 0.8	6.8
	minimal	1.1 - 0.8	6.8
	minimal & maximal	1.1 - 0.8	6.8
VM-5k	keine	3.4 - 5.2	21.7
	minimal	3.3 - 4.6	21.4
	minimal & maximal	3.3 - 4.7	21.5

Die Ergebnisse für die Verwendung beider Einschränkungen hinsichtlich der Zwischenwortpause sind nicht besser als die Ergebnisse, bei denen lediglich die minimale Pauselänge

für Nicht-Koartikulationsübergänge erzwungen wurde. Daher wurde in den weiteren Tests nur die erste Einschränkung verwendet.

6.4.2.3 Interpolation

Eine wichtige Verbesserung bei der *n-best*-Suche war die Interpolation zwischen den Satzbewertungen der Suche mit rein wortinternen Modellen und der Neubewertung mit Wortgrenzenmodellen. Neben einer leichtn Verbesserung der Fehlerrate konnte außerdem beobachtet werden, daß der Effekt der wiederansteigenden Fehlerrate bei großen n nicht mehr auftrat. Daher kann vermutet werden, daß sich eine solche Interpolation auch bei der einphasigen Suche mit wortübergreifenden Triphonen positiv auswirken wird, da hier der abgesuchte Suchraum einem n nahe unendlich bei der *n-best*-Suche entspricht.
Für die einphasige Suche wird die Interpolation zustandsweise durchgeführt, d.h. für jede aktive Hypothese werden zwei lokale Bewertungen berechnet, eine mit wortinternen Modellen und eine mit Wortgrenzenmodellen. Die Ergebnisse zu dieser Modifikation, für die eine minimale Zwischenwortpause von 1 vorgegeben war (siehe oben), finden sich in Tabelle 6.20.

Tabelle 6.20: Wortfehlerrate [%] für Interpolation auf WSJ-5k und VM-5k.

Korpus	λ	DEL-INS[%]	WER[%]
WSJ-5k	1.0	1.0 - 0.8	6.8
	0.7	1.0 - 0.6	6.2
VM-5k	1.0	3.3 - 4.9	21.4
	0.7	3.5 - 4.2	20.2

Die Verbesserungen durch die Interpolation sind dramatisch, man erreicht auf dem WSJ-5k-Korpus statt 6.8% nun 6.2% und VM-5k-Korpus statt 21.4% nun 20.2%. Die mit der *n-best*-Suche erreichten Fehlerraten liegen mit 6.1% bzw. 20.1% allerdings immer noch etwas darunter. Trotzdem zeigt auch dieses Ergebnis, daß bei der Verwendung von wortübergreifenden Triphonmodellen eine undifferenzierte Vermischung von wortinternen Triphonen und Wortgrenzentriphonen zu suboptimalen Resultaten führt. Durch die Interpolation wird vor allem die Modellierung der wortinternen Triphone verbessert, da die rein wortinternen Triphonmodelle zwischen Wortinnerem und Wortgrenze eindeutig unterscheiden, während die wortübergreifenden Triphonmodelle ein Triphon unabhängig von seiner Position mit derselben Mischverteilung modellieren.

6.5 Zusammenfassung

In diesem Kapitel wurde die Integration der Wortgrenzenmodellierung mit wortübergreifenden Triphonmodellen in das am Institut verwendete Spracherkennungssystem untersucht. Dabei wurden verschiedene Aspekte sowohl für die akustische Modellierung der Wortgrenzenkoartikulation als auch die verwendeten Verfahren für die Suche betrachtet. Für die akustische Modellierung waren dies

- Schwellwert N_{sil} für die Zwischenwortpause,
- Anzahl der Mischverteilungen,
- Schätzung von getrennten Modellen für Zwischenwortpausen,
- Interpolation zwischen rein wortinternen Modellen und Wortgrenzenmodellen.

Weiterhin wurden für die *n-best*-Suche verschiedene Aspekte des Suchalgorithmus untersucht. Dies waren

- Länge der *n-best*-Liste,
- Bestimmung der Zwischenwortpauselänge im ersten oder zweiten Suchdurchgang,
- Einfluß des Suchaufwands im ersten Suchdurchgang auf die Fehlerrate,
- Hinzufügung des gesprochenen Satzes zur *n-best*-Liste.

Die durch die Wortgrenzenmodellierung erzielten Verbesserungen auf den verschiedenen Korpora lagen zwischen 10% und 15% und sind damit mit den von anderen Autoren berichteten Verbesserungen vergleichbar (siehe auch Kapitel 2). Zusammen mit dem State-Tying mit phonetischen Entscheidungsbäumen wurde die Fehlerrate insgesamt auf den verschiedenen Korpora um mindestens 20% gesenkt. Weitere Betrachtungen und Ergebnisse finden sich in [Elting 99].

Kapitel 7

Automatische Fragengenerierung

Neben der Erzielung einer möglichst geringen Fehlerrate ist ein sekundäres Ziel in der automatischen Spracherkennung, möglichst viel von dem Wissen, das in die Erkennung der Sprache eingeht, automatisch aus den Trainingsdaten zu lernen. Damit sind nicht nur die Parameter des Spracherkenners gemeint, also z.B. Mittelwerte der Mischverteilungskomponenten, sondern auch die Strukturen selbst, die zur Konstruktion des Erkenners verwendet werden. Durch das automatische Lernen dieser Strukturen sollen zwei Ziele erreicht werden:

- Durch eine optimale Anpassung der Strukturen des Spracherkennungssystems an die spezielle Aufgabe soll die Erkennungsgenauigkeit optimiert werden.

- Durch das *automatische* Lernen der Strukturen kann die Anpassung von Spracherkennungssystemen an neue Aufgabenstellungen wesentlich schneller geschehen.

Beispielsweise wird in [Takami & Sagayama 92] ein Verfahren beschrieben, mit dem die Struktur der HMM für die akustische Modellierung gelernt werden kann. Damit konnte die Fehlerrate auf einem Korpus zur Phonemerkennung im Vergleich zur Verwendung von Standard-HMM um ca. 7% relativ gesenkt werden. Allerdings war die Topologie der zum Vergleich verwendeten Standard-HMM nicht optimiert, so daß nicht klar ist, ob die Verwendung dieses Verfahrens in erster Linie zu Verbesserungen der Fehlerrate führt oder vor allem eine datengetriebene Generierung von HMM-Topologien erlaubt, die ähnliche Fehlerraten wie von Hand optimierte HMM ermöglichen.
Beim State-Tying mit phonetischen Entscheidungsbäumen existieren ebenfalls verschiedene Parameter und Strukturen, die ad hoc von einem Experten definiert werden müssen bzw. durch *try and error* optimiert werden. Zu diesen gehört z.B. die optimale Anzahl der Mischverteilungen, die für das State-Tying mit phonetischen Entscheidungsbäumen äquivalent ist mit der Anzahl der Blätter der Entscheidungsbäume. Ein weiteres Beispiel für eine Struktur, die auf Expertenwissen zurückgreift, ist die Menge der phonetischen Fragen, die bei der Konstruktion des phonetischen Entscheidungsbaums eingeht. Für diese Fragenmenge soll nun im folgenden ein Verfahren vorgestellt werden, mit dem auf effiziente Weise automatisch phonetische Fragenmengen erzeugt werden können, die zu gleichen oder besseren Fehlerraten führen verglichen mit der Verwendung von Expertenfragen.

7.1 Einführung

Die bei der Konstruktion der phonetischen Entscheidungsbäume verwendeten phonetischen Fragen wurden bereits in Kapitel 5 angesprochen. Eine phonetische Frage im Zusammenhang mit Entscheidungsbäumen besteht aus einer Menge von Phonemen und dem Kontext, für den diese Frage Triphone klassifizieren soll, also entweder "rechter Kontext" oder "linker Kontext" (die im Kapitel 5 angesprochene Methode der Verwendung von Fragen an das zentrale Phonem und das Segment soll hier nicht betrachtet werden). Die Menge von Phonemen einer phonetischen Frage legt dabei fest, für welche Kontextphoneme eines Triphons die Frage mit "Ja" beantwortet wird und für welche nicht.
Beispielsweise würde für eine phonetische Frage mit den Phonemen *aa, ah, eh* und dem Kontext "links" für das Triphon ${}_{aa}p_s$ die Frage mit "Ja" beantwortet werden (das Modell zu dem jeweiligen Triphonzustand würde sich im linken Teilbaum befinden), für ein Triphon ${}_{s}p_{aa}$ würde die Antwort auf die phonetische Frage "Nein" lauten.
Die für das State-Tying mit phonetischen Entscheidungsbäumen verwendeten phonetischen Fragen entsprechen i.A. phonetischen *Klassen* (siehe auch Abschnitt 5.3.2). Diese eignen sich deshalb besonders gut als phonetische Fragen, weil sie jeweils Phoneme enthalten, deren Artikulation bestimmte Gemeinsamkeiten aufweist. Beispiele für solche Phonemklassen sind

- stimmhaft oder stimmlos

  Einige Phoneme, z.B. Vokale, werden unter Verwendung der Stimmbänder erzeugt, d.h. bei einem stimmhaften Laut schwingen die Stimmbänder bei der Erzeugung mit, bei stimmlosen Lauten nicht.

- Plosive

  Bestimmte Laute, z.B. p oder t, werden gebildet, indem die beteiligten Artikulatoren den Weg des Luftstromes kurzzeitig verschließen, um ihn dann wieder freizugeben, was einen "explosionsartigen" Lauteindruck hervorruft.

- Frikative

  Laute wie f oder s werden gebildet, indem die Artikulatoren den Weg des Luftstromes so verengen, daß an der Verengungsstelle die Luft mit hoher Geschwindigkeit fließt und so Turbulenzen bildet, die einen zischenden Lauteindruck bewirken.

D.h. die Phoneme, die ausschließlich durch die in Kapitel 5 erwähnte Minimalpaaranalyse definiert worden sind, also als informationstragende Lauteinheiten identifiziert worden sind, werden nun zusätzlich nach der Art und Weise ihrer Artikulation kategorisiert.
Ein Beispiel für eine solche auf phonologischer Kategorisierung beruhende Fragenliste findet sich z.B. in [Hon 92]:

```
...
DENTAL        dh th
LIQUID        l r ur
S/SH          s sh
```

```
NASAL        m n
ALVEOLAR     n d t s z un
DIPHTHONG    aw awh ey ow
VOWEL        ae ee eh ih UH uh ah aa ...
...
```

Einige der Fragen beschreiben relativ spezielle Eigenschaften von Phonemen (i.A. kurze Phonemliste, Beispiel "Nasal"), während andere Fragen eine breitere Klassifikation der Phoneme vornehmen (lange Phonemliste, Beispiel "Vowel"). Tabelle 7.1 zeigt für den WSJ-5k-Korpus eine typische Aufstellung der zehn wichtigsten Fragen (Kriterium für die Wichtigkeit einer Frage ist hierbei der mittlere Gewinn an Log-Likelihood pro Beobachtung durch die Frage).

Tabelle 7.1: 10 Fragen mit maximalem Log-Likelihood-Gewinn pro Auftreten für das linke Phonemsegment auf WSJ-5k.

Frage	# Verwendung	rel. Log-Likelihood-Gewinn
ORAL-STOP1	14	4.9
VOWEL	26	4.7
SONORANT	72	4.6
LAX-VOWEL	20	3.5
S/Z/SH/ZH	42	3.2
LIQUID	30	2.8
TENSE-VOWEL	22	2.5
R-LABIAL	44	2.2
PALATL	34	2.1
LIQUID-GLIDE	20	2.0

Hier fällt auf, daß sowohl die allgemeineren Fragen wie "Vowel" oder "Sonorant" als auch spezielle Fragen wie "Liquid" oder "S/Z/SH/ZH" besonders wichtig für die Erzeugung des phonetischen Entscheidungsbaums zu sein scheinen. Insgesamt sind diese 10 wichtigsten von insgesamt 88 Fragen für ungefähr ein Drittel des gesamten Gewinns an Log-Likelihood durch die Konstruktion des Entscheidungsbaums verantwortlich. Dies zeigt, daß zwischen den phonetischen Klassen, die im Kontext eines Phonems auftreten können, Unterschiede in der Beeinflussung der akustischen Realisierung dieses Phonems durch diese Klassen bestehen. Einige Klassen wie z.B. Vokale oder Liquide scheinen die akustische Realisierung eines Phonems stärker zu beeinflussen als andere wie z.B. Nasale. Dies kann als ein erster Hinweis darauf verstanden werden, daß ein Verfahren, das Phoneme automatisch zu phonetischen Fragen zusammenfaßt und dabei die Bedeutung dieser Phoneme in einem phonetischen Kontext berücksichtigt (was durch die phonetische Klassifikation durch einen Experten i.A. nicht getan wird), u.U. zu besseren Ergebnissen sowohl beim Log-Likelihood-Gewinn bei der Konstruktion des Baums als auch bei der Fehlerrate führen kann.

7.2 Problemstellung

Wie in Kapitel 5 bereits erwähnt, besitzt das State-Tying mit phonetischen Entscheidungsbäumen den Nachteil, daß die zur Konstruktion des phonetischen Entscheidungsbaums notwendigen phonetischen Fragen manuell definiert werden müssen. Dabei fließt Expertenwissen über die Artikulation ein, nach denen die Phoneme gruppiert werden. Diese Vorgehensweise ist aus folgenden Gründen problematisch:

- Nicht immer ist ein Experte vorhanden, der die phonetischen Fragen definieren kann.
- Die Definition der Fragen mit Expertenwissen ist nicht unbedingt optimal bzgl. der Problemstellung des State-Tying mit phonetischen Entscheidungsbäumen.
- Bei der Verwendung eines neuen Korpus muß Zeit in die Definition der phonetischen Fragen investiert werden.

Zur Lösung dieses Problems existieren grundsätzlich zwei Ansätze:

- Verwendung eines äquivalenten Verfahrens, das ohne die phonetischen Fragen auskommt,
- Automatische Definition der phonetischen Fragen.

Ein Verfahren für das rein datengetriebene State-Tying (siehe Abschnitt 5.2.3.1), das eine Zuordnung von nicht gesehenen Triphonzuständen ohne die Verwendung von Entscheidungsbäumen mit phonetischen Fragen versucht, wird in [Aubert *et al.* 96] beschrieben. Dort werden anhand von Statistiken über die in den Clustern von Triphonzuständen vorkommenden Kontextphonemen entschieden, welche Mischverteilung für ein nicht gesehenes Triphon verwendet werden soll. Es wird der Cluster $\hat{C}$ als "passender" Cluster für einen nicht gesehenen Triphonzustand s^u gewählt (im folgenden soll der Exponent u für ungesehene (eng. *unobserved*), der Exponent o für gesehene (eng. *observed*) Ereignisse stehen), für den gilt:

$$\hat{C} = \underset{C}{\operatorname{argmax}}\, P(C|s^u)$$

Die Wahrscheinlichkeit $P(C|s^u)$ gibt an, mit welcher Wahrscheinlichkeit ein nicht gesehenes Triphon s^u einem Cluster C zugeordnet werden kann. Diese läßt sich umformen zu

$$\begin{aligned} P(C|s^u) &= \sum_{s \in S} P(C, s|s^u) && (7.1)\\ &= \sum_{s^o \in S^o} P(C, s^o|s^u) + \sum_{s^{'u} \in S^u} P(C, s^{'u}|s^u) && (7.2)\\ &= \sum_{s^o \in S^o} P(C|s^o, s^u) \cdot P(s^o|s^u) + \tilde{C}^u_{C,s^u} && (7.3) \end{aligned}$$

S^o ist dabei die Menge der im Training beobachteten Triphone, s^o ist ein Element dieser Menge, $\tilde{C}^u_{C,s^u}$ ist der Beitrag der ungesehenen Triphone S^u zu der Wahrscheinlichkeit

$P(C|s^u)$, der i.A. aufgrund des geringen Anteils an ungesehenen Triphonen (typischerweise 2-5%) vernachlässigt werden soll.
Nimmt man an, daß die Zugehörigkeit eines gesehenen Triphons s^o zu einem Cluster C deterministisch ist (d.h. $P(C|s^o, s^u) \in \{0, 1\}$), erhält man

$$\hat{C} = \operatorname*{argmax}_C \sum_{s \in C} P(s|s^u) \tag{7.4}$$

$$= \operatorname*{argmax}_C \sum_{s \in C} P(s, s^u) \tag{7.5}$$

Seien nun l und r der linke bzw. rechte Kontext eines Triphonzustandes s mit zentralem Phonem m und Segmentindex i und l' und r' der linke bzw. rechte Kontext eines Triphons s' mit zentralem Phonem m und Segmentindex i. Dann kann $P(s, s')$ approximiert werden durch

$$P(s, s') = P_{m,i}(r, l, r', l') \tag{7.6}$$

$$= P_{m,i}(r, r'|l, l') \cdot P_{m,i}(l, l') \tag{7.7}$$

$$= P_{m,i}(r, r') \cdot P_{m,i}(l, l') \tag{7.8}$$

$P_{m,i}(r, r')$ ist die Verbundwahrscheinlichkeit für die rechten Kontexte r und r', mit der für ein Zustandscluster für ein zentrales Phonem m und einen Segmentindex i die rechten Kontexte r und r' zusammen auftreten. $P_{m,i}(l, l')$ beschreibt dasselbe für die linken Kontexte l und l'. Diese Wahrscheinlichkeiten können über die beim Bottom-Up-Clustern gebildeten Zustandscluster geschätzt werden.
Damit ergibt sich für den optimalen Cluster $\hat{C}$ zu einem nicht gesehenen Triphonzustand s'

$$\hat{C} = \operatorname*{argmax}_C \sum_{s \in C} P_{m,i}(r, r^u) \cdot P_{m,i}(l, l^u)$$

Die mit dieser Methode erzielten Verbesserungen der Fehlerrate auf verschiedenen Korpora liegt zwischen 2% und 5% relativ verglichen mit einem Monophon-Backing-Off-System. Allerdings wurde in [Beyerlein *et al.* 97] nachgewiesen, daß dieses Verfahren für die Wortgrenzenmodellierung mit wortübergreifenden Triphonen Fehlerraten liefert, die ca. 6% schlechter sind im Vergleich zur Ergebnissen unter Verwendung von phonetischen Entscheidungsbäumen.
In dieser Arbeit wurde statt dessen der alternative Ansatz untersucht, die automatische Definition der phonetischen Fragen. Ein grundsätzlicher Vorteil dieses Verfahrens ist, daß die automatisch generierten phonetischen Fragen direkt für das State-Tying mit phonetischen Entscheidungsbäumen verwendet werden können, Änderungen an den Algorithmen zum State-Tying müssen also nicht vorgenommen werden. Fraglich ist allerdings, inwieweit es möglich ist, die z.Z. mit phonetischem Expertenwissen definierten Fragen durch einen Algorithmus erzeugen zu lassen.
Im folgenden soll nun das in dieser Arbeit entwickelte Verfahren zur Konstruktion von phonetischen Fragen vorgestellt werden.

7.3 Generierung der Fragen

Die Grundidee bei der automatischen Generierung von phonetischen Fragen besteht darin, zunächst eine Menge von Kandidatenfragen durch einen Algorithmus zu erzeugen, der die oben erwähnte Bedeutung der Phonemklassen in einem phonetischen Kontext berücksichtigt. Mit dieser Menge von Kandidatenfragen wird auf den Trainingsdaten ein phonetischer Entscheidungsbaum konstruiert. Während der Konstruktion dieses Baums werden Statistiken bzgl. der Verwendung der Fragen der Kandidatenmenge gesammelt. Diese Statistiken sind

- die Häufigkeit des Auftretens einer Frage im Baum und
- der mittlere Log-Likelihood-Gewinn pro Frage.

Aufgrund dieser Statistiken werden dann die n "besten" Fragen ausgewählt. Dazu werden zunächst alle Fragen, die weniger als zweimal verwendet worden sind, aus der Kandidatenmenge entfernt, um zu spezifische Fragen zu vermeiden. Danach werden die restlichen Fragen nach mittlerem Log-Likelihood-Gewinn sortiert, d.h. der Log-Likelihood-Gewinn wird für jede Frage durch die Anzahl der Verwendungen geteilt. Aus dieser Liste werden dann die ersten n Fragen ausgewählt und für das State-Tying mit phonetischen Entscheidungsbäumen verwendet.
Der grundsätzliche Ablauf des Verfahrens stellt sich also folgendenmaßen dar:

1. Erzeugung einer Menge von Kandidatenfragen,
2. Erzeugung eines phonetischen Entscheidungsbaums mit diesen Kandidatenfragen,
3. Reduktion der Zahl der Kandidatenfragen auf ca. 50-100 mit den Statistiken über den Generierungsprozeß des Entscheidungsbaums. Kriterien:
   - Häufigkeit des Auftretens einer Frage,
   - Mittlerer Log-Likelihood-Gewinn pro Frage.

7.3.1 Auswahlkriterium

Um zu vermeiden, daß Fragen, die lediglich für die im Training verwendete Menge von Triphonen eine gute Aufteilung der Triphonzustände bewirken, ausgewählt werden, wird zur Berechnung des Log-Likelihood-Gewinns bei der Konstruktion des Entscheidungsbaums ein modifiziertes Kriterium zum Aufteilen der Knoten verwendet. Dazu wird die Menge der Triphone in zwei disjunkte Mengen M_1 und M_2 unterteilt, wobei die Zugehörigkeit zu einer der beiden Mengen zufällig ist. Ähnlich wie bei dem im Kapitel 5 beschriebenen Verfahren zur Verwendung von zwei geschlechtsspezifischen Modellen pro Knoten des Entscheidungsbaums, werden pro Knoten für jede der beiden Triphonmengen eine Gaußverteilung verwendet. Der Gewinn an Log-Likelihood durch das Aufteilen eines Knotens wird nun als eine gewichtete Summe der Gewinne bzgl. der beiden Modelle in dem Knoten berechnet:

$$\hat{\delta} = (\delta_1 + \delta_2) \cdot \left(2 \cdot \frac{\sqrt{\delta_1 \delta_2}}{\delta_1 + \delta_2} \right)^2$$

δ_1 und δ_2 sind die Log-Likelihood-Gewinne bzgl. der beiden Triphonmengen M_1 und M_2, $\hat{\delta}$ der gewichtete Gewinn. Der Gewichtungsfaktor (rechter Term) hat die Eigenschaft, für $\delta_1 = \delta_2$ gleich 1 zu sein und für stark unterschiedliche δ auf 0 abzufallen.
Dadurch kann erreicht werden, daß der Algorithmus zur Konstruktion des Entscheidungsbaums die Fragen bevorzugt, die auf beiden (disjunkten) Triphonmengen eine ähnliche Verbesserung erzielen, während Fragen, die nur auf einer der beiden Mengen eine Verbesserung erzielt, unterdrückt werden.
Die Wirksamkeit des modifizierten Kriteriums wurde auf dem WSJ0-Trainingskorpus überprüft. Die Menge der Triphone des Korpus wurde dazu in *drei* disjunkte Teile aufgeteilt. Zwei der Teile wurden dazu verwendet, um Entscheidungsbäume mit und ohne das modifizierte Aufteilungskriterium zu konstruieren. Die dritte Menge wurde als Testmenge verwendet, um die konstruierten Entscheidungsbäume auf ungesehenen Daten zu evaluieren. Es ergab sich das in Tabelle 7.2 dokumentierte Bild.

Tabelle 7.2: Modifiziertes Aufteilungskriterium für Trainings- und Testdaten auf dem WSJ0-Trainingskorpus.

Kriterium	Δ LL (Training)	Δ LL (Test)
Standard	4.21	3.50
Modifiziert	4.05	4.09

Auf den Trainingsdaten war der mittlerem Log-Likelihood-Gewinn des Baums, der mit dem Standardkriterium erzeugt worden ist, um 0.16 absolut besser. Auf den Testdaten allerdings war der Entscheidungsbaum, für den das modifizierte Splitkriterium verwendet worden war, deutlich besser, es ergab sich ein Unterschied von 0.62. Weiterhin fällt auf, daß der Baum, der durch das modifizierte Kriterium erzeugt worden ist, für Trainingsdaten wie für Testdaten ungefähr denselben Log-Likelihood-Gewinn erzielt, so daß vermutet werden kann, daß dieser Baum gesehene wie ungesehene Triphone gleich gut klassifiziert. Allerdings haben Erkennungstests auf einem Teilkorpus von WSJ-5k gezeigt, daß die Fehlerrate dieses Baums um ca 10% schlechter ist verglichen mit dem durch das Standardkriterium erzeugten Baum. Hier zeigt sich, daß der Gewinn, der durch das modifizierte Aufteilungskriterium für ungesehene Triphone erzielt wurde, nicht ausreicht, um den Verlust an Modellierungsgenauigkeit für gesehene Triphone zu kompensieren, da den Anteil der ungesehenen Triphone an der Gesamtheit der Triphone des Testkorpus von WSJ-5k relativ gering ist (ca. 4%). Das Verfahren sollte aber bei Anwendungen von Entscheidungsbäumen, bei denen die Testdatenmenge nicht oder nur zu einem geringen Teil in der Trainingsdatenmenge enthalten ist, zu besseren Resultaten führen.
Für die automatische Fragengenerierung soll durch die Verwendung dieses Verfahrens zur Berechnung von Log-Likelihood-Gewinnen für die Fragen der Kandidatenmenge verhindert werden, daß Fragen, die lediglich auf der Triphonmenge der Trainingsdaten zu Aufteilungen mit hohen Log-Likelihood-Gewinnen führen, überbewertet und daher bevorzugt ausgewählt werden.
Tests auf verschiedenen Spracherkennungskorpora mit einem geringen Anteil an ungesehenen Triphonen haben gezeigt, daß diese Vorgehensweise zu mindestens gleichen Fehlerraten wie bei der Verwendung von Expertenfragen bzw. automatisch generierten Fragen, die nicht mit dem modifizierten Splitkriterium ausgewählt wurden, führt. Wegen der größeren

Robustheit kann man hoffen, daß bei einem signifikanten Anteil an ungesehenen Triphonen in einem Testkorpus die mit dem modifizierten Splitkriterium ausgewählten Fragen zu besseren Fehlerraten führen.
Zu klären bleibt noch, wie die Kandidatenmenge der phonetischen Fragen erzeugt werden kann. Dazu wurden drei Verfahren mit steigender Komplexität getestet:

- Zufallsbasierte Erzeugung

  Die Phonemmengen werden zufällig erzeugt.

- Monophonbasierte Erzeugung

  Die Phonemmengen werden durch Bottom-Up-Clustern von Monophonzuständen mit gleichem Segmentindex erzeugt.

- Diphonbasierte Erzeugung

  Die Phonemmengen werden durch Bottom-Up-Clustern von Diphonzuständen mit gleichem Phonem und gleichem Segmentindex erzeugt.

Diese Verfahren werden in den nächsten drei Unterabschnitten genauer erläutert.

7.3.2 Zufallsbasierte Generierung

Die zufallsbasierte Erzeugung der Kandidatenfragenmenge ist die naheliegendste und auch einfachste Methode. Um eine Kandidatenfrage M zu erzeugen, wird zunächst die *Kardinalität*, d.h. die Anzahl der Phoneme in der Kandidatenfrage, durch Zufall bestimmt. Diese Kardinalität l wird gleichverteilt aus dem Intervall $[1, 2, ..., N/2]$ gewählt, wobei N die Gesamtzahl der Phoneme im Phoneminventar ist. Durch die Beschränkung der Kardinalität auf $N/2$ werden Fragen mit einer hohen Kardinalität vermieden, die in einer phonetischen Kategorisierung i.A. ebenfalls nicht vorkommen. Diese Fragen hoher Kardinalität werden bei dieser Methode durch die Komplementäreigenschaft der phonetischen Fragen (die Verneinung einer phonetischen Frage M ist gleich der Bejahung einer phonetischen Frage, die alle Phoneme enthält, die *nicht* in M enthalten sind) indirekt modelliert.
Nach der Bestimmung der Kardinalität l wird in einem zweiten Schritt bestimmt, welche Phoneme in der Frage enthalten sind. Dazu wird aus dem Phoneminventar l-mal ein Phonem ohne Zurücklegen gezogen, d.h. jedes Phonem ist gleich wahrscheinlich in der Kandidatenfrage enthalten. Die Zusammensetzung der Kandidatenfrage aus Phonemen ist somit hypergeometrisch verteilt.
Dieser zweigeteilte Aufbau wurde gewählt, weil er durch die zufallsbasierte Bestimmung der Kardinalität im ersten Schritt eine gleichmäßige Längenverteilung der generierten Fragen garantiert. Würde man lediglich die Zugehörigkeit eines Phonems zu einer Kandidatenfrage gleichverteilt modellieren (d.h. ein Phonem p ist mit einer Wahrscheinlichkeit x in der Kandidatenfrage und mit einer Wahrscheinlichkeit $1-x$ nicht in der Kandidatenfrage enthalten), würden Fragen generiert, die abhängig vom Parameter x eine Verteilung der Kardinalität zeigen, die bestimmte Kardinalitäten bevorzugt, während andere Kardinalitäten praktisch nicht vorkommen. Z.B. würden bei sehr kleinem x fast nur Fragen der Kardinalität 1 erzeugt, bei sehr großem x würden die Fragen fast immer die Kardinalität des Phoneminventars haben. Dies wird durch die zufallsbasierte Bestimmung der Kardinalität der Kandidatenfrage im ersten Schritt vermieden.

Ein Beispiel für eine mit dieser Methode erzeugte Menge von phonetischen Fragen (d.h. die nach der Reduktion der Kandidatenfragenmenge verbleibenden Fragen) ist

```
QUEST0  r ul um un
QUEST1  b d j
QUEST4  awh oo s
QUEST5  aa k
QUEST6  UH n v
QUEST7  awh d un w z zh
QUEST8  ee um
...
```

Die auf diese Weise erzeugten Fragen scheinen mit den durch Experten definierten Fragen (siehe oben, [Hon 92]) nicht viel gemeinsam zu haben. Dies wird von uns auch nicht gefordert, wie sich aber im Ergebniskapitel zeigen wird, sind diese Fragen bzgl. Log-Likelihood-Gewinn und Fehlerrate sowohl den durch Experten definierten Fragen als auch den durch die beiden folgenden, komplexeren automatischen Verfahren erzeugten Fragenmengen unterlegen. Dies stößt uns auf ein Dilemma, daß einer solchen relativ simplen Methode anhängt. Erzeugt man eine relativ kleine Menge von Kandidatenfragen, ist die Wahrscheinlichkeit gering, daß sich in dieser Menge genug "gute" Kandidaten befinden. Erzeugt man dagegen sehr viele Kandidaten, dann sind die in der anschließenden Auswahl der Fragen erzeugten Statistiken (Anzahl der Verwendungen und mittlerer Log-Likelihood-Gewinn) nicht genau genug, da insgesamt bei einer Baumkonstruktion nur ca. 2000-3000 Aufteilungen auftreten. Aufgrund dieser Unzulänglichkeit wurde dieses Verfahren nicht weiter betrachtet. Statt dessen führte die Verwendung von *datengetriebenen* Methoden zur Generierung der Kandidatenfragenmenge zu wesentlich besseren Ergebnissen.

7.3.3 Fragengenerierung durch Bottom-Up-Clustern von HMM-Zuständen

Wie bereits oben erwähnt, sind die durch Experten definierten phonetischen Fragen im wesentlichen Phonemklassen, die aufgrund der Art der Erzeugung der Phoneme durch den menschlichen Artikulationsapparat zusammengefaßt werden. Das bedeutet aber auch, daß sich die Phoneme einer Phonemklasse bzw. einer phonetischen Frage akustisch mehr oder weniger ähneln (siehe Abbildung 7.1).
Diese Überlegung führt nun zu den datengetriebenen Verfahren zur Fragengenerierung. Diese verwenden zur Erzeugung der Kandidatenfragenmenge die akustischen Beobachtungen des Trainingskorpus, um auf der Basis von akustischen Ähnlichkeiten Phonemcluster zu bilden. Aufgrund der Elemente, die diese Cluster enthalten, werden dann die Kandidatenfragen gebildet. Die Verfahren laufen dabei grundsätzlich folgendermaßen ab:

1. Definiere eine Menge von zu clusternden (kontextabhängigen) Phonemmodellen.
2. Schätze anhand des Trainingskorpus zu diesen Phonemmodellen HMM-Zustände mit Einzelverteilungen.
3. Clustere diese HMM-Zustände aufgrund eines Abstandsmaßes mit dem in Kapitel 5 beschriebenen Bottom-Up-Clusterverfahren zusammen.

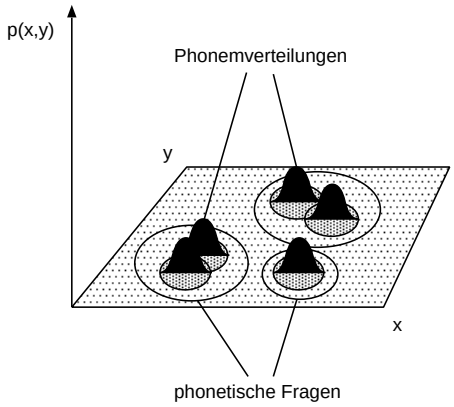


Abbildung 7.1: Phonetische Klassen im akustischen Raum

4. Verwende die beim Clustervorgang aufgetreteten Zustandsmengen, um die phonetischen Fragen zu definieren.

Offensichtlich ähnelt das Verfahren den Methoden, die in Kapitel 5 zum State-Tying verwendet wurden. In beiden Fällen werden Gruppen von HMM-Zuständen gebildet, deren Emissionsverteilungen ähnlich bezüglich eines Abstandsmaßes sind. Daher kann auch für die automatische Fragengenerierung das im Abschnitt 5.2.3.1 beschriebene Verfahren zum Clustern von Triphonzuständen verwendet werden. Als Abstandsmaß für die akustische Ähnlichkeit wurde das in Abschnitt 5.2.3 beschriebene Log-Likelihood-Kriterium benutzt. Im weiteren werden nun die beiden implementierten Verfahren zur datengetriebenen Erzeugung der Kandidatenfragenmenge beschrieben. Diese Verfahren unterscheiden sich im wesentlichen durch die Modelle, die mit dem oben beschriebenen Verfahren geclustert werden. Das erste Verfahren verwendet Monophonzustände, die nach Segmentindex getrennt geclustert werden, das zweite Verfahren Diphonzustände, die nach Phonem und Segmentindex getrennt geclustert werden. Beide Verfahren erzielen vergleichbare Fehlerraten auf den getesteten Spracherkennungskorpora, wobei das diphonbasierte Verfahren durchgängig minimal geringere Fehlerraten aufweist.

7.3.3.1 Monophonbasierte Generierung

Die monophonbasierte Fragengenerierung basiert auf der Annahme, daß akustisch ähnliche Phoneme, die in einem phonetischen Kontext stehen, auch ähnliche Einflüsse auf die akustische Realisierung des kontextabhängigen Phonems haben. Dies ist natürlich nur dann korrekt, wenn aus der durch das Abstandsmaß definierten akustischen Ähnlichkeit auch eine ähnliche Positionierung der Artikulatoren folgert, die letzlich für die Koartikulationseffekte verantwortlich sind. Diese geforderte Eigenschaft soll hier nicht weiter untersucht werden, da eine Untersuchung der Korrelation zwischen akustischer und artikulatorischer Ähnlichkeit über die Zielsetzung dieser Arbeit hinausgehen würde. Die Validierung der Behauptung soll statt dessen anhand der durch dieses Verfahren erzielten Ergebnisse auf den Spracherkennungskorpora durchgeführt werden.
Die monophonbasierte Fragengenerierung arbeitet nach folgendem Schema:

1. Schätze Monophon-HMM mit dem vorhandenen Trainingskorpus.

2. Unterteile die Menge der HMM-Zustände nach Segmentnummer.
3. Clustere die Zustände mit gleicher Segmentnummer zusammen.
4. Verwende die während des Clustervorgangs entstandenen Zustandsmengen zur Definition der Kandidatenmenge.

Die Fragen werden dabei aus den Zustandsmengen ermittelt, indem die Phonemlabel der Zustände in einem Cluster als Phoneme der phonetischen Frage verwendet werden. Die Unterteilung der HMM-Zustände nach Segmentnummer geschieht zum einem, um eine möglichst gute Vergleichbarkeit der Emissionsverteilungen der Zustände zu erreichen. Zum anderen kann so die Menge der durch das Verfahren produzierten Fragen um den Faktor 3 gesteigert werden. Trotzdem erreicht man mit der monophonbasierten Fragengenerierung bei n Ausgangsphonemen nur ca. $2n$ Kandidatenfragen, so daß eine anschließende Selektion der k besten Fragen für k im Bereich der typischen Anzahl von expertendefinierten phonetischen Fragen (ca. 40-50) bei einer angenommen Zahl von 43 Phonemen nicht mehr sinnvoll erscheint. Statt dessen werden bei den Tests, die zu dieser Methode durchgeführt wurden, die erzeugte Kandidatenfragenmenge direkt zur Erzeugung des phonetischen Entscheidungsbaums verwendet.
Abbildung 7.2 zeigt anhand eines Beispiels den Ablauf des Algorithmus für die fünf Vokale *ah, eh, ih, oh, uh.*

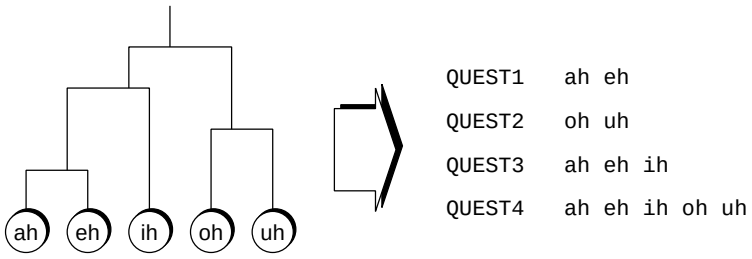


Abbildung 7.2: Monophonbasierte Fragengenerierung

Im ersten Clusterschritt werden die Zustände der Phoneme *ah* und *eh* zusammengefaßt. In Schritt 2 werden *oh* und *uh* geclustert. In Schritt 3 wird der Cluster *ah,eh* mit dem Zustand *ih* vereinigt. Im letzten Schritt werden dann die Cluster *ah,eh,ih* und *oh,uh* zusammengefaßt. Die durch diese Clusterschritte definierten phonetischen Fragen sind im rechten Teil der Abbildung zu finden.
Ein Beispiel für die mit dieser Methode generierten Fragen für den WSJ-5k-Korpus:

```
QUEST1   l   ul
QUEST2   k   p
QUEST3   j   zh
QUEST4   ih  uh
QUEST5   aw  ow
QUEST6   ae  ah
QUEST7   ae  eh
QUEST8   ae  oo
```

```
QUEST9   ah  awh
QUEST10  f   th
QUEST11  ey  ih
QUEST12  ch  sh
QUEST13  ee  y
QUEST14  d   g

...

QUEST32 f h k p t th
QUEST33 b d dh f g h k p t th v
QUEST34 ee ey ih ooh uh y
QUEST35 UH aa ae aw eh ow
QUEST36 b ch d dh f g h j k p sh t th v zh
QUEST37 ee ey ih m n ng ooh uh um un y
QUEST38 UH aa ae ah aw awh eh l oh ow ul w
QUEST39 UH aa ae ah aw awh eh l oh oo ow r ul ur w
QUEST40 UH aa ae ah aw awh ee eh ey ih l m n ng oh oo ooh ow r
uh ul um un ur w y
QUEST41 b ch d dh f g h j k p s sh t th v z zh
```

Erfreulicherweise sind die meisten kurzen Fragen bzgl. der akustischen und der artikulatorischen Ähnlichkeit relativ plausibel. Z.B. enthält Frage 2 zwei stimmlose Plosive und Frage 10 zwei Frikative. Für die längeren Fragen ist dies natürlich nicht unbedingt der Fall, allerdings sind beispielsweise in Frage 33 vor allen Plosive und Frikative zu finden. Insgesamt werden für den WSJ-5k-Korpus 94 Fragen durch den monophonbasierten Algorithmus erzeugt.
Obwohl die durch dieses Verfahren generierten Fragen, vergleicht man diese mit den durch einen Experten definierten Fragen, relativ plausibel sind, können doch zumindest zwei Kritikpunkte an dieser Methode festgestellt werden:

- Durch diese Methode können, wie schon weiter oben erwähnt, bei einem Phoneminventar von n Phonemen nur ca. $2n$ phonetische Fragen generiert werden, so daß eine anschließende Reduktion der Fragenmenge nicht mehr sinnvoll ist.

- Die Methode gruppiert die Phoneme nach ihrer akustischen Ähnlichkeit. Was aber bei der Erzeugung der phonetischen Entscheidungsbäume wichtiger ist, ist der Einfluß eines Phonems auf die akustische Realisierung eines Nachbarphonems. Dies wird aber durch die monophonbasierte Fragengenerierung nicht berücksichtigt.

Eine Methode, die diese Kritikpunkte vermeidet, soll nun im nächsten Abschnitt dargestellt werden. Ob diese tatsächlich relevant für die auf den Testkorpora erzielten Fehlerraten sind, wird im Ergebnisteil geklärt werden.

7.3.3.2 Diphonbasierte Generierung

Bei der diphonbasierten Fragengenerierung verwendet man statt der Monophonzustände des vorherigen Verfahrens Diphonzustände. D.h. für die Beobachtungen des Trainingskorpus werden Diphon-HMM geschätzt, deren Emissionsverteilungen aufgrund der Kontextabhängigkeit links oder rechts den Einfluß des jeweiligen Kontextphonems auf die akustische Realisierung des kontextabhängigen Phonems enthalten sollten.
Die diphonbasierte Fragengenerierung arbeitet nach folgendem Schema:

1. Schätze Diphon-HMM-Zustände für die Beobachtungen des Trainingskorpus.
2. Unterteile die Menge der HMM-Zustände nach kontextabhängigem Phonem und Art des Kontextes (links/rechts).
3. Clustere die Zustände mit gleichem kontextabhängigen Phonem und gleicher Kontextart zusammen.
4. Verwende die während des Clustervorgangs entstandenen Zustandsmengen zur Definition der Kandidatenmenge.

Die Ermittelung der in einer Kandidatenfrage enthaltenen Phoneme geschieht hier nicht über das Phonem selbst, da dieses innerhalb eines Clusters aufgrund der Aufteilung der HMM-Zustände gleich ist. Statt dessen wird hier das Kontextphonem verwendet.
Abbildung 7.3 zeigt anhand eines Beispiels den Ablauf des Algorithmus für das Phonem *p* als kontextabhängiges Phonem und die fünf Vokale *aa, ah, eh, ih, oh* als linke Kontextphoneme.

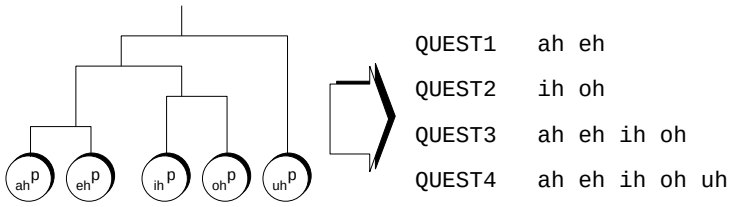


Abbildung 7.3: Diphonbasierte Fragengenerierung

Im ersten Clusterschritt werden die Zustände der Phoneme mit Kontext *ah* und *eh* zusammengefaßt. In Schritt 2 werden die Kontexte *ih* und *oh* geclustert. In Schritt 3 wird der Cluster *ah,eh* mit dem Cluster *ih,oh* vereinigt. Im letzten Schritt werden dann die Cluster *ah,eh,ih,oh* und der Zustand mit *uh* zusammengefaßt. Die durch diese Clusterschritte definierten phonetischen Fragen sind ebenfalls im rechten Teil der Abbildung zu finden.
Offensichtlich werden durch diese Methode wesentlich mehr Kandidatenfragen generiert, da hier statt drei Clustervorgängen $2n$ Clustervorgänge durchgeführt werden. Für den oben erwähnten WSJ-5k-Korpus werden durch das Verfahren ca. 1500 Kandidatenfragen erzeugt. Diese werden dann durch das in Abschnitt 7.3 beschriebene Verfahren auf eine Zahl von 50-100 Fragen reduziert. Außerdem berücksichtigt das Verfahren implizit die Ähnlichkeit der Beeinflussung durch die jeweiligen Mengen von Kontextphonemen, da

bei den Clustervorgängen bevorzugt die HMM-Zustände zusammengefaßt werden, deren Emissionsverteilungen und damit akustische Realisierung ähnlich sind.
Ein Beispiel für die mit dieser Methode generierten Fragen für den WSJ-5k-Korpus:

```
QUEST1   aw  un
QUEST2   b   w
QUEST3   s   z
QUEST4   k   p
QUEST5   m   w
QUEST6   UH  aa
QUEST7   aa  awh
QUEST8   aa  eh
QUEST9   ee  ey  y
QUEST10  ch  j   sh
QUEST11  l   ul  w
QUEST12  s   th  z
QUEST13  s   sh  z
QUEST14  oh  r   ur

...
```

Wie bei der monophonbasierten Fragengenerierung finden sich auch hier viele phonetische Fragen, die auch im Hinblick auf eine expertenbasierte Kategorisierung der Phoneme plausibel erscheinen. Beispiele sind *s,z* (stimmloser, stimmhafter Frikativ) oder *k,p* (Plosiv). Andere Fragen, wie z.B. *l,ul,w* sind aufgrund ihrer phonetischen Klassifizierung eher als heterogen einzustufen. Letztlich entscheidet aber über die Plausibilität der hier gezeigten Ansätze zur automatischen Fragengenerierung weniger die Stringenz der phonetischen Fragen im Kontext der Phonetik, sondern in erster Linie die Qualität der mit diesen Fragen erzeugten phonetischen Entscheidungsbäume. Diese soll im nächsten Kapitel anhand von erzieltem Log-Likelihood-Gewinn bei der Erzeugung der Bäume und der auf den Testkorpora erreichten Fehlerraten nachgewiesen werden.

7.4 Ergebnisse

In diesem Abschnitt sollen die für die verschiedenen Methoden erzielten Ergebnisse auf dem

- WSJ-5k-Korpus und dem
- VM-5k-Korpus

dargestellt werden. Dabei werden sowohl die mit den Methoden erzielten Modellierungsverbesserungen mit Hilfe des Log-Likelihood-Gewinns für den phonetischen Entscheidungsbaum als auch die auf den oben genannten Testkorpora erzielten Fehlerraten als

Maßstab verwendet. Als Merkmale wurden ebenfalls die in in Kapitel 1 erwähnten cepstralen Merkmale (33 Komponenten) verwendet. Weitere Randbedingungen bei den folgenden Tests sind:

- Laplace-Verteilungen mit gepooltem Varianzvektor,
- State-Tying mit phonetischem Entscheidungsbaum (2000 Blätter),
- Bigramm-Sprachmodell mit
  - $PP = 107$ für WSJ-5k und
  - $PP = 66$ für VM-5k.

Diese Randbedingungen gelten für alle in diesem Abschnitt erwähnte Tests, falls keine anderen Angaben gemacht werden.

7.4.1 Log-Likelihood-Gewinn

Als Maß für die durch eine Modifikation der Entscheidungsbaum-Methode erzielte Verbesserung der akustischen Modellierung durch den Entscheidungsbaum kann die Verbesserung des Log-Likelihood-Gewinns für die Trainingsdatenmenge bei der Konstruktion des Baums dienen. Hierbei wird die Differenz zwischen der Log-Likelihood der Trainingdaten, die sich vor bzw. nach dem Aufteilen der Knoten nach dem phonetischen Kontext ergibt, verwendet. Diese wird auf die Zahl der akustischen Beobachtungen der Trainingsdaten normiert. D.h. die Differenz ist hier geringer als in Kapitel 5, da dort die gesamte Log-Likelihood- Differenz zwischen Wurzel (ein Blatt) und vollem Baum (2000 Blätter) angegeben ist, während hier nur die Differenz zwischen dem Baum nach der Aufteilung nach zentralem Phonem und Segmentindex (ca. 130 Blätter) und dem vollen Baum (2000 Blätter) angegeben ist. Die so ermittelte mittlere Log-Likelihood-Verbesserung für die getesteten Verfahren und rein wortinterne Triphnmodelle findet sich in Tabelle 7.3.

Tabelle 7.3: Mittlere Log-Likelihood-Verbesserung für automatische Fragengenerierung auf WSJ-5k und VM-5k.

Verfahren	mittlerer LL-Gewinn WSJ-5k	mittlerer LL-Gewinn VM-5k
Experte	4.20	3.12
Zufallsbasiert	4.19	–
Monophonbasiert	4.25	3.33
Diphonbasiert (Standard)	4.27	3.33
Diphonbasiert (Modifiziert)	4.26	3.33

Zeile 1 enthält das Ergebnis für die in Kapitel 5 verwendeten manuell generierten phonetischen Fragen nach [Hon 92]. Die Zeile 2-5 zeigen die Ergebnisse für die verschiedenen automatischen Methoden, wobei “Standard” das Standard-Kriterium zum Aufteilen der Knoten und “Modifiziert” das in diesem Kapitel beschriebene modifizierte Kriterium referenziert.

Die Tabelle zeigt, daß die Fragen, die mit dem einfachen zufallsbasierten Verfahren generiert wurden, zu ähnlichen Log-Likelihood-Verbesserungen führen wie die Expertendefinierten Fragen. Die aufwendigeren datengetriebenen Verfahren führen dagegen zu einer Verbesserung des Log-Likelihood-Gewinns gegenüber den Expertenfragen (die Ergebnisse auf VM-5k für die verschiedenen automatischen Methoden sind nur aufgrund der Rundung auf zwei Nachkommastellen identisch). Allerdings korrellieren solche Verbesserungen der Log-Likelihood auf Trainingsdaten i.A. nur bedingt mit einer Verbesserung der Fehlerrate auf Testdaten, wie sich im folgenden Kapitel zeigen wird. Dort spielt neben einer möglichst exakten Modellierung der Trainingdaten auch die Verallgemeinerungsfähigkeit des erzeugten Modells eine Rolle, so daß dort erwartet werden kann, daß die aufgrund von allgemeinem phonetischen Wissen definierten Expertenfragen gegenüber den automatisch generierten Fragen bevorteilt sein werden.

7.4.2 Fehlerraten

In diesem Abschnitt werden die für die verschiedenen Methoden ermittelten Fehlerraten auf den Testkorpora dargestellt und mit den Ergebnissen für die durch Experten definierten Fragen verglichen. Die dafür verwendeten Entscheidungsbäume entsprechen dabei den im vorherigen Abschnitt getesteten Bäumen.
Die Tabellen 7.4 und 7.5 zeigen die erzielten Fehlerraten auf WSJ-5k und VM-5k für die drei beschriebenen Methoden (zufallsbasiert, monophonbasiert und diphonbasiert), verglichen mit der Fehlerrate für die expertendefinierten Fragen. Dabei wurden nur wortinterne Triphone verwendet, die Zahl der nicht gesehenen Triphone ist also relativ gering.

Tabelle 7.4: Wortfehlerrate [%] für automatische Fragengenerierung auf WSJ-5k.

Verfahren	Anzahl Fragen	LL-Gewinn	DEL-INS [%]	WER [%]
Experte	88	4.20	1.3 - 0.7	7.1
Zufallsbasiert	144	4.19	1.4 - 0.8	7.5
Monophonbasiert	137	4.25	1.4 - 0.6	7.1
Diphonbasiert (Standard)	144	4.27	1.4 - 0.6	7.2
Diphonbasiert (Modifiziert)	144	4.26	1.4 - 0.6	7.0

Tabelle 7.5: Wortfehlerrate [%] für automatische Fragengenerierung auf VM-5k.

Verfahren	Anzahl Fragen	LL-Gewinn	DEL-INS [%]	WER [%]
Experte	78	3.12	4.4 - 3.7	21.9
Monophonbasiert	153	3.33	4.6 - 3.5	22.0
Diphonbasiert (Standard)	145	3.33	4.5 - 3.4	21.2
Diphonbasiert (Modifiziert)	145	3.33	4.9 - 3.8	22.2

Die durch Bottom-Up-Clustern generierten Fragen führen auf beiden Testkorpora zu vergleichbaren Fehlerraten, während die zufallsgenerierten Fragen (nur für WSJ-5k getestet) ca. 5% relativ schlechter abschneiden. Dies zeigt vor allem, daß die Generierung von plausiblen Kandidatenfragen bei dem Verfahren zur Erzielung von guten Fehlerraten wichtig ist, während das Auswahlverfahren hier keine so große Rolle zu spielen scheint. Allerdings zeigt sich bei der diphonbasierten Methode, daß bei Verwendung des modifizierten Aufteilungskriteriums bei der Auswahl der Fragen die Fehlerrate gegenüber dem Standardkriterium noch einmal leicht sinkt, von 7.2% auf 7.0%. Dies zeigt, daß die durch das modifizierte Kriterium ausgewählten Fragen zwar die Trainingsdaten weniger gut modellieren (Log-Likelihood-Verbesserung), dafür aber die nicht gesehenen Triphonen besser beschreiben.

Auffällig ist auch die signifikante Verbesserung der Fehlerrate auf VM-5k durch die diphonbasierten Fragen mit dem Standard-Splitkriterium. Grund dafür könnte die spezielle Eigenschaft dieses Verbmobil-Korpus sein, daß die beim Test vorkommende Triphonmenge vollständig durch das Training abgedeckt wird. D.h. durch eine bzgl. des Trainings optimale Aufteilung der Triphone bei der Erzeugung des phonetischen Entscheidungsbaums sollte man die besten Fehlerraten erwarten. Daher spielt bei VM-5k die Verallgemeinerungsfähigkeit der phonetischen Fragen keine Rolle, statt dessen sollte eine Auswahl der Fragen, die diese optimale Aufteilung der Triphone ermöglicht, die geringsten Fehlerraten ermöglichen.

Um den Einfluß der Anzahl der durch das Auswahlverfahren gewählten Fragen zu bestimmen, wurde für die diphonbasierte Fragengenerierung eine zweite Versuchsreihe mit Fragenmengen verschiedener Kardinalität durchgeführt.

Tabelle 7.6: Wortfehlerrate [%] für diphonbasierte Fragengenerierung (modifiziertes Aufteilungskriterium) und verschieden große Fragenmengen auf WSJ-5k.

Anzahl Fragen	LL-Gewinn	DEL-INS [%]	WER [%]
94	4.25	1.4 - 0.6	7.1
144	4.25	1.4 - 0.6	7.0
1548	4.30	1.2 - 0.7	7.0

Tabelle 7.6 zeigt, daß die Menge der verwendeten Fragen zwar einen Einfluß auf den Log-Likelihood-Gewinn bei der Erzeugung des Entscheidungsbaums hat, nicht aber auf die Fehlerrate. Diese ist in allen drei Tests praktisch identisch.

Eine weitere Beobachtung ist, daß trotz des höhreren Log-Likelihood-Gewinns durch die automatisch generierten Fragen (monophon- und diphonbasiert) die Fehlerrate nicht besser ist im Vergleich zu den expertendefinierten Fragen. Bei den zufallsbasiert erzeugten Fragen ist sogar trotz vergleichbarem Log-Likelihood-Gewinn die Fehlerrate deutlich schlechter als bei den Expertenfragen. Dies könnte vermuten lassen, daß die automatisch generierten Fragen für Tests mit einem geringen Anteil an nicht gesehenen Triphonen relativ gut funktioniert, bei einem höheren Anteil, wie er z.B. bei der Wortgrenzenmodellierung mit wortübergreifenden Triphonen auftritt, schlechter funktionieren könnte.

Dazu wurde eine weitere Versuchsreihe mit wortübergreifenden Triphonen und der diphonbasierten Fragengenerierung durchgeführt. Die verwendeten Entscheidungsbäume wurden mit wortübergreifenden Triphonen neu berechnet (siehe Tabellen 7.7 und 7.8). Für die Erkennung wurde dabei die *n-best*-Suche verwendet, wobei die Länge der Zwischenwortpause durch die dynamische Pauseschätzung ($N_{sil} = 1$) ermittelt wurde und keine Interpolation verwendet wurde.

Tabelle 7.7: Wortfehlerrate [%] für diphonbasierte Fragengenerierung (modifiziertes Aufteilungskriterium) und verschiedene Wortgrenzenmodellierungen auf WSJ-5k ($n = 20$).

Wortgrenzen-modellierung	Fragendefinition	Anzahl Fragen	DEL-INS [%]	WER [%]
wortintern	Experte	88	1.3 - 0.7	7.1
wortübergreifend	Experte	88	1.0 - 0.8	6.3
wortübergreifend	diphonbasiert	94	1.0 - 0.7	6.5
wortübergreifend	diphonbasiert	144	1.0 - 0.6	6.3

Tabelle 7.8: Wortfehlerrate [%] für diphonbasierte Fragengenerierung und verschiedene Wortgrenzenmodellierungen auf VM-5k ($n = 100$).

Wortgrenzen-modellierung	Fragendefinition	Anzahl Fragen	DEL-INS [%]	WER [%]
wortintern	Experte	77	4.4 - 3.7	21.9
wortübergreifend	Experte	77	3.9 - 3.9	20.5
wortübergreifend	diphonbasiert (mod.)	98	3.7 - 3.9	20.4
wortübergreifend	diphonbasiert (mod.)	124	3.5 - 3.9	20.3
wortübergreifend	diphonbasiert (Std.)	134	3.6 - 4.1	20.2

Auch bei der Verwendung von wortübergreifenden Triphonen liefern die automatisch generierten Fragen ähnlich gute Resultate wie die expertendefinierten Fragen. Dies zeigt, daß auch die mit diesen Fragen generierten Bäume ausreichend gute Generalisierungseigenschaften aufweisen, obwohl die automatisch generierten Fragen, im Gegensatz zu den expertendefinierten Fragen, stärker an den Trainingskorpus adaptiert sind.

7.5 Vergleich mit anderen Verfahren

Wie schon in Abschnitt 2.3.1 gesagt, existiert nur ein weiteres Verfahren, das auf ähnliche Weise automatisch Fragen generiert [McAllaster *et al.* 97]. Bei diesem Verfahren werden ebenfalls *Phoneme* mit einem Bottom-Up-Verfahren geclustert, deren Abstand über das Abstandsmaß

$$d(l_i, l_j) = \sum_{mr} \frac{||\vec{\mu}_{l_i mr} - \vec{\mu}_{l_j mr}||^2}{n_{l_i mr} + n_{l_j mr}}$$

definiert ist. l_i und l_j sind linke Kontexte, r rechte Kontexte zum Phonem m. Eine symmetrische Formel kann für das Clustern von rechten Kontexten angegeben werden. Mit diesem Abstandsmaß werden die Kontextphoneme sukzessive zusammengeclustert, die dabei entstehenden Cluster werden dann als phonetische Fragen verwendet. D.h. das Verfahren ist eine Art Mischung aus dem monophonbasierten Verfahren (es werden also nur relativ wenige Fragen generiert) und dem diphonbasierten Verfahren (die Relevanz von Phonemen im Kontext wird berücksichtigt). Die Fehlerraten, die die Autoren berichten, sind praktisch identisch mit denen der durch Experten definierten Fragen, so daß die in dieser Arbeit gefundenen Ergebnisse dadurch bestätigt werden.

7.6 Zusammenfassung

Abschließend kann gesagt werden, daß es mit den hier vorgestellten Methoden zur automatischen Fragengenerierung möglich ist, gute und relativ robuste Mengen von Fragen für das State-Tying mit phonetischen Entscheidungsbäumen zu erzeugen, die zu Fehlerraten führen, welche vergleichbar sind mit den Ergebnissen, die unter Verwendung von mit phonetischem Expertenwissen definierten Fragen erreichbar sind. Dies ist konsistent mit der Aussage, die D. Allaster in seinem Vortrag trifft (siehe Abschnitt 2.3.1).
Allerdings können die expertendefinierten Fragen immer nur so gut sein wie derjenige, der diese definiert. Wir wollen also nicht ausschließen, daß ein anderer Experte eine Menge von Fragen für diesen Testkorpus definieren kann, der zu noch besseren Fehlerraten führt. Andererseits kann festgehalten werden, daß die automatische Fragengenerierung insbesondere für Korpora mit einem geringen Anteil an nicht gesehenen Triphonen sehr gute Ergebnisse erzielt, wenn diese spezielle Eigenschaft berücksichtigt wird. Weiterhin kann durch die Verwendung dieses Verfahrens weitgehend sichergestellt werden, daß die für das State-Tying verwendeten Fragen zu Fehlerraten führen, die relativ nahe am Optimum liegen. Dadurch kann vermieden werden, daß zur Optimierung einer Mengen von Fragen extensive Tests durchgeführt werden müssen.

Kapitel 8

Ausblick

In diesem Kapitel sollen zu den in dieser Arbeit betrachteten Verfahren einige Erweiterungen vorgeschlagen werden, die hier nicht mehr berücksichtigt werden konnten. Insbesondere bei der einphasigen Suche für wortübergreifende Triphone kann durch die Implementierung von Rekombinations- und Pruningverfahren deutlich verbessert werden, da in der jetzigen Realisierung derartige Techniken nicht verwendet werden und dadurch der Suchraum im Vergleich zur Suche mit rein wortinternen Triphonen ca. 5-10 mal größer ist.

8.1 State-Tying

Beim State-Tying mit phonetischen Entscheidungsbäumen werden die Beobachtungen an den Baumknoten mit Gauß-Einzelverteilungen modelliert. Dies steht in Diskrepanz zu Training und Erkennung, bei denen Gaußsche Mischverteilungen verwendet werden. Eine naheliegende Verbesserung würde also daraus bestehen, auch an den Knoten des phonetischen Entscheidungsbaums Mischverteilungen zur Modellierung der Beobachtungen zu verwenden. Hierzu sollen im folgenden zwei Verfahren vorgeschlagen werden.

8.1.1 Mischverteilungen an den Baumknoten

Zur Implementierung von Mischverteilungen sind in der letzten Zeit mehrere Verfahren vorgeschlagen worden.

- *Boulianne* [Boulianne & Kenny 96] verwendet als Mischverteilungen *tied mixtures*, d.h. zur Schätzung des Log-Likelihood-Gewinns beim Aufteilen eines Knoten müssen nur die Gewichte des Codebuchs neu geschätzt werden. Dies kann in einer Iteration über den Trainingsdaten geschehen. Zur Modellierung der Trainingsdaten werden allerdings kontinuierliche HMM eingesetzt, so daß wiederum eine Diskrepanz zwischen der Modellierung der Daten im Entscheidungsbaum bzw. im akustischen Modell selbst besteht.

- In [Chou & Reichl 98] wird ein interessantes Verfahren beschrieben, das die Modellierung der Beobachtungen mit Mischverteilungen mit minimalem Aufwand erlaubt. Dazu wird zur Bestimmung der Log-Likelihood an einem Knoten zunächst dieser

Knoten in einen Baum der Tiefe n expandiert ($n \geq 2$). Die Blätter dieses Baums werden als Mischverteilungskomponenten aufgefaßt, d.h. die Log-Likelihood an einem Knoten ergibt sich als die Summe der Log-Likelihoods der Blätter des expandierten Baums. Nach Bestimmung der Log-Likelihood wird die Expansion wieder rückgängig gemacht. Mit diesem einfachen Verfahren konnte die Fehlerrate auf verschiedenen Testkorpora zwischen 3% und 10% verbessert werden.

Für das zweite Verfahren können zwei Kritikpunkte genannt werden:

1. Die Zuordnung der Beobachtungen ist durch eine initiale Segmentierung der Trainingsdaten fest vorgegeben. U.U. könnte es vorteilhaft sein, für jede Aufteilung der Knoten diese Segmentierung neu zu schätzen.

2. Die Verteilung der Beobachtungen auf die Mischverteilungskomponenten ist durch die Zuordnung der Trainingsdaten zu den Triphonmodellen *und* der zur Verfügung stehenden Fragenmenge eingeschränkt.

Um diese Einschränkung in Bezug auf die möglichen Mischverteilungen zu verringern, könnte das in [Chou 91] beschriebene Verfahren eingesetzt werden, daß tatsächlich die optimale Aufteilung einer Menge von Daten in m Untermengen erlaubt. Diese Untermengen könnten dann als Mischverteilungskomponenten zur Estimation der Log-Likelihood eines Knotens verwendet werden.
Ein alternatives, wesentlich aufwendigeres Verfahren verwendet *n-best*-Listen von phonetischen Fragen für jeden Knoten und eine anschließende Estimation von echten Mischverteilungen für diese n Aufteilungen:

1. Ermittle für alle Blätter des aktuellen Baums die n besten Aufteilungen aufgrund von Gaußschen Einzelverteilungen (Standardmethode).

2. Schätze für diese n Aufteilungen auf den Trainingsdaten in mehreren Iterationen Gaußsche Mischverteilungen mit eingeschränkter Komponentenzahl (4-8) aufgrund einer festen Segmentierung.

3. Verwende für jedes Blatt die Aufteilung, die die größte Log-Likelihood-Verbesserung erzielt.

4. Reestimiere die Segmentierung der Trainingsdaten mit dem neuen Baum

Dies wird so lange iteriert, bis ein Stopkriterium erreicht ist (z.B. eine bestimmte Anzahl von Blättern).

8.2 Wortgrenzenmodellierung

Für die Wortgrenzenmodellierung mit wortübergreifenden Triphonen (einphasige Suche) werden folgende Verfahren vorgeschlagen:

- Automatisches Lernen der optimalen Koartikulationsschwelle,

- Rekombination nach der ersten Phonemgeneration,
- Modifiziertes Sprachmodell-Look-Ahead-Pruning,
- Phonem-Look-Ahead,
- Baumabhängiges Pruning.

8.2.1 Automatisches Lernen der optimalen Koartikulationsschwelle

Eine weniger harte Entscheidung könnte beispielsweise dadurch erfolgen, indem verschiedene Pause-Modelle definiert werden, die als Aussprachevarianten für die Pause zwischen den Worten verwendet werden. Je nach Länge oder Art der Pause wird dann die Modellierung des jeweiligen Wortes beim Training angepaßt. Problematisch ist, zu entscheiden, welches Pausemodell eine Koartikulation noch zuläßt und welches nicht. Eine mögliche Lösung ist die folgende:

- Definiere n_s Pausemodelle
- Definiere $2n_pn_s$ Diphone, die aus einem Pausemodell und einem der n_p Phonemmodelle bestehen
- Clustere diese Diphone auf eine Zahl nahe n_p zusammen, wobei nur Diphone mit demselben Pausemodell an derselben Position geclustert werden
- Verwende diese Diphoncluster als Kontext bei der Konstruktion der phonetschen Entscheidungsbäume

Dieses Verfahren legt automatisch für jedes Wortgrenzentriphon fest, bei welcher Pauselänge eine Koartikulation noch stattfindet.

8.2.2 Rekombination nach der ersten Phonemgeneration

Die Modifikationen des Suchalgorithmus bei der Verwendung von Wortgrenzenmodellen werden in erster Linie in der ersten bzw. letzten Phonemgeneration des lexikalischen Baums durchgeführt. Die dazwischenliegenden Phonemgenerationen und damit der Suchraum in diesen Generationen ändert sich dagegen nicht. Insbesondere ergibt sich nach der ersten Phonemgeneration keine Abhängigkeit bzgl. des linken phonetischen Kontexts des Baums (Koartikulation oder keine Koartikulation) mehr, so daß ab dort die Hypothesen mit gleichem Vorgängerwort und gleichem Triphon in der ersten Generation des Baums rekombiniert werden können (siehe Abbildung 8.1).
Durch diese Rekombination kann der potentielle Suchraum theoretisch um den Faktor 2 reduziert werden, da die Verdopplung des Suchraums durch die Abhängigkeit der Worthypothesen von der vorangegangenen Art der Wortgrenze (Koartikulation oder keine Koartikulation) bei Anwendung dieser Rekombination nach der ersten Phonemgeneration nicht mehr besteht.

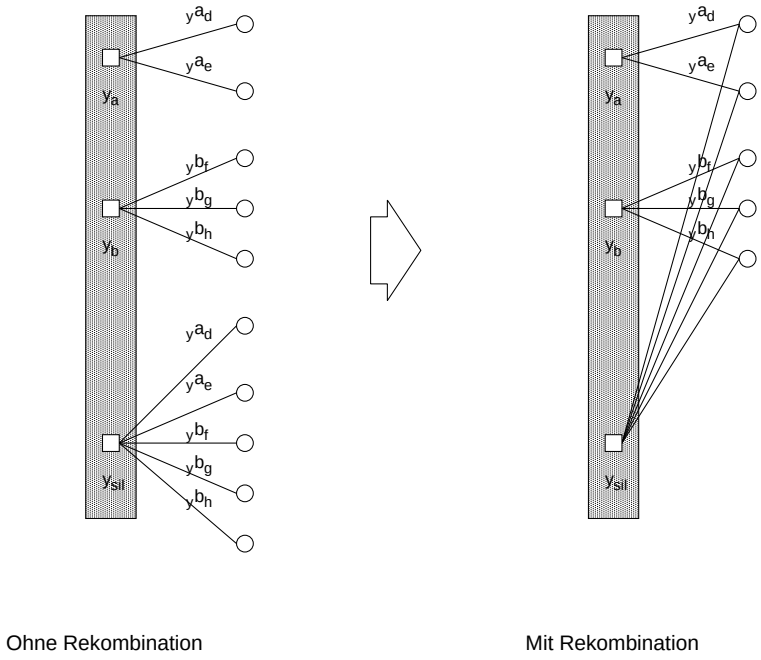


Abbildung 8.1: Rekombination nach der ersten Phonemgeneration

8.2.3 Modifiziertes Sprachmodell-Look-Ahead-Pruning

Beim Sprachmodell-Look-Ahead-Pruning [Ortmanns 98] werden die n-gramm-Wahrscheinlichkeiten des Sprachmodells, die im Prinzip erst bei Erreichen eines Wortendes im lexikalischen Baum zu einer Hypothesenbewertung hinzuaddiert werden können, über den lexikalischen Baum verteilt und dann zur Reduktion der aktiven Hypothesen während der Suche verwendet (*Pruning*). Dazu wird für jeden Knoten des lexikalischen Baums die maximale Sprachmodellwahrscheinlichkeit $\pi_v(s)$ aller über diesen Zustand erreichbarer Wortenden diesem Zustand s zugeordnet (siehe Abbildung 8.2).

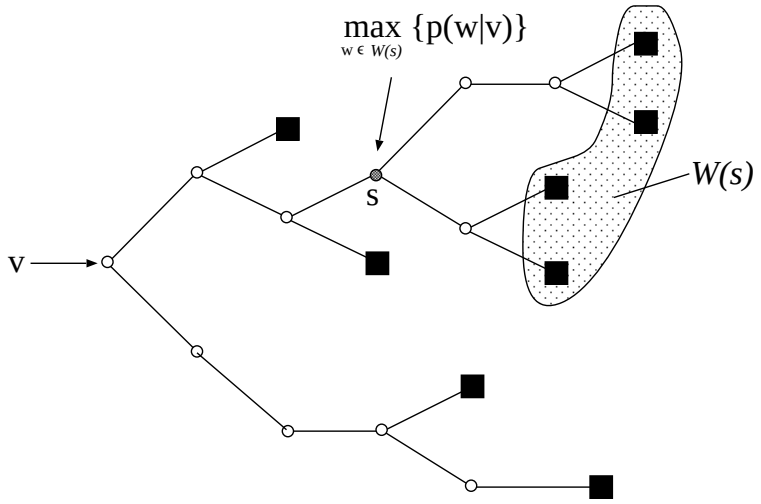


Abbildung 8.2: Prinzip des Bigramm-Sprachmodell-Look-Ahead

Diese verteilten Sprachmodellwahrscheinlichkeiten $\pi_v(s)$ werden dann in die Bewertung

der Hypothesen an diesem Knoten integriert:

$$\tilde{Q}_v(t,s) := \pi_v(s) \cdot Q_v(t,s)$$

wobei $Q_v(t,s)$ die Bewertung der Hypothese zum Zeitpunkt t am Zustand s darstellt. Mit dieser modifizierten Bewertung kann dann die Zahl der aktiven Hypothesen reduziert werden, indem man nur die Hypothesen weiterverfolgt, für die gilt:

$$\tilde{Q}_v(t,s) < f_{AC} \cdot \tilde{Q}_{AC}(t)$$

f_{AC} ist der sogenannte *Pruning-Schwellwert*, $Q_{AC}(t)$ die Bewertung der *besten* Hypothese zum Zeitpunkt t. Mit dem Parameter f_{AC} kann eingestellt werden, wie stark die Anzahl der Hypothesen durch das Pruning reduziert wird.
Dieses Konzept kann für die Wortgrenzensuche weiter verfeinert werden. Bei der Auffächerung der Wortenden wird zu einem Wortende gleichzeitig das Nachfolgephonem hypothetisiert (ausgenommen die Hypothese für Nicht-Koartikulation). Daher könnte man zusätzlich zur Sprachmodellwahrscheinlichkeit die Wahrscheinlichkeit des Nachfolgephonems, gegeben das Vorgängerwort, in die Bewertung der Hypothesen integrieren. Diese Wahrscheinlichkeit ist folgendermaßen definiert: In einem Wortbigramm vw habe ein Wort w die phonetische Transkription $c_1(w)c_2(w)\ldots c_n(w)$. Dann definiert man

$$P(c|v) = \sum_{w:c_1(w)=c} P(w|v)$$

als die Wahrscheinlichkeit, daß das Phonem c auf das Wort v folgt. Die Summe wird dabei über alle Wörter w des Lexikons gebildet, deren erstes Phonem $c_1(w)$ dem Phonem c entspricht. Diese Wahrscheinlichkeit wird dann ähnlich wie die verteilte Sprachmodellwahrscheinlichkeit für das Pruning der aktiven Hypothesen verwendet. Dazu werden diese Wahrscheinlichkeiten an die jeweiligen Äste des aufgefächerten Wortendes geschrieben. Für den Ast, der die Nicht-Koartikulation repräsentiert, wird die Summe über alle Wahrscheinlichkeiten dieser Auffächerung verwendet (siehe Abbildung 8.3). Während der Suche werden dann diese Wahrscheinlichkeiten in die Bewertung der Hypothesen miteinbezogen:

$$\tilde{Q}_v(t,s) := P(c_s|v) \cdot \pi_v(s) \cdot Q_v(t,s)$$

wobei s dem ersten Zustand des Astes und c_s dem rechten Kontextphonem des jeweiligen Astes der Auffächerung entspricht. Diese Phonemwahrscheinlichkeit muß allerdings, anders als die verteilte Sprachmodellwahrscheinlichkeit $\pi_v(s)$, beim Start des Folgebaums wieder abgezogen werden, da ansonsten die Bewertung durch das Sprachmodell nicht mehr korrekt wäre.

8.2.4 Phonem-Look-Ahead

Ähnlich wie das im vorherigen Abschnitt beschriebene Verfahren zielt auch die Verwendung des Phonem-Look-Ahead darauf ab, die Zahl der Hypothesen im Bereich der Auffächerung der letzten Phonemgeneration zu verringern. Da in der Menge der Folgeäste zu einem Ast des lexikalischen Baums die verwendeten Modelle relativ ähnlich sind

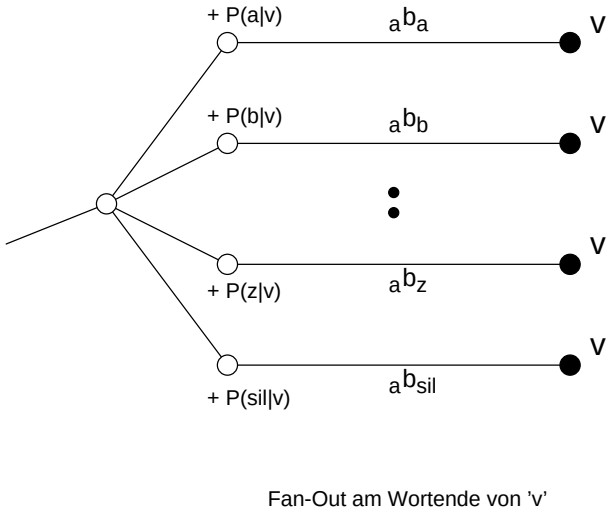


Abbildung 8.3: Addition der logarithmierten Phonemwahrscheinlichkeiten

(die Triphone unterscheiden sich nur durch den rechten Kontext), ist die Wahrscheinlichkeit, daß innerhalb dieser Menge von Folgeästen Hypothesen verworfen werden, relativ gering. Eine Möglichkeit, dies zu verbessern, ist die Verwendung eines *Phonem-Look-Ahead*-Prunings. Beim Phonem-Look-Ahead werden relativ einfache Modelle verwendet (Monophone), um mit Hilfe eines einfachen Suchalgorithmus ein oder zwei Phonemgenerationen ab dem aktuellen Zeitpunkt t im voraus zu bewerten, diese Bewertung kann dann dazu verwendet werden, zusätzlich Hypothesen, die bzgl. dieser Vorbewertung schlecht waren, zu prunen (siehe [Ortmanns 98]).
Im Fall der Wortgrenzensuche kann der Phonem-Look-Ahead über zwei Phonemgenerationen dazu verwendet werden, die Hypothesen innerhalb der Auffächerung, die ja abhängig vom rechten Kontextphonem vorgenommen wurde, zu prunen. Dazu wird bei Erreichen der letzten Phonemgeneration ein Phonem-Look-Ahead gestartet, der dann sowohl das Folgephonem als auch dessen rechtes Kontextphonem vorbewertet. Diese Vorbewertung wird dann zur Reduktion der Hypothesenanzahl innerhalb der Auffächerung verwendet (siehe Abbildung 8.4).

8.2.5 Baumabhängiges Pruning

Die in [Ortmanns 98] beschriebenen und in diesem System verwendeten Pruningverfahren reduzieren die Anzahl der Hypothesen aufgrund eines globalen Maximums bzgl. eines Zeitpunktes t. D.h., zu einem Zeitpunkt t werden *alle* aktiven Hypothesen zur Ermittlung der maximalen Bewertung herangezogen. Dies ist aber nicht unbedingt optimal, da die Bewertung Q einer Hypothese die gesamten bis dahin addierten lokalen Bewertungen für die durch diese Hypothese repräsentierte Wortfolge enthält. Daher werden die Hypothesen der aufgrund der initialen Bewertung Q gut bewerteten Baumkopien mit hoher Wahrscheinlichkeit gar nicht verworfen, während die Hypothesen der aufgrund der initialen Bewertung Q' schlecht bewerteten Baumkopien nur zum Teil weiterverfolgt werden oder ganz aussterben. D.h. innerhalb einer Baumkopie können u.U. aufgrund der *lokalen* akustischen Bewertungen gar keine Hypothesen verworfen werden, weil die "Vorbewertung"

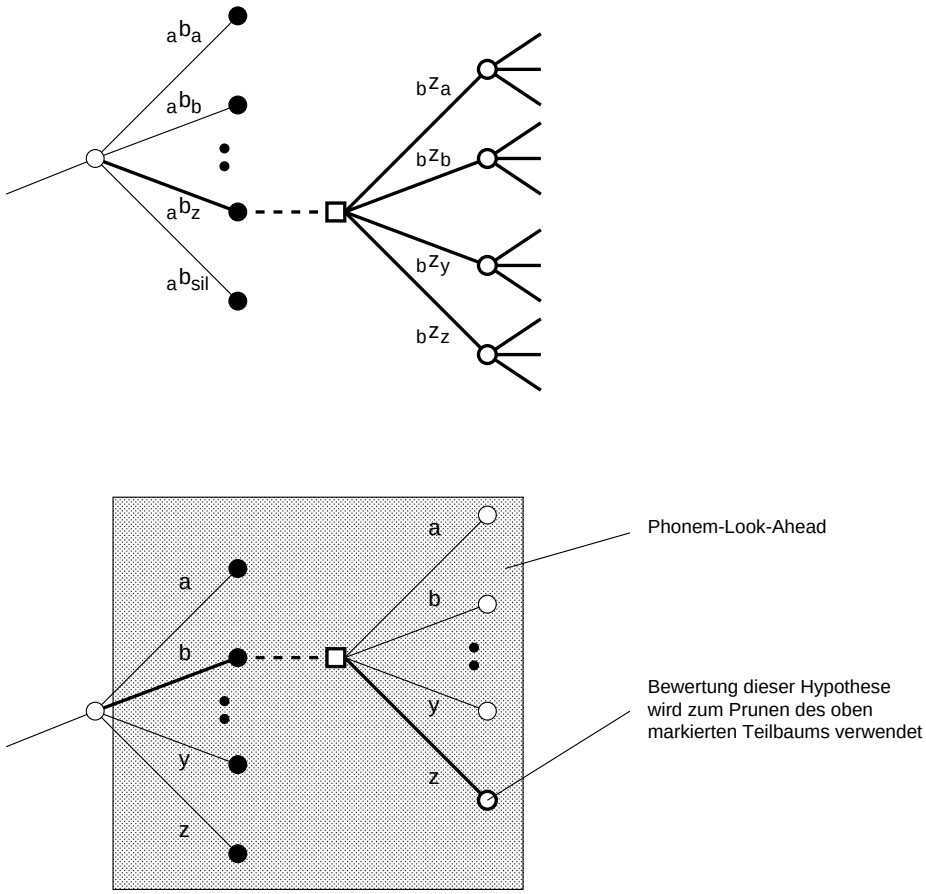


Abbildung 8.4: Phonem-Look-Ahead

durch die Historie (Akustik und Sprachmodell) im Vergleich zu anderen Baumkopien zu gut ist. Würde nun der akustische Pruningschwellwert so weit gesenkt werden, daß auch in diesen Baumkopien akustisch schlecht bewertete Hypothesen verworfen würden, würden die meisten anderen Baumkopien ganz aussterben, wodurch das Finden der *global* besten Wortfolge nicht mehr möglich wäre.

Eine Verbesserung stellt die zusätzliche Verwendung von *baumabhängigen* maximalen Bewertungen dar. Diese Bewertungen $Q_{AC,v}$ werden zu jedem aktiven Baum getrennt berechnet und dann auch nur innerhalb dieses Baumes zum Pruning verwendet. Der Vorteil ist, daß nun die für das Pruning verwendeten Maxima die unterschiedlichen Bewertungen durch die verschiedenen Startbewertungen Q der Baumkopien berücksichtigen (durch die Historie v), wodurch auch in den bzgl. des Sprachmodells "guten" Baumkopien noch akustisch geprunt werden kann (siehe Abbildung 8.5).

Eine Fortführung dieses Gedankens besteht nun darin, im Hinblick auf die in den letzten beiden Abschnitten beschriebenen Pruningmethoden solche spezifischen Maxima Q_{AC} auch für jede Auffächerung innerhalb eines Baums zu verwenden. Denn die beschriebenen Pruningmethoden können nur dann die Zahl der Hypothesen innerhalb der Auffächerung spürbar verringern, wenn die Bewertungen dieser Hypothesen innerhalb einer Auffächerung weit genug vom Maximum entfernt sind. Würde man dieses Maximum nun "global" für einen Baum bestimmen, würde man denselben Effekt für die Auffächerungen erhal-

ten wie für die Baumkopien, nämlich daß die Hypothesen einiger "gut" vorbewerteter Auffächerungen komplett überleben würden, während andere Auffächerungen ganz aussterben würden. Man berechnet also für jede Auffächerung einer Baumkopie ein Maximum, das dann zum Pruning mit den weiter oben beschriebenen Methoden verwendet wird.

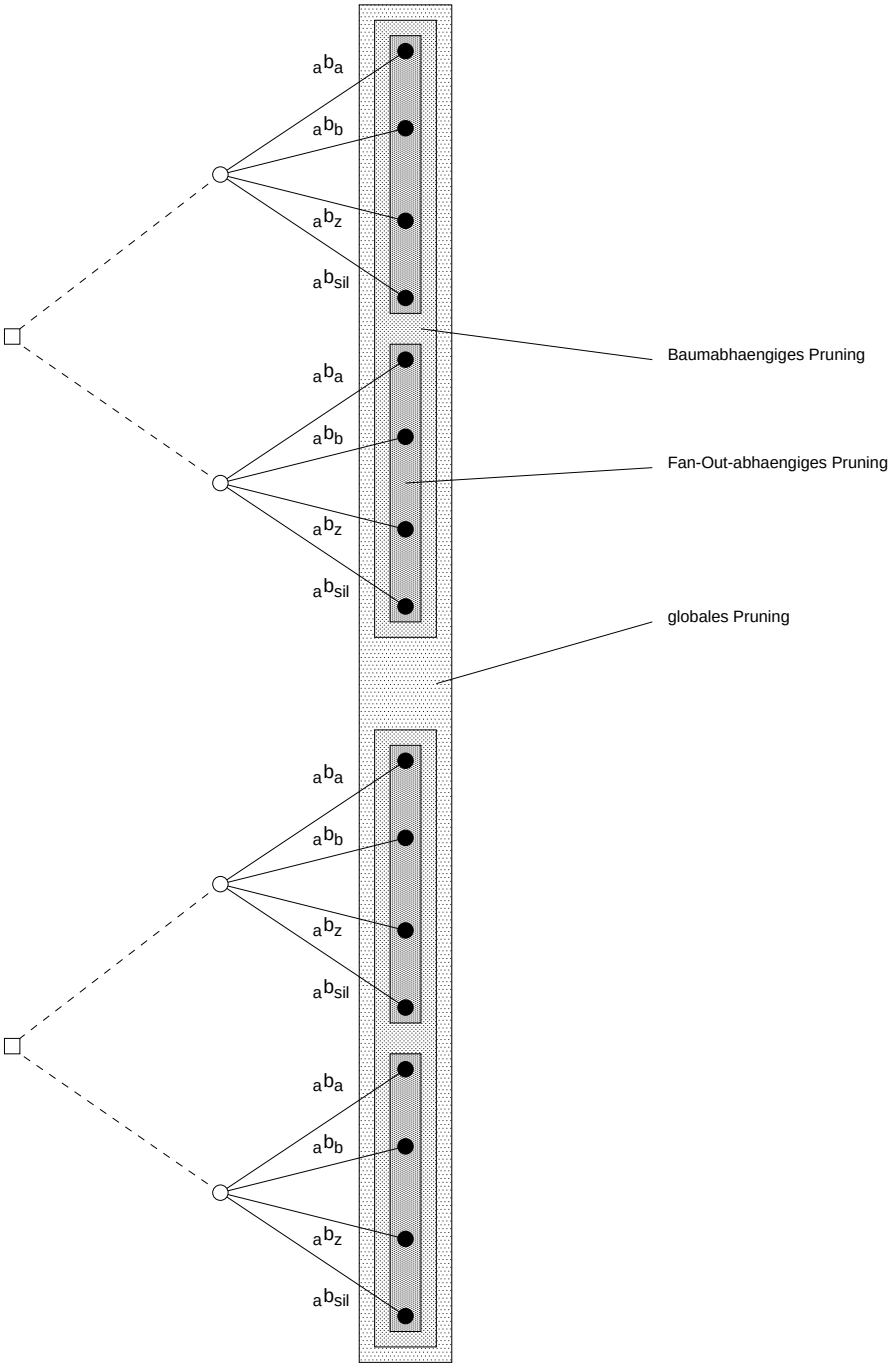


Abbildung 8.5: Baum- und Fan-Out-abhängiges Pruning

Kapitel 9

Zusammenfassung

In dieser Arbeit wurden verschiedene Aspekte bei der Verwendung von State-Tying mit phonetischen Entscheidungsbäumen in einem Spracherkennungssystem untersucht. Dies waren

- die Integration des State-Tying mit phonetischen Entscheidungsbäumen in ein bestehendes Spracherkennungssystem,
- die Untersuchung verschiedener Optimierungsmöglichkeiten des State-Tying,
- die Integration und Optimierung der Wortgrenzenmodellierung mit wortübergreifenden Triphonen,
- ein Vergleich der *n-best*- und der einphasigen Suche für die Verwendung von wortübergreifenden Triphonen und
- ein Verfahren zur automatischen Generierung von phonetischen Fragen.

Diese Verfahren wurden auf zwei verschiedenen Testkorpora evaluiert, dem *Wall Street Journal*-Korpus (englisch, gelesene Sprache) und dem *Verbmobil*-Korpus (deutsch, Spontansprache).

9.1 State-Tying

Bei der Integration des State-Tyings in das am Lehrstuhl verwendete Spracherkennungssystem wurde der Einfluß verschiedener Parameter des Verfahrens untersucht, wie z.B. die Zahl der Mischverteilungen des Entscheidungsbaums oder die optimale Menge der Triphone, die bei der Konstruktion des Baums betrachtet werden sollte. Mit diesen Optimierungen konnte die Fehlerrate auf den verwendeten Testkorpora um ca. 10-15% im Vergleich zu einem System ohne State-Tying gesenkt werden. Weiterhin wurden Erweiterungen des Standardverfahrens untersucht, wie z.B. die Verwendung eines Entscheidungsbaums für alle Phonemsegmente statt eines Baums pro Phonem und Segment, die Verwendung von geschlechtsabhängigen Modellen innerhalb der Knoten oder einer vollen Kovarianzmatrix für die Gaußmodelle der Triphonzustände. Dabei zeigte sich, daß das Basisverfahren gegenüber den Modifikationen gleiche oder sogar geringere Fehlerraten lieferte, was mit den

Resultaten anderer Publikationen auf diesem Gebiet übereinstimmt [Lazarides *et al.* 96] [Nock *et al.* 97]. Als Ergebnis wird in der aktuellen Version des Spracherkennungssystems am Institut das Basisverfahren des State-Tyings mit phonetischen Entscheidungsbäumen eingesetzt.

9.2 Wortgrenzenmodellierung

Aufbauend auf dieser Methode wurde die Verwendung der Wortgrenzenmodellierung mit wortübergreifenden Triphonen in diesem System untersucht. Dazu wurde sowohl der Trainingsalgorithmus modifiziert, als auch zwei Suchstrategien implementiert, die *n-best*-Methode und die einphasige Suche mit wortübergreifenden Triphonen. Beim Training wurde mit einer iterativen Schätzung der Pauselänge zwischen den Wörtern sichergestellt, daß die durch die Verwendung von wortübergreifenden Triphonen veränderte Segmentierung an den Wortgrenzen berücksichtigt wurde. Bei den Suchverfahren wurden folgende Punkte untersucht:

- *n-best*-Suche
  - Zeitdauer der Pause zwischen den Wörtern, ab der ein Koartikulationseffekt modelliert wird,
  - Länge der *n-best*-Liste,
  - Bestimmung der Wortgrenzenbehandlung (Koartikulation/keine K.) in der ersten oder in der zweiten Suchphase,
  - Auswirkung von variierender Hypothesenzahl in der ersten Suchphase,
  - Interpolation zwischen rein wortinternen und wortübergreifenden Modellen.
- einphasige Suche
  - Zeitdauer der Pause zwischen den Wörtern, ab der ein Koartikulationseffekt modelliert wird,
  - Interpolation zwischen rein wortinternen und wortübergreifenden Modellen.

Durch die Optimierung der Wortgrenzenmodellierung mit wortübergreifenden Triphonen konnte die Fehlerrate auf den verwendeten Testkorpora noch einmal um ca. 12-18% gesenkt werden. Weiterhin wurde die Methode der Interpolation zwischen rein wortinternen Modellen und den wortübergreifenden Triphonmodellen beschrieben und getestet. Mit dieser Methode läßt sich bei der *n-best*-Suche der beim Basisverfahren beobachtete Effekt vermeiden, daß für sehr große *n-best*-Satzlisten die Fehlerrate für die *n-best*-Suche wieder ansteigt.
Für die einphasige Suche hat sich die Interpolation als vorteilhaft herausgestellt. Während die Verbesserung der Fehlerrate durch wortübergreifende Triphone bei der einphasigen Suche ohne Interpolation nur ca. 6% relativ beträgt, erhält man mit Interpolation eine Verbesserung von ca. 10% relativ. Die in der Literatur beschriebenen Verbesserungen von 10-15% relativ ohne Interpolation wurden für die einphasige Suche nicht erreicht (z.B.

[Odell 95]). Der Grund dafür konnte in dieser Arbeit nicht gefunden werden. Allerdings deutet die starke Verbesserung durch die Interpolation darauf hin, daß es sich um ein Problem der hinreichend robusten Parameterschätzung handelt könnte.

9.3 Automatische Fragengenerierung

Weiterhin wurde ein neues Verfahren zur automatischen Generierung von Fragenlisten für das State-Tying mit phonetischen Entscheidungsbäumen entwickelt und evaluiert. Es basiert auf einem Bottom-Up-Clusterverfahren, das akustisch ähnliche Phoneme zusammenfaßt, die dann als phonetische Fragen verwendet werden können. Die mit diesen Verfahren erzielten Fehlerraten auf den Testkorpora zeigen, daß die mit dem automatischen Verfahren generierten Fragenlisten zu vergleichbaren Fehlerraten wie die aufgrund von phonetischem Wissen definierten Fragen führen.

Anhang A

Volle Kovarianzmatrix

Gegeben sei eine Gaußverteilung $p(\vec{x})$ mit Mittelwert $\vec{m}$ und Kovarianzmatrix Σ für die Beobachtungsvektoren $\vec{x}_1^N = \vec{x}_1 \ldots \vec{x}_N$ der Dimension D.

$$p(\vec{x}_n) = \frac{1}{(2\pi)^{\frac{D}{2}} \det(\Sigma)^{\frac{1}{2}}} \exp(-\frac{1}{2}((\vec{x} - \vec{m})^T \Sigma^{-1} (\vec{x} - \vec{m}))) \tag{A.1}$$

Dann berechnet sich die Log-Likelihood für $\vec{x}_1^N$ zu

$$\begin{aligned} LL(\vec{m}, \Sigma | \vec{x}_1^N) &= \sum_{n=1}^{N} \log p(\vec{x}_n) \\ &= \sum_{n=1}^{N} \log(\frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}} \exp(-\frac{1}{2}((\vec{x}_n - \vec{m})^T \Sigma^{-1} (\vec{x}_n - \vec{m})))) \\ &= \sum_{n=1}^{N} \log(\frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}}) - \frac{1}{2}((\vec{x}_n - \vec{m})^T \Sigma^{-1} (\vec{x}_n - \vec{m})) \\ &= \sum_{n=1}^{N} \log(\frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}}) - \frac{1}{2} \sum_{n=1}^{N} ((\vec{x}_n - \vec{m})^T \Sigma^{-1} (\vec{x}_n - \vec{m})) \end{aligned} \tag{A.2}$$

Der Mittelwert m läßt sich wie im Fall einer diagonalen Kovarianzmatrix bestimmen, die Elemente σ_{kj} der Kovarianzmatrix selbst zu

$$\begin{aligned}
\sigma_{kj} &= \frac{1}{N}\sum_{n=1}^{N}(x_{n,k}-m_k)(x_{n,j}-m_j) \\
&= \frac{1}{N}\sum_{n=1}^{N} x_{n,k}x_{n,j} - m_k x_{n,j} - m_j x_{n,k} + m_k m_j \\
&= \frac{1}{N}\left(\sum_{n=1}^{N} x_{n,k}x_{n,j} - \sum_{n=1}^{N} m_k x_{n,j} - \sum_{n=1}^{N} m_j x_{n,k} + \sum_{n=1}^{N} m_k m_j\right) \\
&= \frac{1}{N}(Q_{kj} - \frac{S_k S_j}{N} - \frac{S_j S_k}{N} + N\frac{S_j S_k}{N^2}) \\
&= \frac{Q_{kj}}{N} - \frac{S_k S_j}{N^2}, \qquad \text{(A.3)}
\end{aligned}$$

wobei

$$\begin{aligned}
Q_{kj} &= \sum_{n=1}^{N} x_{n,k}x_{n,j} \\
S_k &= \sum_{n=1}^{N} x_{n,k}
\end{aligned}$$

Sei nun

$$\Sigma = A\Omega A^T \qquad \text{(A.4)}$$

wobei A eine orthonormale Matrix und Ω eine Diagonalmatrix mit Diagonalelementen $\omega_1, \ldots, \omega_N$ ist. Dann läßt sich Σ^{-1} umformen zu

$$\begin{aligned}
\Sigma^{-1} &= (A\Omega A^T)^{-1} \\
&= (A^T)^{-1}\Omega^{-1}A^{-1} \qquad \text{(A.5)} \\
&= A\Omega^{-1}A^T
\end{aligned}$$

Damit kann man den zweiten Term der rechten Seite von Gleichung (2) umformen zu

$$\begin{aligned}
\sum_{n=1}^{N}(\vec{x}_n-\vec{m})^T\Sigma^{-1}(\vec{x}_n-\vec{m}) &= \sum_{n=1}^{N}(\vec{x}_n-\vec{m})^T A\Omega^{-1}A^T(\vec{x}_n-\vec{m}) \\
&= \sum_{n=1}^{N}(\vec{y}_n-\vec{\mu})^T\Omega^{-1}(\vec{y}_n-\vec{\mu}) \\
&= \sum_{n=1}^{N}\sum_{d=1}^{D}\frac{(y_{n,d}-\mu_d)^2}{\omega_D^2} \\
&= \sum_{d=1}^{D}\sum_{n=1}^{N}\frac{(y_{n,d}-\mu_d)^2}{\omega_D^2} \\
&= \sum_{d=1}^{D}\frac{N\omega_D^2}{\omega_D^2} \\
&= ND
\end{aligned} \tag{A.6}$$

Die negative Log-Likelihood $LL(\vec{m},\Sigma|x_1^N)$ für eine Gaußverteilung mit voller Kovarianzmatrix ohne Glättung der Varianzen ist damit

$$\begin{aligned}
LL(\vec{m},\Sigma|x_1^N) &= \sum_{n=1}^{N}\log\left(\frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma|^{\frac{1}{2}}}\right)-\frac{1}{2}ND \\
&= \sum_{n=1}^{N}-\frac{1}{2}(D+D\log(2\pi)+\log(|\Sigma|)) \\
&= -\frac{N}{2}(D+D\log(2\pi)+\log(|\Sigma|))
\end{aligned} \tag{A.7}$$

Will man die Varianzen glätten, muß die Umformung von Gleichung (3) modifiziert werden:

$$
\begin{aligned}
&\sum_{n=1}^{N}(\vec{x}_n-\vec{m})^T\Sigma^{-1}(\vec{x}_n-\vec{m})\\
&=\sum_{n=1}^{N}\vec{x}_n^T\Sigma^{-1}\vec{x}_n-\vec{x}_n^T\Sigma^{-1}\vec{m}-\vec{m}^T\Sigma^{-1}\vec{x}_n+\vec{m}^T\Sigma^{-1}\vec{m}\\
&=\sum_{n=1}^{N}\vec{x}_n^T\Sigma^{-1}\vec{x}_n-2\vec{x}_n^T\Sigma^{-1}\vec{m}+\vec{m}^T\Sigma^{-1}\vec{m}\\
&=\sum_{n=1}^{N}\sum_{j=1}^{D}x_{n,j}\sum_{k=1}^{D}x_{n,k}\sigma_{kj}^{(-1)}-2\sum_{j=1}^{D}m_j\sum_{k=1}^{D}x_{n,k}\sigma_{kj}^{(-1)}+\sum_{j=1}^{D}m_j\sum_{k=1}^{D}m_k\sigma_{kj}^{(-1)}\\
&=\sum_{n=1}^{N}\sum_{j=1}^{D}\sum_{k=1}^{D}x_{n,j}x_{n,k}\sigma_{kj}^{(-1)}-2\sum_{j=1}^{D}\sum_{k=1}^{D}m_jx_{n,k}\sigma_{kj}^{(-1)}+\sum_{j=1}^{D}\sum_{k=1}^{D}m_jm_k\sigma_{kj}^{(-1)}\\
&=\sum_{n=1}^{N}\sum_{j=1}^{D}\sum_{k=1}^{D}x_{n,j}x_{n,k}\sigma_{kj}^{(-1)}-2m_jx_{n,k}\sigma_{kj}^{(-1)}+m_jm_k\sigma_{kj}^{(-1)} \qquad \text{(A.8)}\\
&=\sum_{j=1}^{D}\sum_{k=1}^{D}\sigma_{kj}^{(-1)}\left(\sum_{n=1}^{N}x_{n,j}x_{n,k}-2\sum_{n=1}^{N}m_jx_{n,k}+\sum_{n=1}^{N}m_jm_k\right)\\
&=\sum_{j=1}^{D}\sum_{k=1}^{D}\sigma_{kj}^{(-1)}\left(Q_{jk}-2\frac{S_jS_k}{N}+\frac{S_jS_k}{N}\right)\\
&=\sum_{j=1}^{D}\sum_{k=1}^{D}\sigma_{kj}^{(-1)}\left(Q_{jk}-\frac{S_jS_k}{N}\right)\\
&=N\sum_{j=1}^{D}\sum_{k=1}^{D}\sigma_{kj}^{(-1)}\hat{\sigma}_{kj}
\end{aligned}
$$

$\sigma_{kj}^{(-1)}$ kann dann entsprechend geglättet werden.

Anhang B

Glättung der Varianzen

Zur Glättung der Varianzen wurden folgende Methoden verwendet:

1. Untere Schranke für Varianz (Clipping)

   Diese untere Schranke wird global für alle Dimensionen der Gauß–Verteilung festgelegt. Dazu wurde zu einer Auswahl von Triphonen (mindestens 50 Beobachtungen im Trainingstext) das globale Minimum der Knotenvarianzen bestimmt (mindestens 500 Beobachtungen). Dieses globale Minimum wurde dann als untere Schranke $\bar{\sigma}^2_{min}$ für die Knotenvarianzen $\hat{\sigma}^2$ verwendet:

   $$\hat{\sigma}^2 = \begin{cases} \bar{\sigma}^2 & \bar{\sigma}^2 > \bar{\sigma}^2_{min} \\ \bar{\sigma}^2_{min} & \text{sonst} \end{cases}$$

   $\bar{\sigma}^2$ ist bestimmt durch die Formel

   $$\bar{\sigma}^2 = \left(\sum_{x \in K} x^2 \right) - \hat{\mu}^2$$

   Der aktuelle Wert für die untere Schranke $\bar{\sigma}^2_{min}$ ist 3000.

2. Glättung der Varianz mit der Varianz an der Wurzel

   Die Glättung der Knotenvarianz mit der Wurzelvarianz wird über die Formel

   $$\hat{\sigma}^2_{smoothed} = \lambda(N) \cdot \hat{\sigma}^2_{node} + (1 - \lambda(N)) \cdot \hat{\sigma}^2_{root}$$

   erreicht, wobei λ durch die Formel

   $$\lambda(N) = \frac{1}{1 + e^{-\frac{N-\delta}{\alpha}}}$$

   abhängig von der Zahl der Beobachtungen am Knoten N berechnet wird. δ legt das Zentrum und α die Steigung der Sigmoidfunktion fest.

   Die aktuellen Werte für δ und α sind 500 bzw. 20.

Anhang C

Fragenlisten

C.1 Wall Street Journal

```
BOUNDARY #
ALVEOLAR-STOP d t
LABIAL-STOP b p
DENTAL dh th
LIQUID l r ur
LW l w
S/SH s sh
VELAR-STOP g k
LQGL-BACK l r ur w
NASAL m n
L-NASAL m n um un
R-NASAL m n ng
L-VELAR ng g k
R-VELAR g k
R-VOICELESS-FRIC th s sh f
L-VOICELESS-FRIC th s sh f j
L-VOICED-FRIC dh z v j zh
R-VOICED-FRIC dh z v j zh ch
LIQUID-GLIDE l r ur w h
S/Z/SH/ZH s z sh zh
W-GLIDE aw awh ow w
PALATL y ch sh
Y-GLIDE ey y
L-LABIAL w m b p v f um
R-LABIAL w m b p v f
HIGH-VOWEL ee ih UH uh y
LAX-VOWEL ee eh ih UH uh ah oh ooh oo awh
LOW-VOWEL ae aa aw
ORAL-STOP2 p t k
R-ORAL-STOP3 b d g
L-ORAL-STOP3 b d g ng
```

```
ALVEOLAR n d t s z un
DIPHTHONG aw awh ey ow
R-FRICATIVE dh th s sh z v f zh
L-FRICATIVE dh th s sh z v f zh j ch
ROUND-VOCALIC UH uh ow w
FRONT-R ae ee eh ih ey ah y aw awh
TENSE-VOWEL ey ae ow aa aw awh
BACK-L UH uh ow aa l r ur w aw awh oh ooh oo
FRONT-L ae ee eh ih ey ah y
BACK-R UH uh ow aa l r ur w oh ooh oo
ORAL-STOP1 b d g p ch j
VOWEL ae ee eh ih UH uh ah aa aw ey ow oh ooh oo awh
SONORANT ae ee eh ih ey ah UH uh ur ow aa aw l r w y oh ooh awh
VOICED ae ee eh ih UH uh ah aa aw ey ow l r w y m n ng j b d dh g v z
   um un ul ur zh
```

C.2 Verbmobil

```
BOUNDARY #
K_A C S b d f g j k p s t v x z
K_B Hm N l m n r
K_C h
VOT C Hm N S f k m n p s t x
NOT_VOT S b d g j v z
PLOSIV b d g k p t
NOT_PLOSIV C S f j s v x z
APIKAL S d l n s t z
NOT_APIKAL C Hm N b f g j k m p v x
LABIAL Hm b f m p v
NOT_LABIAL C N S d g j k n s t x z
LATERAL l
NOT_LATERAL r
ENG s z
NOT_ENG S
FRIKATIV C S f j s v x z
NASAL Hm N m n
NOT_NASAL l r
VOKAL 2: 6 9 @ E E: HE I OY O U a~ aI aU a: a e: i: o: u: y:
LANG 2: E: HE aI aU a: e: i: o: u: y:
GERUNDET 2: y:
NOT_GERUNDET E E: HE e: i:
TIEF I U i: u: y:
MITTEL 2: O a~ e: o:
HOCH E E: HE a: a
DIPHTONG OY S aI aU pf tS ts
```

```
HAESITATION HE Hf Hm Hp
HAES_REST Hf Hp
NOT_HAES_REST HE Hm
```

Die Fragenlisten werden dann jeweils mit den Fragen nach einzelnen Phonemen ergänzt.

Anhang D

Abkürzungen und Symbole

D.1 Einführung

w	Wortindex
x	akustischer Vektor
N	Anzahl der Wörter eines Satzes
T	Anzahl der akustischen Vektoren eines gesprochenen Satzes
P	Wahrscheinlichkeitsfunktion
p	Wahrscheinlichkeitsdichtefunktion
FFT	schnelle Fouriertransformation
LDA	Lineare Diskiminantenanalyse
HMM	Hidden-Markov-Modell
D	Dimension des akustischen Vektors
μ	Mittelwert einer Gaußverteilung
σ^2	Varianz eines Gaußverteilung
v	Standardabweichung einer Laplaceverteilung
p_i	i-te Komponente einer Mischverteilung
c_i	Gewicht der i-ten Komponente einer Mischverteilung

s	Zustandsindex eines HMM
Θ	Parameter des akustischen Modells
$\alpha_t(s)$	Forward-Wahrscheinlichkeit
$\beta_t(s)$	Backward-Wahrscheinlichkeit
$\gamma_t(s)$	Zustandswahrscheinlichkeit

D.2 State-Tying mit Entscheidungsbäumen

w	Wortindex
x	akustischer Vektor
N	Anzahl der Wörter eines Satzes
T	Anzahl der akustischen Vektoren eines gesprochenen Satzes
P	Wahrscheinlichkeitsfunktion
p	Wahrscheinlichkeitsdichtefunktion
HMM	Hidden-Markov-Modell
s	Zustandsindex eines HMM
D	Dimension des akustischen Vektors
μ	Mittelwert einer Gaußverteilung
σ^2	Varianz eines Gaußverteilung
L	Likelihood-Funktion
LL	Log-Likelihood-Funktion
C	Zustandscluster
ΔLL	Log-Likelihood-Differenz
τ_{dist}	Clusterschwellwert Abstand
τ_{minobs}	Clusterschwellwert Anzahl Beobachtungen
B	Blatt des phonetischen Entscheidungsbaums
Q	phonetische Frage

D.3 Wortgrenzenmodellierung

v, w	Wortindizes

t	Zeitpunkt
τ	Zeitpunkt einer Wortgrenze (Wortgraph)
h	akustische Wortbewertung (Wortgraph)
T	Anzahl der akustischen Vektoren eines gesprochenen Satzes
N_{sil}	Pauseschwellwert
n	Länge der *n-best*-Liste
H	Satzbewrtung (akustisch & Sprachmodell)
λ	Interpolationsfaktor
τ	Anzahl Pausezustände zwischen Wörtern

D.4 Automatische Fragengenerierung

HMM	Hidden-Markov-Modell
C	Zustandscluster
P	Wahrscheinlichkeitsfunktion
tr	Triphon
Tr	Triphonmenge
tr^o	Im Trainingskorpus gesehenes Triphon
tr^u	Im Trainingskorpus nicht gesehenes Triphon
m	zentrales Phonem zu einem Triphon
s	Segmentindex zu einem Triphon
l, r	linkes und rechtes Kontextphonem zu einem Triphon $\hat{\delta}$ modifizierte Log-Likelihood-Differenz
ΔLL	Log-Likelihood-Differenz
N_m	Anzahl der Phoneme

Literaturverzeichnis

[Alleva *et al.* 92] F. Alleva, H. Hon, X. Huang, M. Hwang, R. Rosenfeld, R. Weide, "Applying SPHINX-II to the DARPA Wall Street Journal CSR Task," *Proc. DARPA Speech and Natural Language Workshop*, Harriman, NY, Februar 1992, pp. 393-398.

[Alleva *et al.* 97] F. Alleva, "Search Organization in the Whisper Continuous Speech Recognition System," , in S Furui, B.-H. Juang, W. Chou (eds.): *Proc. 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, Santa Barbara, CA, Dezember 1997, pp. 295–302.

[Aubert *et al.* 93] X. Aubert, R. Haeb-Umbach, H. Ney, "Continuous Mixture Densities and Linear Discriminant Analysis for Context-Dependent Acoustic Models," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Minneapolis, MN, April 1993, pp. 648-651.

[Aubert *et al.* 94] X. Aubert, C. Dugast, H. Ney, V. Steinbiss, "Large Vocabulary Continuous Speech Recognition of Wall Street Journal Corpus," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Adelaide, Australien, April 1994, pp. 129-132.

[Aubert *et al.* 96] X. Aubert, P. Beyerlein, M. Ullrich, "A Bottom-Up Approach for Handling Unseen Triphones in Large Vocabulary Continuous Speech Recognition," *Proc. Int. Conf. on Spoken Language Processing*, Philadelphia, PA, Oktober 1996, pp. 14-17.

[Aust 98] H. Aust, *Sprachverstehen und Dialogmodellierung in natürlichsprachlichen Informationssystemen*, Dissertation, Aachener Informatik-Berichte, RWTH Aachen, 1998.

[Bahl *et al.* 83] L.R. Bahl, F. Jelinek, R.L. Mercer, "A Maximum Likelihood Approach to Continuous Speech Recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 5, pp. 179-190, März 1983.

[Bahl *et al.* 91] L.R. Bahl, P.V. de Souza, P.S. Gopalakrishnan, D. Nahamoo, M.A. Picheney, "Context Dependent Modelling of Phones in Continuous Speech using Decision Trees," *Proc. DARPA Speech and Natural Language Workshop*, Pacific Grove, CA, Februar 1991, pp. 264-269.

[Baker 75] J.K. Baker, "Stochastic Modelling for Automatic Speech Understanding," in D.R. Reddy (ed.): *Speech Recognition*, Academic Press, New York, NY, pp. 512-542, 1975.

[Baum *et al.* 70] L.E. Baum, T. Petrie, G. Soules, N. Weiss, "A Maximization Technique Occuring in the Statistical Analysis of Probabilistic Functions of Markov Chains," *The Annals of Mathematical Statistics*, 41, pp. 164-171, 1970.

[Bellman 57] R.E. Bellman, *Dynamic Programming*, Princeton University Press, Princeton, NJ, 1957.

[Beulen *et al.* 95] K. Beulen, L. Welling, H. Ney, "Experiments with Linear Feature Extraction in Speech Recognition," *Proc. Europ. Conf. on Speech Communication and Technology*, Madrid, Spanien, September 1995, pp. 1415-1418.

[Beulen *et al.* 97] K. Beulen, E. Bransch, H. Ney, "State Tying for Context Dependent Phoneme Models," *Proc. Europ. Conf. on Speech Communication und Technology*, Rhodos, Griechenland, September 1997, pp. 1179-1182.

[Beulen *et al.* 98] K. Beulen, H. Ney, "Automatic Question Generation for Decision Tree Based State Tying," *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, Seattle, WA, September 1998, pp. 805-808.

[Beyerlein *et al.* 97] P. Beyerlein, M. Ullrich, P. Wilcox, "Modelling and Decoding of Crossword Context Dependent Phones in the Philips Large Vocabulary Continuous Speech Recognition System," *Proc. Europ. Conf. on Speech Communication and Technology*, Rhodos, Griechenland, September 1997, pp. 1163-1166.

[Boulianne & Kenny 96] G. Boulianne, P. Kenny, "Optimal Tying of HMM Mixture Densities using Decision Trees," *Proc. Int. Conf. on Spoken Language Processing*, Philadelphia, PA, Oktober 1996, pp. 350-353.

[Bransch 95] E. Bransch, *Parameterreduktion bei kontextabhängigen Phonemmodellen durch kombinierte Top-Down- und Bottom-Up-Entscheidungsbäume*, Diplomarbeit am Lehrstuhl für Informatik VI, RWTH Aachen, Aachen, September 1996.

[Breiman *et al.* 84] L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, *Classification and Regression Trees*, The Wadsworth Statistics/Probability Series, Belmont, CA, 1984.

[Bridle & Sedgewick 77] J. S. Bridle, N. C. Sedgewick, "A Method for Segmenting Acoustic Patterns, with Applications to Automatic Speech Recognition," *Proc. 1977 IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Hartford, CN, Mai 1977, pp. 656-659.

[Bridle *et al.* 82] J.S. Bridle, M.D. Brown, R.M. Chamberlain, "An Algorithm for Connected Word Recognition," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Paris, Frankreich, Mai 1982, pp. 899-902.

[Chen & Shrager 89] F.R. Chen, J. Shrager, "Automatic Discovery of Contextual Factors Describing Phonological Variation," *Proc. DARPA Speech and Natural Language Workshop*, Philadelphia, PA, Februar 1989, pp. 284-289.

[Chou 91] P.A. Chou, "Optimal Partitioning for Classification and Regression Trees," *Proc. IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 13, No. 4, April 1991, pp. 340-354.

[Chou & Reichl 98] W. Chou, W. Reichl, "High Resolution Decision Tree Based Acoustic Modelling beyond CART," *Proc. Int. Conf. on Spoken Language Processing*, Sydney, Australien, November 1998.

[Demuynck *et al.* 97] K. Demuynck, J. Duchateau, D. v. Compernolle, "A Static Lexicon Network Representation for Cross-Word Context Dependent Phones," *Proc. Europ. Conf. on Speech Communication and Technology*, Rhodos, Griechenland, September 1997, pp. 143-146.

[Dragon Web Site] *Dragon Systems, Inc.*, Spracherkennungssoftware, Newton, MA 02460, USA, http://www.dragonsys.com/, Februar 1999.

[Duda & Hart 73] R.O. Duda, P.E. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.

[Dugast *et al.* 95a] C. Dugast, R. Kneser, X. Aubert, S. Ortmanns, K. Beulen, H. Ney, "Continuous Speech Recognition Tests and Results for the NAB'94 Corpus," *Proc. ARPA Spoken Language Technology Workshop*, Austin, TX, Januar 1995, pp. 156-161.

[Dugast *et al.* 95b] C. Dugast, P. Beyerlein, R. Haeb-Umbach, "Application of Clustering Techniques to Mixture Density Modelling for Continuous Speech Recognition," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Detroit, MI, Mai 1995, pp. 524-527.

[Elting 99] Ch. Elting, "Wortgrenzenmodellierung und Suche für die automatische Spracherkennung," Diplomarbeit am Lehrstuhl für Informatik VI, RWTH Aachen, Aachen, Mai 1999.

[Haeb-Umbach & Ney 92] R. Haeb-Umbach, H. Ney, "Linear Discriminant Analysis for Improved Large Vocabulary Continuous Speech Recognition," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, San Francisco, CA, März 1992.

[Henrichson & Fu 69] J. Henrichson, K. Fu, "A Nonparametric Partitioning Procedure for Pattern Classification," *IEEE Transactions on Computers*, Vol. C-18, Mai 1969, pp. 604-624.

[Hon 92] H.-W. Hon, *Vocabulary-Independent Speech Recognition: The VOCIND System*, Ph.D. Thesis, School of Computer Science, Carnegie Mellon University, Pittsburg, PA, 1992.

[Hwang *et al.* 92] M.Y. Hwang, X. Huang, F. Alleva, "Predicting Unseen Triphones with Senones,", *Technischer Report*, No. 510.7808 C28R 93-139 2, Carnegie Mellon University, 1993.

[Hwang 93] M.Y. Hwang, "Subphonetic Acoustic Modeling for Speaker-Independent Continuous Speech Recognition," Ph.D. Thesis CMU-CS-93-230, Carnegie Mellon University, 1993.

[IBM Web Site] *International Business Machines* (IBM), Spracherkennungssoftware, Armonk, NY 10504, USA, http://www.software.ibm.com/speech/, Februar 1999.

[Jelinek 76] F. Jelinek, "Continuous Speech Recognition by Statistical Methods," *Proc. of the IEEE*, Vol. 64, No. 10, pp. 532-556, April 1976.

[Klovstad & Mondshein] J. W. Klovstad, L. F. Mondshein, "The CASPERS Linguistic Analysis System," *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol. 23, pp. 118-123, Februar 1975.

[Kramer 96] M. Kramer, *Bottom-Up-Clustering für Phonemmodelle in der Spracherkennung*, Diplomarbeit am Lehrstuhl für Informatik VI, RWTH Aachen, Aachen, März 1996.

[Ladefoged 82] P. Ladefoged, *A Course in Phonetics*, 2nd Edition, Harcourt Brace Jovanovich, Publishers, Orlando, FL, 1982.

[Lamel 95] L. Lamel, "Issues in Large Vocabulary, Multilingual Speech Recognition," *Proc. Europ. Conf. on Speech Communication and Technology*, Madrid, Spanien, September 1995, pp. 185-188.

[Lazarides *et al.* 96] A. Lazaridès, Y. Normandin, R. Kuhn, "Improving Decision Trees for Acoustic Modeling," *Proc. Int. Conf. on Spoken Language Processing*, Philadelphia, PA, Oktober 1996, pp. 1053-1056.

[Lee 88] K.-F. Lee, *Large Vocabulary Speaker Independent Continuous Speech Recognition: The SPHINX System*, Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA, April 1988.

[McAllaster *et al.* 97] D. McAllaster, L. Gillick, B. Peskin, "Acoustic Modeling Experiments," *Hub-5 Conversational Speech Recognition Workshop*, Maritime Institute of Technology, Linthicum Heights, MD, November 1997.

[Moore *et al.* 94] R. Moore, M. Russell, P. Nowell, S.N. Downey, S.R. Browning, "A Comparison of Phoneme Decision Tree (PDT) and Context Adaptive Phone (CAP) Based Approaches to Vocabulary-Independent Speech Recognition," *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, Adelaide, Australien, April 1994, pp. 541-544.

[Ney 84] H. Ney, "The Use of a One-Stage Dynamic Programming Algorithm for Connected Word Recognition," *IEEE Trans. on Acoustic, Speech and Signal Processing*, Vol. ASSP-32, No. 2, April 1984.

[Ney 90] H. Ney, "Acoustic Modelling of Phoneme Units for Continuous Speech Recognition," *Proc. Fifth Europ. Signal Processing Conf.*, Barcelona, Spanien, September 1990, pp. 65-72.

[Ney *et. al.* 94] H. Ney, X. Aubert, "A Word Graph Algorithm for Large Vocabulary Speech Recognition," *Proc. Int. Conf. on Spoken Language Processing*, Yokohama, Japan, Vol. 3, September 1994, pp. 1355-1358.

[Ney *et al.* 98] H. Ney, L. Welling, S. Ortmanns, K. Beulen, F. Wessel, "The RWTH Large Vocabulary Continuous Speech Recognition System," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Seattle, WA, Mai 1998, pp. 853-856.

[Nguyen & Schwartz 97] L. Nguyen, R. Schwartz, "Efficient 2-Pass N-Best Decoder," *Proc. Europ. Conf. on Speech Communication and Technology*, Rhodos, Griechenland, September 1997, pp. 167-170.

[Nock *et al.* 97] H. J. Nock, M. J. F. Gales, S. Young, "A Comparative Study of Methods for Phonetic Decision-Tree State Clustering,", *Proc. Europ. Conf. on Speech Communication and Technology*, Rhodos, Griechenland, September 1997, pp. 111-114.

[Odell *et al.* 94] J. J. Odell, V. Valtchev, P. C. Woodland, S. J. Young, "A One-Pass Decoder Design for Large Vocabulary Recognition", *Proc. ARPA Spoken Language Technology Workshop*, Plainsboro, NJ, März 1994, pp. 405-410.

[Odell 95] J. J. Odell, *The Use of Context in Large Vocabulary Speech Recognition*, Ph.D. Thesis, Cambridge University, Cambridge, März 1995.

[Ortmanns & Ney 95] S. Ortmanns, H. Ney, "Experimental Analysis of the Search Space for 20 000-Word Speech Recognition," *Proc. Europ. Conf. on Speech Communication and Technology*, Madrid, September 1995, pp. 901-904.

[Ortmanns *et al.* 96a] S. Ortmanns, H. Ney, A. Eiden, "Language-Model Look-Ahead for Large Vocabulary Speech Recognition," *Proc. Int. Conf. on Spoken Language Processing*, Philadelphia, PA, Oktober 1996, pp. 2095-2098.

[Ortmanns *et al.* 96b] S. Ortmanns, H. Ney, A. Eiden, N. Coenen, "Look-Ahead Techniques for Improved Beam Search," *Proc. CRIM-FORWISS Workshop*, Montreal, Oktober 1996, pp. 10-22.

[Ortmanns & Aubert 97] S. Ortmanns, X. Aubert, "A Word Graph Algorithm for Large Vocabulary Continuous Speech Recognition," *Computer Speech and Language*, Vol. 11, No. 1, pp. 43-72, Januar 1997.

[Ortmanns 98] S. Ortmanns, *Effiziente Suchverfahren zur Erkennung kontinuierlich gesprochener Sprache*, Doktorarbeit, Fachgruppe Informatik, RWTH Aachen, Aachen, 1998.

[Paul 89] D. Paul, "The Lincoln Robust Continuous Speech Recognizer," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Glasgow, Schottland, Großbritannien, 1989, pp. 449-452.

[Paul 97] D. Paul, "Extensions to Phone-State Decision-Tree Clustering: Single Tree and Tagged Clustering," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, München, Deutschland, April 1997, pp. 1487-1490.

[Press *et al.* 86] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, *Numerical Recipes*, Cambridge University Press, Cambridge, 1986.

[Price *et al.* 88] P.J. Price, W. Fisher, J. Bernstein, D. Pallett, "A Database for Continuous Speech Recognition in a 1000-Word Domain," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, New York, NY, April 1988, pp. 651-654.

[Quinlan 86] J.R. Quinlan, "Induction of Decision Trees," *Machine Learning*, Kluwer Academic Publishers, Boston, Vol. 1, pp. 1-86, 1986.

[Rabiner & Juang 93] L. Rabiner, B.-H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, Englewood Cliffs, NJ, 1993.

[Takami & Sagayama 92] J. Takami, S. Sagayama, "A Successive State Splitting Algorithm for Efficient Allophone Modeling," *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, San Francisco, CA, März 1992, pp. 573-576.

[Schwartz *et al.* 85] R. Schwartz, Y. Chow, O. Kimball, S. Roucos, U. Krasner, J. Makhoul, "Context Dependent Modeling for Acoustic-Phonetic Recognition of Continuous Speech," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Tampa, FL, März/April 1985, pp. 1205-1208.

[Schwartz *et al.* 91] R. Schwartz, S. Austin, "A Comparison of Several Approximate Algorithms for Finding Multiple (N-Best) Sentence Hypotheses," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Toronto, Kanada, September 1991, pp. 701-704.

[Schwartz *et al.* 92] R. Schwartz, S. Austin, F. Kubala, J. Makhoul, L. Nguyen, P. Placeway, G. Zavaliagkos, "New Uses for the N-Best Sentence Hypotheses within the BYBLOS Speech Recognition System," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, San Francisco, CA, März 1992, pp. 1-4.

[Schukat-Talamazzini *et al.* 92] E.G. Schukat-Talamazzini, H. Niemann, W. Eckert, T. Kuhn, S. Rieck, "Acoustic Modelling of Subword Units in the ISADORA Speech Recognizer," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, San Francisco, CA, März 1992, pp. 577-580.

[Schukat-Talamazzini 95] E.G. Schukat-Talamazzini, *Automatische Spracherkennung*, Vieweg, Braunschweig, 1995.

[Steinbiss *et al.* 93] V. Steinbiss, H. Ney, R. Haeb-Umbach, B.-H. Tran, U. Essen, R. Kneser, M. Oerder, H.G. Meier, X. Aubert, C. Dugast, D. Geller, "The Philips Research System for Large-Vocabulary Continuous-Speech Recognition," *Proc. Europ. Conf. on Speech Communication and Technology*, Berlin, September 1993, pp. 2125-2128.

[Steinbiss *et al.* 94] V. Steinbiss, B.-H. Tran, H. Ney," Improvements in Beam Search," Proc. *Int. Conf. on Spoken Language Processing*, Yokohama, Japan, September 1994, pp. 2143-2146.

[Verbmobil Web Site] *Verbmobil*, DFKI, Kaiserslautern, http://www.dfki.de/verbmobil/, Februar 1999.

[Weintraub *et al.* 89] M. Weintraub, H. Murveit, M. Cohen, P. Price, J. Bernstein, G. Baldwin, D. Bell, "Linguistic Constraints in Hidden Markov Model Based Speech Recognition," *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Glasgow, Schottland, 1989.

[Welling *et al.* 97] L. Welling, N. Haberland, H. Ney, "Acoustic Front-End Optimization for Large Vocabulary Speech Recognition," *Proc. European Conference on Speech Communication and Technology*, Rhodos, Griechenland, September 1997, pp. 2099-2102.

[Welling 98] L. Welling, *Merkmalsextraktion in Spracherkennungssystemen für großen Wortschatz*, Doktorarbeit, Fachgruppe Informatik, RWTH Aachen, Aachen, 1998.

[Young & Woodland 93] S.J. Young, P.C. Woodland, "The Use of State Tying in Continuous Speech Recognition," *Proc. Europ. Conf. on Speech Communication and Technology*, Berlin, Deutschland, September 1993, pp. 2203-2206.

[Young *et al.* 94] S.J. Young, J.J. Odell, P.C. Woodland, "Tree-Based State Tying for High Accuracy Acoustic Modelling," *Proc. ARPA Human Language Technology Workshop*, Plainsboro, NJ, Morgan Kaufmann, März 1994, pp. 405-410.

Lebenslauf

Persönliche Daten

Geburtstag:	3. März 1968
Geburtsort:	Rheydt

Schulbesuch

1974 bis 1978	Grundschule Jüchen
1978 bis 1987	Gymnasium Odenkirchen
1987	Abitur

Studium

1987 bis 1993	Studium der Informatik an der RWTH Aachen
1992 bis 1993	Praktikum und anschließende Diplomarbeit am IBM Entwicklungszentrum in Heidelberg
1993	Diplom

Promotion

1993 bis 1999	Wissenschaftlicher Angestellter im Bereich Spracherkennung am Lehrstuhl für Informatik VI der RWTH Aachen
1999	Promotion