

Maschinelles Lernen von Aspekten des deutschen Wortakzents

Stefan Rapp

*Universität Stuttgart, Institut für Maschinelle Sprachverarbeitung,
Lehrstuhl für Experimentelle Phonetik*

Alle Rechte vorbehalten / All rights reserved

Zusammenfassung

Thema dieser Arbeit ist die Anwendung von Techniken des Maschinellen Lernens auf phonetische und phonologische Fragestellungen.

Im phonetischen Teil der Arbeit werden die Korrelate der Betonung untersucht. Ein Sprecher kann den Wortakzent durch verschiedene phonetische Ausdrucksmittel realisieren, nämlich durch Grundfrequenzänderung, Änderung der Lautstärke und Änderung der Länge der betonten Silbe. Ziel der Untersuchungen ist festzustellen, welche dieser Ausdrucksmittel die zuverlässigste Vorhersage des Wortakzents erlauben. Für diese Untersuchungen stand eine Datenbasis mit Meßwerten von zweihundert Silben zur Verfügung, die in einem Experiment von Versuchspersonen produziert wurden. Die Untersuchungen haben unter anderem ergeben, daß die Länge eindeutig am besten geeignet ist, die betonten von den unbetonten Silben zu unterscheiden.

Der phonologische Teil der Arbeit befaßt sich mit der Vorhersage des Wortakzents im Deutschen. Die in der Phonologie verbreitete These, daß der Hauptakzent eines morphologisch einfachen Wortes aus seiner Silbenstruktur und der segmentalen Anordnung seiner Silben vorhergesagt werden kann, wurde von den Lernreihen bestätigt. In der phonologischen Forschung sind die Regeln zur Vorhersage des Wortakzents meist deduktiv bestimmt worden. Die Verfahren des Maschinellen Lernens bieten hier eine Möglichkeit, diese deduktiv gewonnenen Annahmen empirisch zu überprüfen. Ein Großteil der in der phonologischen Literatur verbreiteten Annahmen fand eine Entsprechung in den gelernten Regeln. Außerdem wurden verschiedene Codierungen hinsichtlich der erzielbaren Erkennungsraten untersucht. Die Ergebnisse rechtfertigen die in der Phonologie übliche Beschränkung auf Silbentypen, da eine vergleichbare Codierung in Einzelmerkmale keine besseren Resultate erzielt. Wenn man zusätzlich Längensinformation als Beschreibungsmerkmal der Silbenstruktur akzeptiert, ergeben sich die besten Erkennungsraten von etwa 80%. Dies ist ein ermutigendes Ergebnis, da das Korpus viele Wörter enthält, die von den meisten phonologischen Theorien ebenfalls als Ausnahmen betrachtet werden. Aufgrund der Ergebnisse des phonetischen Teils der Arbeit ist eine solche Codierung für die phonologische Forschung methodologisch problematisch. Andererseits ist diese Codierung mit dem Längenmerkmal für die praktische Anwendung am besten geeignet, da die Regeln für diese Codierung mit Abstand am einfachsten aufgebaut sind und die besten Ergebnisse liefern.

Inhaltsverzeichnis

1	Einleitung	153
2	Rough Sets	157
2.1	Rough Sets Theorie	157
2.1.1	Rough Sets	157
2.1.2	Beurteilung von Approximationen	159
2.1.3	Beziehungen zwischen Mengen	160
2.1.4	Sätze	160
2.1.5	Redukt und Kern	162
2.2	Variable Precision Rough Sets Model	166
2.2.1	Unterschiede zur klassischen Rough Sets Theorie	166
2.2.2	Redukte im VP-Modell	168
2.3	Rough Sets Theorie und Maschinelles Lernen	169
2.3.1	Unterstützte Merkmalstypen	169
2.3.2	Generalisierung	170
3	Akustische Korrelate der Betonung des Deutschen	173
3.1	Untersuchungsgegenstand	173
3.2	Datenmaterial und Datenaufbereitung	174
3.2.1	Gewinnung der Daten	174
3.2.2	Merkmalsextraktion	175
3.3	Lernreihen und Ergebnisse	176
3.3.1	Erste Lernreihe: Alle Merkmale	177
3.3.2	Zweite Lernreihe: Ohne Zeitinformation	178
3.3.3	Dritte Lernreihe: nur Lautstärkemerkmale	181
3.3.4	Vierte Lernreihe: Grundfrequenzmerkmale und Änderung der Lautstärke	182
3.3.5	Fünfte Lernreihe: Grundfrequenzmerkmale und Lautstärke	184
3.3.6	Sechste Lernreihe: nur Grundfrequenzmerkmale	185
3.3.7	Siebte Lernreihe: nur Änderung der Lautstärke	185
3.3.8	Achte Lernreihe: nur Lautstärke	186
3.4	Zwischenbilanz	186
3.5	Pitchakzent und Differenzmerkmale	188
3.6	Zusammenfassung und Bewertung	192

4	Phonologische Betonungsregeln des Deutschen	197
4.1	Untersuchungsgebiet	197
4.2	Datenmaterial und -aufbereitung	198
4.2.1	Auswahl eines Korpus	198
4.2.2	Transkription der Wörter	199
4.2.3	Umsetzung in Merkmale	200
4.3	Lernreihen	201
4.3.1	Erste Lernreihe	203
4.3.2	Zweite Lernreihe	205
4.3.3	Dritte Lernreihe	208
4.4	Regeln	209
4.4.1	Erste Codierung: nur Typinformation	209
4.4.2	Zweite Codierung: Einzelmerkmale	212
4.4.3	Dritte Codierung: Einzelmerkmale ohne Länge	213
4.5	Beziehungen zu verbreiteten phonologischen Regeln	219
4.6	Zusammenfassung und Bewertung	224
5	Abschließende Bemerkungen und Ausblick	229
A	Durchschnittliche Silbenanzahl im Deutschen	233
B	Literatur	237

Kapitel 1

Einleitung

Die Prosodie ist ein interessantes Teilgebiet der Linguistik, deren wichtigste Themenkomplexe die Untersuchung von Betonung, Satzmodusmarkierung und Gliederung von Äußerungen sind. Wichtige prosodische Eigenschaften sind Tonhöhe, Lautheit, zeitliche Strukturierung, Sprechtempo, Stimmlage, Stimmqualität, Klangfarbe, Rhythmus und auch das Fehlen eines sprachlichen Ereignisses, die Pause [Nöth91]. Meist werden in der Prosodie suprasegmentale Merkmale betrachtet, denn in der Regel ist es nicht möglich, die prosodischen Eigenschaften nur einem einzelnen Laut zuzuordnen. Die sprachlichen Einheiten, über die sich eine Untersuchung erstrecken kann, sind nicht eindeutig festgelegt, man betrachtet Silben, Wörter, Phrasen, Sätze oder ganze Redebeiträge.

Betonung kann aus zumindest drei Blickwinkeln analysiert werden. Man unterscheidet Satzakzent, Phrasenakzent und Wortakzent. In dieser Arbeit werden zwei Aspekte des Wortakzents untersucht, die man vereinfachend wie folgt formulieren kann:

1. Wodurch unterscheiden sich betonte von unbetonten Silben?
2. Welche Silbe eines Wortes wird betont?

Bei der ersten Fragestellung müssen meßbare Größen wie Tonhöhe, Lautstärke und zeitliche Strukturierung daraufhin untersucht werden, ob sie für betonte Silben andere Werte annehmen als für unbetonte. Es handelt sich dabei also um eine Untersuchung der Betonung aus phonetischer Sicht. Bei der zweiten Fragestellung wird Betonung aus phonologischer Sicht untersucht. Es wird versucht herauszufinden, warum gerade die Silbe *a* und nicht die Silbe *b* in einem Wort *X* betont ist. Hier interessieren Fragen wie: Kann man aus der Struktur der Silben ableiten, auf welche Silbe die Betonung fällt? Wenn es sie gibt, wie sehen diese Regeln aus?

Die Prosodie weist Verbindungen zu fast allen anderen Teilgebieten der Linguistik auf. Literatur über prosodische Untersuchungen stammt deshalb von Autoren sehr unterschiedlicher Schulen. Dies hat zur Folge, daß Begriffe wie Betonung und Akzent sowie daraus abgeleitete Formen wie zum Beispiel be-

tont / unbetont, Akzentuierung, Satz- und Wortakzent, nicht einheitlich verwendet werden. Für diese Arbeit können die Begriffe Betonung und Akzent als austauschbar betrachtet werden, eine Ansicht, die auch in [Nöth91] vertreten wird und die auch in Übereinstimmung mit [Bußm83] ist. Die dort erwähnten weiteren Bedeutungen von Akzent, nämlich die Verwendung im Sinne von Fremdwortakzent und die Verwendung als Bezeichnung für einen Bestandteil der Orthographie, sind hier nicht relevant.

Thema des phonetischen Teils dieser Arbeit ist, aus gemessenen Daten die Unterschiede zwischen als betont und unbetont wahrgenommenen Silben zu entdecken. Daher ist es gerechtfertigt, Wortakzent als ein Merkmal der Silbe aufzufassen, die innerhalb des Wortes die größte Prominenz aufweist. Dabei kann der Wortakzent jedoch nicht ganz unabhängig vom Satzakzent untersucht werden, der hier als ein Merkmal der Silbe verstanden wird, die im Satz die größte Prominenz aufweist. Prominenz ist hier bewußt nicht näher definiert, um nicht dadurch schon begrifflich festzulegen, wie im Deutschen Wortakzent und möglicherweise Satzakzent realisiert werden.

Weiterhin werden hier die Begriffe „fokussiert“ und „trägt den Satzakzent“ ebenfalls synonym verwendet. Dies steht im Gegensatz zur Terminologie mancher Autoren, die durch den Begriff Satzakzent die Stelle kennzeichnen, die normalerweise, das heißt losgelöst von einem bestimmten Kontext, die Betonung erhalten würde. Wenn durch den Kontext eine andere Stelle größere Prominenz erhält, spricht man dann von Fokusakzent, emphatischem Akzent oder kontrastivem Akzent. Eine solche Unterscheidung wird hier nicht vorgenommen, da emphatische und kontrastive Akzente in diesem Zusammenhang nicht berücksichtigt werden.

Betonung wird zumeist nicht als binäres Merkmal angesehen. Silben können nicht nur betont oder unbetont sein, sondern vielmehr verschiedene Grade von Betonung annehmen [Fudg84, ChomHall68]. In [ChomHall68] werden zum Beispiel theoretisch unbegrenzt viele Stufen unterschieden. Eine solch feine Unterteilung kann sicherlich vorgenommen und möglicherweise auch aus dem Signal gerechtfertigt werden. Allerdings ist fraglich, ob eine so feine Einteilung auch interpretierbar ist, ob also Hörer diese Einteilung in der Interpretation des Geäußerten unterscheiden können ([Lieb65] seq. [Nöth91]). Diese interessante Frage wird in dieser Arbeit nicht behandelt und es wird auch nicht in verschiedene Abstufungen der Betonung unterschieden. Vielmehr wird, möglicherweise in Verkennung der kontinuierlichen Natur der Betonung, stets nur zwischen betont und unbetont unterschieden. Für den phonetischen Teil ist auch die Bereitstellung von Lerndaten nur auf dieser binären Basis machbar und objektiv vorzunehmen, da sonst umfangreiche Wahrnehmungsexperimente hätten durchgeführt werden müssen. Für den phonologischen Teil ist die binäre Betrachtungsweise ebenfalls begründbar. Das für die Transkription herangezogene Aussprachewörterbuch verzeichnet in sämtlichen in das Korpus aufgenommene Einträgen lediglich einen Primärakzent. Die Notwendigkeit für die Einführung feinerer Abstufungen scheint erst bei der Betrachtung zusammengesetzter Wörter notwendig zu werden.

Neben dem rein linguistischen Interesse, den deutschen Wortakzent besser zu verstehen, ist ein besseres Verstehen der Betonung auch für andere Bereiche wichtig. Prosodische Information kann dazu benutzt werden, die Erkennung kontinuierlich gesprochener Sprache zu verbessern [Nöth91]. Zum Beispiel kann durch den Vergleich von Betonungsmustern zwischen Gemessenem und Erwartetem eine Reduzierung der Kandidatenzahl bei den Wort- oder Satzhypothesen bei der automatischen Spracherkennung erzielt werden. Dazu muß man einerseits Akzente im Signal erkennen können und andererseits wissen, an welcher Stelle ein Akzent zu erwarten ist. Noch direkter lassen sich Erkenntnisse über den Wortakzent bei der Generierung von Sprache einsetzen. Auch hier muß man wissen, wie und wo Akzente im Deutschen realisiert werden, damit die erzeugte Sprache natürlich und verständlich wird. Schließlich ist man bei der Umsetzung von unaufbereitetem Text in Sprache (*text-to-speech*) — wie bei der Spracherkennung — mit dem Problem konfrontiert, auch unbekannte, nicht aus einem Lexikon abrufbare Wörter adäquat zu behandeln.

Als Instrument für die Untersuchungen dient in dieser Arbeit ein Verfahren des Maschinellen Lernens. Dieses Vorgehen bietet die Möglichkeit, Regeln, die in der linguistischen Forschung häufig deduktiv gewonnen werden, mit empirischen Mitteln zu überprüfen. Der gewählte Ansatz ist das Lernen aus Beispielen, das wie folgt charakterisiert werden kann:

Gegeben ist eine Menge von Beispielen U . Jede Teilmenge C von U kann als zu lernendes *Konzept* aufgefaßt werden. Ein Beispiel $e^+ \in C$ ist ein *positives Beispiel* für C , ein Beispiel $e^- \in U \setminus C$ ist ein *negatives Beispiel* für C . Die Beispiele sind üblicherweise durch Merkmalsvektoren beschrieben. Die Zugehörigkeit eines Beispiels zu einem Konzept wird durch den Wert eines Merkmals d in der Beschreibung des Beispiels festgelegt. d wird als Entscheidungsvariable oder Entscheidungsattribut bezeichnet. Jeder Wert w des Entscheidungsattributs d beschreibt genau ein Konzept C . Die Aufgabe des Lernens aus Beispielen besteht darin, eine *Beschreibung* des Konzeptes C zu lernen.

Man unterscheidet zwischen symbolischen und nichtsymbolischen Lernverfahren. Zu den nichtsymbolischen Lernverfahren gehören beispielsweise die Verfahren des Lernens mit künstlichen neuronalen Netzen. Der Nachteil nichtsymbolischer Verfahren liegt darin, daß die erzeugten Konzeptbeschreibungen für Menschen kaum zu verstehen sind. Beispielsweise besteht die Beschreibung D bei künstlichen neuronalen Netzen aus den Gewichten, die mit den Neuronen und ihren Eingängen assoziiert sind. Eine solche Konzeptbeschreibung kann effizient von Maschinen angewendet werden, dagegen ist die Beschreibung für Menschen nur sehr schwer nachvollziehbar.

Die symbolischen Lernverfahren erzeugen symbolische Beschreibungen eines Konzeptes C , zum Beispiel eine Menge von Regeln oder einen Entscheidungsbaum. Solche Beschreibungen sind für Menschen meist leichter verständlich und liefern auch inhaltliche Aufschlüsse über die gelernten Konzepte. Zugleich können auch diese Konzeptbeschreibungen von anderen Programmen genutzt werden, beispielsweise von Expertensystemen, Regelungssystemen zur Maschinensteuerung etc.

Überblick über die Arbeit

Im nächsten Kapitel der Arbeit wird die Rough Set Theorie vorgestellt. Die Rough Sets Theorie liefert eine klar definierte mathematische Theorie für das Lernen aus Beispielen, insbesondere für das Lernen mit Attribut–Wert–Repräsentationen. Sie bildet die Grundlage des in dieser Arbeit benutzten symbolischen Lernverfahrens, das ebenfalls in diesem Kapitel kurz erklärt wird. Anschließend folgen zwei Kapitel, in denen das Lernverfahren auf phonetische und phonologische Daten angewandt wird.

Im dritten Kapitel, dem phonetischen Teil der Arbeit, wird das Lernverfahren eingesetzt, um Unterschiede zwischen betonten und unbetonten Silben herauszufinden. Als Datenmaterial werden dazu silbenbezogene Meßwerte von Äußerungen verwendet, die an der Universität Bielefeld erhoben wurden. Die Lernaufgabe besteht darin, Regeln zu erzeugen, mit denen eine durch ihre Merkmale beschriebene Silbe als betont oder unbetont klassifiziert werden kann. Am Ende des Kapitels werden die erzielten Ergebnisse zusammengefaßt.

Das vierte Kapitel, der phonologische Teil der Arbeit, behandelt die Frage, auf welche Silbe eines morphologisch einfachen Wortes normalerweise, das heißt unabhängig von einem Kontext, die Wortbetonung fällt. Als Merkmale stehen dem Lernalgorithmus symbolische Werte zur Verfügung, die die Struktur der Silben beschreiben. Die Lernaufgabe ist hier, die richtige Stelle der Betonung vorherzusagen. Auch dieses Kapitel schließt mit einer Zusammenfassung und Bewertung der erzielten Ergebnisse.

Das letzte Kapitel enthält einen Ausblick auf weitere interessant erscheinende Fragestellungen, die sich aus dieser Arbeit ergeben.

Im Anhang findet sich eine Abschätzung der durchschnittlichen Silbenanzahl in deutschen Wörtern. Die Arbeit schließt mit einem Verzeichnis der verwendeten Literatur.

Kapitel 2

Rough Sets

2.1 Rough Sets Theorie

Die Theorie der Rough Sets wurde von Zdzisław Pawlak entwickelt als Erweiterung der klassischen Mengenlehre auf Basis der dreiwertigen Logik [Paw82]. Sie liefert eine mathematische Theorie, in der auch unsichere Information, nur vage beschriebene Konzepte etc. behandelt werden können. Dieses Kapitel bietet eine kurze Einführung in die grundlegenden Konzepte der Theorie, die im wesentlichen einer Monographie über die Rough Sets Theorie von Pawlak [Paw91] folgt.

Wie die klassische Mengenlehre nimmt auch die Rough Sets Theorie eine Grundmenge von Individuen an, die Objekte x_i eines Universums U . Im Unterschied zur klassischen Mengenlehre gibt es in der Rough Sets Theorie jedoch eine Äquivalenzrelation R , die sogenannte Ununterscheidbarkeitsrelation (engl. *indiscernibility relation*),¹ die das Universum in Äquivalenzklassen zerlegt. Die Objekte innerhalb einer Äquivalenzklasse sind voneinander nicht unterscheidbar. Diese Äquivalenzklassen werden auch als Elementarmengen (engl. *elementary sets*) bezeichnet. Die Einführung einer solchen Ununterscheidbarkeitsrelation ist ein recht natürlicher Schritt: Auch wir können, wenn wir unsere Umwelt erfassen, nicht die gesamte Information eines Objektes aufnehmen, sondern beschränken uns auf wenige Merkmale oder Meßwerte. Zwei verschiedene Objekte erscheinen uns gleich, wenn sie in allen Merkmalen übereinstimmen, denen wir Beachtung schenken.

2.1.1 Rough Sets

Die Einführung der Ununterscheidbarkeitsrelation hat eine entscheidende Auswirkung auf das zentrale Konzept der Mengenlehre, die Menge. Manche Mengen

¹Dies ist eine aus didaktischen Gründen vorgenommene Vereinfachung. Genau genommen fügt man der klassischen Mengenlehre eine ganze Familie \mathbf{R} von Relationen R_i zu. Eine exakte algebraische Definition der Ununterscheidbarkeitsrelation findet der mathematisch interessierte Leser in [Paw91, Seite 3].

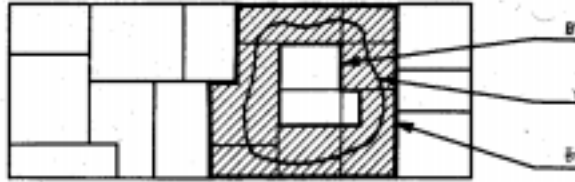


Abbildung 2.1: Approximation einer Menge Y durch B -exakte Mengen (entnommen [Wong et al.86])

X können nun nicht mehr exakt beschrieben werden, nämlich wenn ein Element einer bestimmten Äquivalenzklasse in der Menge X und ein anderes Element derselben Äquivalenzklasse außerhalb von X liegt. Eine solche Menge X , deren Grenze durch eine Äquivalenzklasse verläuft, kann mit der durch die Äquivalenzrelation R gegebenen Information nicht genau definiert werden. Sie heißt dann R -undefinierbar, R -inexakt oder R -rauh. Die Teilmengen des Universums, die eine Vereinigung von Elementarmengen darstellen, deren Grenze also entlang der Grenzen der Äquivalenzklassen verläuft, werden als R -definierbar oder R -exakt bezeichnet.

Jede beliebige Menge X von Objekten des Universums U kann durch zwei R -exakte Mengen, die R -untere Approximation und die R -obere Approximation, angenähert werden. Die R -untere Approximation $\underline{R}X$ und die R -obere Approximation $\overline{R}X$ einer Menge X sind definiert wie folgt:

$$\underline{R}X = \bigcup \{Y \in U/R \mid Y \subseteq X\}$$

$$\overline{R}X = \bigcup \{Y \in U/R \mid Y \cap X \neq \emptyset\}$$

Die untere Approximation $\underline{R}X$ ist also die Vereinigung aller Elementarmengen, die vollständig in der Menge X enthalten sind. $\overline{R}X$ ist die Vereinigung aller Elementarmengen, die mindestens ein Element mit X gemeinsam haben. $\underline{R}X$ enthält also Objekte, die sicher in X liegen, die Elemente von $\overline{R}X$ liegen möglicherweise in X . Abbildung 2.1 veranschaulicht die beiden Konzepte.

Manchmal wird anstelle $\underline{R}X$ und $\overline{R}X$ ein weiterer Satz von Bezeichnungen verwendet. Als $BN_R(X)$ wird die Vereinigung der Elementarmengen bezeichnet, die von der Grenze von X zerschnitten werden. In Abbildung 2.1 ist dies das schraffierte Gebiet. $POS_R(X)$ ist die Vereinigung der Äquivalenzklassen, die vollständig innerhalb von X liegen. In Abbildung 2.1 ist dies das innere nicht schraffierte Gebiet. Die Vereinigung der Äquivalenzklassen, die vollständig außerhalb von X liegen, wird mit $NEG_R(X)$ bezeichnet. In Abbildung 2.1 ist dies das äußere nicht schraffierte Gebiet. Man kann diese drei Begriffe durch die Begriffe $\underline{R}X$ und $\overline{R}X$ wie folgt definieren:

$$\begin{aligned} POS_R(X) &= \underline{R}X \\ BN_R(X) &= \overline{R}X - \underline{R}X \\ NEG_R(X) &= U - \overline{R}X \end{aligned}$$

2.1.2 Beurteilung von Approximationen

Verschiedene Mengen können durch ihre oberen und unteren Approximationen unterschiedlich gut angenähert werden. Für die Güte der Approximierbarkeit einer Menge gibt es zwei verschiedene Beurteilungen: eine topologisch-algebraische und eine numerische Beurteilung.

Topologisch-algebraisches Beurteilung

Die topologisch-algebraische Beurteilung weist einer Menge X eine von vier „Güteklassen“ zu:

- X heißt *rauh R -definierbar*, wenn $\underline{R}X \neq \emptyset$ und $\overline{R}X \neq U$, wenn also $POS_R(X)$ und $NEG_R(X)$ nichtleer sind.
- X heißt *intern R -undefinierbar*, wenn $\underline{R}X = \emptyset$ und $\overline{R}X \neq U$, wenn also $POS_R(X)$ leer und $NEG_R(X)$ nichtleer ist. Es gibt also keine Äquivalenzklasse, die vollständig innerhalb von X liegt, das Grenzgebiet $BN_R(X)$ erstreckt sich „bis ganz innen“.
- X heißt *extern R -undefinierbar*, wenn $\underline{R}X \neq \emptyset$ und $\overline{R}X = U$, also wenn $POS_R(X)$ nichtleer und $NEG_R(X)$ leer ist. Keine Äquivalenzklasse liegt komplett außerhalb von X , das Grenzgebiet $BN_R(X)$ erstreckt sich nach außen „bis an den Rand des Universums“.
- X heißt *total R -undefinierbar*, wenn $\underline{R}X = \emptyset$ und $\overline{R}X = U$, wenn also $POS_R(X)$ und $NEG_R(X)$ nichtleer sind. In diesem Fall gilt $BN_R(X) = U$, das heißt, in jeder Elementarmenge gibt es Elemente, die zu X gehören, und Elemente, die nicht zu X gehören.

Numerische Beurteilung

Für die Approximierbarkeit von Rough Sets gibt es zwei numerische Maße, die *Akkuratheit* und die *Roughness*. Für praktische Anwendungen sind diese numerischen Maße besser geeignet als die oben beschriebenen topologischen Beurteilungen.

Die Akkuratheit $\alpha_R(X)$ einer Menge X beschreibt die Vollständigkeit des Wissens über die Menge X . Sie gibt an, wie gut X mit den von R induzierten Äquivalenzklassen beschrieben werden kann. Die Akkuratheit $\alpha_R(X)$ von X ist folgendermaßen definiert:

$$\alpha_R(X) = \frac{\text{card}(\underline{R}X)}{\text{card}(\overline{R}X)}$$

Da die Beziehung $\underline{R}X \subset \overline{R}X$ gilt, kann $\alpha_R(X)$ Werte zwischen 0 und 1 annehmen. Wenn die Menge X R -exakt, also durch Elementarmengen genau beschreibbar ist, so ist $\underline{R}X = \overline{R}X$ und damit $\alpha_R(X) = 1$. Wenn das Grenzgebiet

$BN_R(X)$ wächst, sinkt das Akkuratheitsmaß $\alpha_R(X)$, bis es schließlich bei Verschwinden der positiven Region $POS_R(X)$ den Wert 0 annimmt.

Die Roughness einer Menge $\rho_R(X)$ wird über ihre Akkuratheit $\alpha_R(X)$ definiert. Sie stellt gewissermaßen das Gegenteil der Akkuratheit dar:

$$\rho_R(X) = 1 - \alpha_R(X)$$

Zwischen den topologischen und numerischen Beurteilungen von Mengen bestehen Beziehungen, beispielsweise hat jede intern R -undefinierbare Menge X die Akkuratheit $\alpha_R(X) = 0$ und die Roughness $\rho_R(X) = 1$.

Der Begriff der Akkuratheit und Roughness läßt sich auch auf Mengen von Mengen ausdehnen. Es sei $F = \{X_1, X_2, \dots, X_n\}$ eine Familie nichtleerer Mengen. Dann sind die untere Approximation $\underline{R}F$ und die obere Approximation $\overline{R}F$ von F definiert wie folgt:

$$\begin{aligned}\underline{R}F &= \{\underline{R}X_1, \underline{R}X_2, \dots, \underline{R}X_n\} \\ \overline{R}F &= \{\overline{R}X_1, \overline{R}X_2, \dots, \overline{R}X_n\}\end{aligned}$$

Die *Akkuratheit der Approximation von F durch R* ist definiert durch:

$$\alpha_R(F) = \frac{\sum \text{card}(\underline{R}X_i)}{\sum \text{card}(\overline{R}X_i)}.$$

Besonders interessante Familien von Mengen sind hier die Klassifikationen. Eine Klassifikation ist eine Menge paarweise disjunkter Teilmengen des Universums, die das ganze Universum abdecken. Klassifikationen sind also auch Partitionen. Die einzelnen Teilmengen werden als *Konzepte* bezeichnet. Für Klassifikationen existiert ein weiteres Maß $\gamma_R(F)$, die *Qualität der Approximation von F durch R* . Sie ist wie folgt definiert:

$$\gamma_R(F) = \frac{\sum \text{card}(\underline{R}X_i)}{\text{card}(U)}.$$

Die Qualität $\gamma_R(F)$ einer Klassifikation gibt an, wieviel Prozent der Objekte korrekt klassifiziert werden können.

2.1.3 Beziehungen zwischen Mengen

2.1.4 Sätze

In der Rough Set Theorie gelten die folgenden aus den Definitionen ableitbaren Sätze:

- (1) $\underline{R}X \subseteq X \subseteq \overline{R}X$
- (2) $\underline{R}\emptyset = \overline{R}\emptyset = \emptyset; \quad \underline{R}U = \overline{R}U = U$
- (3) $\overline{R}(X \cup Y) = \overline{R}X \cup \overline{R}Y$
- (4) $\underline{R}(X \cap Y) = \underline{R}X \cap \underline{R}Y$
- (5) $X \subseteq Y \quad \text{impliziert} \quad \underline{R}X \subseteq \underline{R}Y$
- (6) $X \subseteq Y \quad \text{impliziert} \quad \overline{R}X \subseteq \overline{R}Y$
- (7) $\underline{R}(X \cup Y) \supseteq \underline{R}X \cup \underline{R}Y$
- (8) $\overline{R}(X \cap Y) \subseteq \overline{R}X \cap \overline{R}Y$
- (9) $\underline{R}(-X) = -\overline{R}X$
- (10) $\overline{R}(-X) = -\underline{R}X$
- (11) $\underline{R}\underline{R}X = \overline{R}\underline{R}X = \underline{R}X$
- (12) $\overline{R}\overline{R}X = \underline{R}\overline{R}X = \overline{R}X$

Die Beweise dazu finden sich in [Pawl91].

Enthaltenseinsrelationen

An die Stelle der Elementbeziehung \in in der klassischen Mengenlehre treten in der Rough Sets Theorie die beiden folgenden Beziehungen:

- (13) $x \underline{\in}_R X$ genau dann, wenn $x \in \underline{R}X$
- (14) $x \overline{\in}_R X$ genau dann, wenn $x \in \overline{R}X$

Die Relation $\underline{\in}_R$ heißt *lower membership* und besagt: „ x gehört sicher zu X bezüglich R “. Die Relation $\overline{\in}_R$ wird als *upper membership* bezeichnet und bedeutet: „ x gehört möglicherweise zu X bezüglich R “.

Es gilt ($\underline{\in}$ und $\overline{\in}$ ohne den Index R):

- (15) $x \underline{\in} X \quad \text{impliziert} \quad x \in X \quad \text{impliziert} \quad x \overline{\in} X$
- (16) $X \subseteq Y \quad \text{impliziert} \quad (x \underline{\in} X \quad \text{impliziert} \quad x \underline{\in} Y \quad \text{und} \quad x \overline{\in} X \quad \text{impliziert} \quad x \overline{\in} Y)$
- (17) $x \overline{\in} (X \cup Y) \quad \text{genau dann, wenn} \quad x \overline{\in} X \quad \text{oder} \quad x \overline{\in} Y$
- (18) $x \underline{\in} (X \cap Y) \quad \text{genau dann, wenn} \quad x \underline{\in} X \quad \text{und} \quad x \underline{\in} Y$
- (19) $x \underline{\in} X \quad \text{oder} \quad x \underline{\in} Y \quad \text{impliziert} \quad x \underline{\in} (X \cup Y)$
- (20) $x \overline{\in} (X \cap Y) \quad \text{impliziert} \quad x \overline{\in} X \quad \text{und} \quad x \overline{\in} Y$
- (21) $x \underline{\in} (-X) \quad \text{genau dann, wenn nicht} \quad x \overline{\in} X$
- (22) $x \overline{\in} (-X) \quad \text{genau dann, wenn nicht} \quad x \underline{\in} X$

Mengengleichheitsrelationen

An die Stelle der Gleichheit von Mengen treten drei Relationen:

- (23) X und Y sind *bottom R -equal*, geschrieben $X \approx_R Y$, genau dann, wenn $\underline{R}X = \underline{R}Y$
- (24) X und Y sind *top R -equal*, geschrieben $X \simeq_R Y$, genau dann, wenn $\overline{R}X = \overline{R}Y$
- (25) X und Y sind *R -equal*, geschrieben $X \approx_R Y$, genau dann, wenn $\underline{R}X = \underline{R}Y$ und $\overline{R}X = \overline{R}Y$

Auch hierzu kann man Sätze angeben; diese sind in [Pawl91] auf Seite 26 zu finden.

Teilmengenbeziehungen

In der Rough Sets Theorie gelten die folgenden Teilmengenbeziehungen:

- (26) X ist *bottom R -included* in Y , geschrieben $X \subsetneq Y$, genau dann, wenn $\underline{R}X \subseteq \underline{R}Y$
- (27) X ist *top R -included* in Y , geschrieben $X \tilde{\subset} Y$, genau dann, wenn $\overline{R}X \subseteq \overline{R}Y$
- (28) X ist *R -included* in Y , geschrieben $X \tilde{\subset} Y$, genau dann, wenn $X \subsetneq Y$ und $X \tilde{\subset} Y$

Beweise hierzu, wie auch einige Sätze, die Zusammenhänge zwischen Teilmengenbeziehungen und Mengengleichheitsbeziehungen aufzeigen, können in [Pawl91] gefunden werden.

2.1.5 Redukt und Kern

In diesem Abschnitt werden die zentralen Begriffe der Rough Set Theorie, Redukt, Kern, relatives Redukt und relativer Kern, mathematisch definiert. Anschließend wird erklärt, was die Begriffe anschaulich bedeuten.

Mathematische Definition

Zur Vereinfachung der Definitionen von Redukt und Kern werden zunächst einige Hilfsbegriffe eingeführt.

Es sei \mathcal{R} eine Familie von Äquivalenzrelationen und $R \in \mathcal{R}$ eine Äquivalenzrelation in \mathcal{R} .

R heißt *entbehrlich* in \mathcal{R} , wenn $IND(\mathcal{R}) = IND(\mathcal{R} \setminus \{R\})$, das heißt, wenn sich die durch \mathcal{R} induzierte Äquivalenzrelation nicht ändert, wenn R aus \mathcal{R} entfernt wird. Andernfalls heißt R *unentbehrlich* in \mathcal{R} .

Die Familie \mathcal{R} ist *unabhängig*, wenn jedes $R \in \mathcal{R}$ unentbehrlich ist in \mathcal{R} . Andernfalls heißt \mathcal{R} *abhängig*.

Außerdem gilt: Wenn \mathcal{R} unabhängig ist und wenn gilt $\mathcal{P} \subseteq \mathcal{R}$, dann ist auch \mathcal{P} unabhängig.

Mit diesen Begriffen können nun Redukt und Kern definiert werden:

Es seien \mathcal{Q} und \mathcal{P} Familien von Äquivalenzrelationen.

$\mathcal{Q} \subseteq \mathcal{P}$ ist ein *Redukt* von \mathcal{P} , wenn \mathcal{Q} unabhängig ist und $IND(\mathcal{Q}) = IND(\mathcal{P})$.

Eine Familie von Äquivalenzrelationen \mathcal{P} kann mehr als ein Redukt besitzen. Die Menge aller unentbehrlichen Relationen aus \mathcal{P} heißt *Kern* und wird $CORE(\mathcal{P})$ geschrieben.

Redukt und Kern sind voneinander abhängig. Es gilt:

$$CORE(\mathcal{P}) = \bigcap RED(\mathcal{P})$$

Dabei ist $RED(\mathcal{P})$ die Familie aller Redukte von \mathcal{P} .

Es folgen nun Hilfsdefinitionen für die Begriffe relatives Redukt und relativer Kern.

Seien P und Q Äquivalenzrelationen über dem Universum U .

Eine *P-positive Region von Q*, geschrieben $POS_P(Q)$, ist die Menge

$$POS_P(Q) = \bigcup_{X \in U/Q} \underline{P}X$$

Die *P-positive Region von Q* ist die Menge aller Objekte des Universums U , die richtig in Klassen von U/Q klassifiziert werden können, allein durch Ausnützen der Information, die durch die Klassifikation U/P gegeben ist.

Analog zu Redukt und Kern werden nun über einige Hilfsdefinitionen die Begriffe relatives Redukt und relativer Kern über einige Hilfsdefinitionen definiert:

Es seien \mathcal{P} und \mathcal{Q} Familien von Äquivalenzrelationen über U .

$R \in \mathcal{P}$ heißt *Q-entbehrlich in P*, wenn gilt:

$$POS_{IND(\mathcal{P})}(IND(Q)) = POS_{IND(\mathcal{P} \setminus \{R\})}(IND(Q))$$

Andernfalls heißt R *Q-unentbehrlich in P*.

Die Familie \mathcal{P} ist *Q-unabhängig*, wenn jedes $R \in \mathcal{P}$ *Q-unentbehrlich* ist in \mathcal{P} ist. Man sagt dann auch: \mathcal{P} ist unabhängig bezüglich \mathcal{Q} . Andernfalls heißt \mathcal{P} *Q-abhängig*.

Damit ist es nun möglich, die Begriffe relatives Redukt und relativer Kern zu definieren:

Die Familie $\mathcal{S} \subseteq \mathcal{P}$ ist ein \mathcal{Q} -Redukt von \mathcal{P} genau dann, wenn \mathcal{S} die \mathcal{Q} -unabhängige Unterfamilie von \mathcal{P} ist und $POS_{\mathcal{S}}(\mathcal{Q}) = POS_{\mathcal{P}}(\mathcal{Q})$. Man sagt auch: \mathcal{S} ist das Redukt von \mathcal{P} relativ zu \mathcal{Q} . Dabei kann \mathcal{P} mehr als ein relatives Redukt besitzen.

Die Menge aller \mathcal{Q} -unentbehrlichen Relationen aus \mathcal{P} heißt \mathcal{Q} -Kern von \mathcal{P} und wird $CORE_{\mathcal{Q}}(\mathcal{P})$ geschrieben.

Auch relatives Redukt und relativer Kern sind voneinander abhängig. Es gilt:

$$CORE_{\mathcal{Q}}(\mathcal{P}) = \bigcap RED_{\mathcal{Q}}(\mathcal{P})$$

Dabei ist $RED_{\mathcal{Q}}(\mathcal{P})$ die Familie aller \mathcal{Q} -Redukte von \mathcal{P} .

Als abschließende Bemerkung sei erwähnt, daß sich für $\mathcal{P} = \mathcal{Q}$ die Definitionen für relatives Redukt und relativer Kern zu den Definitionen von Redukt und Kern vereinfachen.

Beispiel

In diesem Abschnitt werden die eben definierten Begriffe an einem Beispiel erläutert. Das Beispiel ist eine Klassifikationsaufgabe, bei der Objekte x_i , von denen die beobachteten Merkmale M_j bekannt sind, in Klassen c_k einzuteilen sind:

Objekt	M_1	M_2	M_3	K
x_1	1	1	1	c_1
x_2	1	1	2	c_1
x_3	1	2	3	c_2
x_4	1	2	3	c_3
x_5	2	2	4	c_3
x_6	2	2	4	c_3
x_7	3	3	4	c_4
x_8	4	3	5	c_5

Zunächst soll die letzte Spalte K außer acht gelassen und nur die drei Merkmalspalten M_1 , M_2 und M_3 betrachtet werden. Jede Spalte definiert eine Äquivalenzrelation. Wenn nur die Information der Spalte M_1 zur Verfügung steht, sind die Objekte $x_1 \dots x_4$ nicht zu unterscheiden, sondern sind äquivalent. Genauso bilden die Objekte x_5 und x_6 zusammen eine Äquivalenzklasse (oder Elementarmenge) und x_7 sowie x_8 bilden je eine einelementige Elementarmenge. Es bezeichne nun R die Relation, die durch die Spalte M_1 definiert ist; die Relation der Spalte M_2 heiße S und die Relation der Spalte M_3 heiße T . Die drei

Relationen R , S und T bilden zusammen eine Familie von Äquivalenzrelationen, die durch \mathcal{P} bezeichnet wird.

Bei näherer Betrachtung der Beispieltabelle wird sichtbar, daß die mittlere Merkmalsspalte keine neue Information bringt, denn die Unterscheidung zwischen x_2 und x_3 läßt sich mit der rechten Merkmalsspalte erzielen und zur Unterscheidung zwischen x_6 und x_7 reichen die linke und rechte Spalte zusammen aus. Das heißt, daß sich das „ $M_2 = 1$ “, also die Menge $\{x_1, x_2\}$, durch das Konzept „ $M_3 = 1$ oder $M_3 = 2$ “ darstellen läßt. Für die Konzepte „ $M_2 = 2$ “ und „ $M_2 = 3$ “ lassen sich ähnliche, wenngleich auch etwas kompliziertere, von der mittleren Spalte unabhängige Konzepte bilden. Die Tabelle kann also ohne die mittlere Spalte M_2 die gleiche Information ausdrücken wie mit dieser Spalte. Die Spalte M_2 ist also *entbehrlich*. Dies kann man auch ausrechnen: Die von R , S und T induzierte Relation ist dieselbe wie die von R und T allein induzierte Relation:

$$\begin{aligned} IND(\{R, S, T\})/U &= \{\{x_1\}, \{x_2\}, \{x_3, x_4\}, \{x_5, x_6\}, \{x_7\}, \{x_8\}\} \\ &= IND(\{R, T\})/U \end{aligned}$$

Dagegen ist T *unentbehrlich*, da ohne die dritte Spalte nicht mehr zwischen x_1 und x_2 unterschieden werden kann. Damit gehört T zum Kern von \mathcal{P} ; wenn T entfernt wird, vermindert sich die Differenzierungsfähigkeit zwischen den Objekten, das heißt, das Wissen wird geringer. Bei weiterer Betrachtung stellt man fest, daß auch R wie schon S *entbehrlich* ist. Damit folgt dann: $CORE(\mathcal{P}) = \{T\}$.

T allein liefert jedoch nicht die gleiche Information wie alle drei Spalten zusammen, denn ohne T kann zum Beispiel nicht zwischen den Objekten x_6 und x_7 unterschieden werden. Wenn der Kern $\{T\}$ jedoch um eine der beiden Relationen S oder R erweitert wird, ergibt sich die gleiche Information wie mit allen drei Relationen zusammen. Damit sind die beiden (einzigen) Redukte von \mathcal{P} , nämlich $\{R, T\}$ und $\{S, T\}$ gefunden. Das heißt:

$$RED(\mathcal{P}) = \{\{R, T\}, \{S, T\}\}$$

Nun soll auch die Klassifikationsspalte K in die Überlegungen miteinbezogen werden. Auch die Klassifikationsspalte definiert eine Äquivalenzrelation, die üblicherweise als *Klassifikation* bezeichnet wird. Bei einer Klassifikation bezeichnet man die einzelnen Äquivalenzklassen als *Konzepte*. Die Klassifikation wird im folgenden mit Q bezeichnet. Auch der Fall, daß eine zweite Klassifikationsspalte gibt, kann mit der Theorie abgedeckt werden, indem man die verschiedenen Klassifikationen zu einer Familie \mathcal{Q} zusammenfaßt. Im hier behandelten Beispiel gibt es jedoch nur eine Klassifikationsspalte, so daß gilt $\mathcal{Q} = \{Q\}$.

Wenn die Klassifikation der Objekte ignoriert wird, ist nur die Relation T , also die rechte Spalte M_1 der Tabelle *unentbehrlich*. Die Trennfähigkeit von T wurde unbedingt benötigt, um die Objekte x_1 und x_2 zu unterscheiden. Wenn die Klassifikationsspalte jedoch auch berücksichtigt wird, ändert sich dies: Da x_1 und x_2 zu gleichen Klasse c_1 gehören, brauchen sie gar nicht unterscheidbar zu

sein. Weiterhin ist festzustellen, daß hinsichtlich der Klassifikation in Klassen c_i die Relationen R und S dieselbe Information liefern wie alle drei Spalten zusammen. Bezüglich Q ist T also *entbehrlich*, mit anderen Worten, T ist Q -entbehrlich in \mathcal{P} . Bei näherer Betrachtung zeigt sich schließlich, daß auch R und S Q -entbehrlich sind. Somit ist der Q -Kern von \mathcal{P} leer.

Wie bereits festgestellt wurde, reichen je zwei der drei Spalten M_1 , M_2 und M_3 aus, um die Unterscheidungsfähigkeit aller Spalten zu erhalten. Für diese drei Mengen ist also die zweite Reduktbedingung erfüllt. Nun ist noch zu prüfen, ob auch die zweite Reduktbedingung, Unabhängigkeit, erfüllt ist. Betrachtet man die Beispieltabelle daraufhin, so erkennt man, daß keine der drei Spalten allein schon ausreicht, um alle Objekte zu klassifizieren. Die Q -Redukte von \mathcal{P} sind damit $\{R, S\}$, $\{S, T\}$ und $\{R, T\}$.

2.2 Variable Precision Rough Sets Model

Die Rough Sets Theorie hat sich als sehr wertvoll erwiesen bei Klassifikationsaufgaben im Bereich der Medizin (zum Beispiel [Pawl&al86]) und der Prozeßsteuerung (zum Beispiel [Mroz89]). Mitunter kann es jedoch vorkommen, daß die Beschreibungsmöglichkeiten der Rough Sets Theorie für unsichere Informationen zu grob sind. Mit dem *variable precision rough set model* oder kurz *VP-Modell* schlägt Wojciech Ziarko eine Erweiterung des Rough Sets Ansatzes vor, die statistische Phänomene in das Rough Sets Konzept miteinbezieht und damit diese Probleme löst [Ziar93].

2.2.1 Unterschiede zur klassischen Rough Sets Theorie

Das Rough Sets Modell funktioniert dann besonders gut, wenn die positiven Regionen von Mengen $POS_R(X)$, also die Gebiete, die sicher zu einem bestimmten Konzept gehören, groß sind gegenüber den Randgebieten $BN_R(X)$, denn nur für diese inneren Gebiete kann korrekt klassifiziert werden. In praktischen Anwendungsfällen kommt es aber häufig vor, daß nur ein verhältnismäßig kleines Gebiet sicher einem Konzept zuordnen werden kann, weil in großen Gebieten des Merkmalraums Vermischungen verschiedener Konzepte vorliegen. Diese Vermischungen können tatsächlich vorhanden sein, beispielsweise wenn die beobachtete Größe das Ergebnis eines Zufallsexperiments ist, oder erst durch die Wahrnehmung hervorgerufen werden, beispielsweise durch Meßfehler oder falsche Klassifikationen, also durch Fehler in den Trainingsdaten. Wenn nun 99 positive Beispiele für ein Konzept innerhalb einer bestimmten Elementarklasse liegen und nur ein negatives Beispiel außerhalb dieser Elementarklasse, kann mit der klassischen Rough Sets Theorie nur sehr wenig über diese Klasse aussagt werden, da sie nicht im Inneren dieses Konzeptes liegt. Das VP-Modell lockert die Bedingungen etwas und läßt eine gewisse Fehlerrate bei der Klassifikation zu, um dadurch mehr Objekte überhaupt klassifizieren zu können.

Das Kernstück der Erweiterung von Ziarko ist eine abgeänderte Teilmengenbeziehung. Damit wird zugelassen, daß eine Menge, die „fast komplett“ innerhalb

einer anderen Menge liegt, als deren Teilmenge angesehen werden kann.

Zunächst wird der *relative Grad der Mißklassifikation* $c(X, Y)$ für zwei Teilmengen X und Y des Universums U definiert:

$$c(X, Y) = \begin{cases} 1 - \frac{\text{card}(X \cap Y)}{\text{card}(X)} & \text{wenn } \text{card}(X) > 0 \\ 0 & \text{wenn } \text{card}(X) = 0 \end{cases}$$

Damit kann die klassische Teilmengenbeziehung wie folgt ausgedrückt werden:

$$X \subseteq Y \quad \text{genau dann, wenn} \quad c(X, Y) = 0$$

Die natürlichste Art, diese Beziehung zu lockern, ist, für $c(X, Y)$ auch Werte größer 0 zuzulassen. Damit wird eine *Majoritäts-Teilmengen-Beziehung* \subseteq^β , kurz β -Inklusion, wie folgt definiert:

$$X \subseteq^\beta Y \quad \text{genau dann, wenn} \quad c(X, Y) \leq \beta$$

Eine derart gelockerte Teilmengenbeziehung ist natürlich nur dann sinnvoll, wenn β nicht zu groß wird. Ziarko beschränkt den zulässigen Bereich für β auf $0 \leq \beta < 0.5$. Ist β größer als 0.5, kann man nicht mehr sinnvoll von einer Teilmengenbeziehung zwischen X und Y sprechen, denn in diesem Fall liegen mehr Elemente von X außerhalb von Y als innerhalb von Y liegen. Für $\beta = 0$ degeneriert die β -Inklusion zur gewöhnlichen Inklusion, wie sie in der klassischen Mengenlehre verwendet wird.

Nun wird die β -Inklusion auf die Definitionen der Mengenapproximationen angewendet. Diese heißen dann β -untere bzw. β -obere Approximation bezüglich R :

$$\begin{aligned} \underline{R}_\beta X &= \bigcup \{E \in U/R \mid E \subseteq^\beta X\} \\ &= \bigcup \{E \in U/R \mid c(E, X) \leq \beta\} \\ \overline{R}_\beta X &= \bigcup \{E \in U/R \mid c(E, X) < 1 - \beta\} \end{aligned}$$

Die alternativen Beschreibungen der Rough Sets lassen sich ebenfalls direkt angeben:

$$\begin{aligned} POSR_\beta X &= \bigcup \{E \in U/R \mid c(E, X) \leq \beta\} \\ BNR_\beta X &= \bigcup \{E \in U/R \mid \beta < c(E, X) < 1 - \beta\} \\ NEGR_\beta X &= \bigcup \{E \in U/R \mid 1 - \beta \leq c(E, X)\} \end{aligned}$$

Auch diese Definitionen degenerieren für $\beta = 0$ zu den Definitionen der klassischen Rough Sets Theorie. Außerdem gilt für jedes $0 \leq \beta < 0.5$:

$$\begin{aligned}
\underline{R}X &\subseteq \underline{R}_\beta X \\
\overline{R}_\beta X &\subseteq \overline{R}X \\
BNR_\beta X &\subseteq BNRX \\
NEGR_\beta X &\subseteq NEGRX
\end{aligned}$$

Das VP-Modell erbt viele Eigenschaften des Rough Sets Modells, beispielsweise gelten fast alle der auf Seite 160 für die Rough Sets Theorie angegebenen Sätze in etwas abgewandelter Form auch im VP-Modell. Für $0 \leq \beta < 0.5$ gilt:

$$\begin{aligned}
(29) \quad \underline{R}_\beta X &\stackrel{\subseteq}{=} X \quad \text{sowie} \quad \underline{R}_\beta X \subseteq \overline{R}_\beta X \\
(30) \quad \underline{R}_\beta \emptyset &= \overline{R}_\beta \emptyset = \emptyset; \quad \underline{R}_\beta U = \overline{R}_\beta U = U \\
(31) \quad \overline{R}_\beta(X \cup Y) &\supseteq \overline{R}_\beta X \cup \overline{R}_\beta Y \\
(32) \quad \underline{R}_\beta(X \cap Y) &\subseteq \underline{R}_\beta X \cap \underline{R}_\beta Y \\
(33) \quad \underline{R}_\beta(X \cup Y) &\supseteq \underline{R}_\beta X \cup \underline{R}_\beta Y \\
(34) \quad \overline{R}_\beta(X \cap Y) &\subseteq \overline{R}_\beta X \cap \overline{R}_\beta Y \\
(35) \quad \underline{R}_\beta(-X) &= -\overline{R}_\beta X \\
(36) \quad \overline{R}_\beta(-X) &= -\underline{R}_\beta X
\end{aligned}$$

Auch im VP-Modell sind Gütemaße definiert. Eines von diesen wird benötigt, um einen dem Redukt der Rough Sets Theorie entsprechenden Begriff zu definieren. Es entspricht dem Qualitätsmaß der Approximation γ , ist jedoch relativ und mit Berücksichtigung von β definiert:

Eine Menge von Attributen Q ist β -abhängig vom Grad $\gamma(P, Q, \beta)$ von der Attributmenge P , wenn gilt:

$$\gamma(P, Q, \beta) = \text{card}(POS(P, Q, \beta)) / \text{card}(U).$$

Dabei ist $POS(P, Q, \beta)$ die β -positive Region von U/Q in P , definiert als:

$$POS(P, Q, \beta) = \bigcup_{Y \in U/Q} \underline{IND(P)}_\beta Y.$$

2.2.2 Redukte im VP-Modell

Redukte werden im VP-Modell über das Maß $\gamma(P, Q, \beta)$ definiert. Dadurch wird die Definition einfacher als beim klassischen Rough Sets Modell:

Ein β -Redukt einer Menge von Merkmalsattributen P bezüglich einer Menge von Entscheidungsattributen Q ist eine Teilmenge $RED(P, Q, \beta)$ von P , die die beiden folgenden Kriterien erfüllt:

1. $\gamma(P, Q, \beta) = \gamma(RED(P, Q, \beta), Q, \beta)$
2. Kein Attribut kann aus $RED(P, Q, \beta)$ entfernt werden, ohne Kriterium 1 zu verletzen.

Im nächsten Abschnitt wird nun der Zusammenhang dieser Begriffe mit dem Maschinellen Lernen kurz aufgezeigt.

2.3 Rough Sets Theorie und Maschinelles Lernen

Auf den Konzepten der Rough Sets Theorie wurden bereits verschiedene Algorithmen zum maschinellen Lernen aufgebaut [Wong&al86, Epst&al87, GrzySiko88, Kras88, Ziar89a, Ziar89b, Grzy93]. So beschreiben zum Beispiel Wong, Ziarko und Ye ein ID3-ähnliches Verfahren, das in [Wong&al86] mit Quinlans ID3 [Quin86] verglichen wird. Das in dieser Arbeit verwendete System DataLogic/R ist ein kommerziell vertriebenes Produkt, dessen Algorithmen nicht detailliert veröffentlicht sind. Deswegen kann hier nur eine oberflächliche Beschreibung dieses Lernverfahrens gegeben kann.

Nach [Ziar&al93] beruht DataLogic/R auf der probabilistischen Erweiterung der Rough Sets Theorie, dem VP-Modell. Ausgehend von einer klassifizierten Beispielmengende erzeugt es eine Regelmengende, die die Klassifikation noch nicht gesehener Fälle erlaubt. Das System verarbeitet bis zu 10 000 Beispiele, die durch maximal 150 Merkmale beschrieben sein können. Die Klassenzugehörigkeit der Trainingsbeispiele wird in einem weiteren Attribut, dem Entscheidungsattribut, festgehalten. Die erzeugte Regelmengende besteht disjunktiv verknüpften Regeln für jede der mögliche Ausprägung des Entscheidungsattributs, die die Konzepte definieren. Eine einzelne Regel besteht aus einer Konjunktion von disjunktiven Merkmalsausprägungen. Damit ist die Lernkomponente von DataLogic/R als symbolisches Lernverfahren einzustufen, das aussagenlogische Regeln erzeugt.

2.3.1 Unterstützte Merkmalstypen

DataLogic/R kann mit verschiedenen Merkmalstypen umgehen. Der Wertebereich der Merkmale, die die Beispiele beschreiben, kann reellwertig, ganzzahlig oder symbolisch sein. Bei symbolischen Merkmalswerten erfolgt keine Vorverarbeitung. Reellwertige und ganzzahlige Merkmale müssen zunächst diskretisiert werden, da das zugrundeliegende Lernverfahren symbolische Eingaben erwartet. DataLogic/R verwendet dazu eine einfache Strategie. Der überstrichene Wertebereich wird in zehn gleich große Klassen eingeteilt. Diese Einteilung kann jedoch auch manuell geändert werden. Man kann angeben, in wieviele gleichgroße Klassen geteilt werden soll, oder alle Klassengrenzen explizit festlegen. Dadurch kann man DataLogic/R an logarithmische Skalen anpassen oder durch vorverarbeitende Stufen, z. B. Clustering, eine bessere symbolische Repräsentation kontinuierlicher Merkmalswerte erreichen. Versuche haben ergeben, daß DataLogic/R bei numerischen Werten für die Regelbildung die *closing intervall*-Heuristik [Mich83] verwendet. Bei den phonetischen Untersuchungen habe ich

für die numerischen Merkmale keine Anpassung der unterschiedenen Bereiche vorgenommen, da mir die Feinheit ausreichend erscheint. Eine Erhöhung der Klassenzahl macht sich zudem sehr deutlich in der Regelgenerierungszeit bemerkbar. Für den phonologischen Teil liegen die Merkmalswerte bereits in symbolischer Form vor.

2.3.2 Generalisierung

Das Lernen erfolgt bei DataLogic/R in einem zweistufigen Prozeß. Im ersten Schritt werden für jedes Konzept die relevanten Merkmale bestimmt. Darauf folgt die Zusammenfassung der durch die verbleibenden Merkmale induzierten Elementarklassen der einzelnen Konzepte zu größeren Blöcken. So erhält man Regeln, die in zweierlei Hinsicht nicht-redundant sind: Keines der beteiligten Merkmale und keine der Regeln kann entfernt werden, ohne daß die Genauigkeit der Klassifikation schlechter wird.

Die Auswahl der relevanten Merkmale erfolgt, indem das β -Redukts der Merkmals-Äquivalenzrelationen zu der vom Entscheidungsattribut induzierten Äquivalenzrelation gebildet wird. Der für β verwendete Wert ist vermutlich von den beiden Parametern *Roughness* und *Rule Precision Threshold* abhängig. Mit diesen Parametern kann man das Verhalten des Algorithmus und damit die generierten Regeln beeinflussen. Zudem kann man festlegen, ob die generierten Regeln zusammenhängende Gebiete beschreiben sollen oder nicht.

Über die Zusammenfassung der Gebiete durch Zusammenfassung von Merkmalswerten eines Attributs werden keine Angaben gemacht. Vermutlich wird auch hier Reduktbildung eingesetzt. Das Zusammenfassen zweier Merkmalsausprägungen bedeutet ein Zusammenfallen von Elementarklassen. Dadurch wird die Feinkörnigkeit der Ununterscheidbarkeitsrelation vermindert. Dies kann solange durchgeführt werden, bis sich die Güte der Approximation verschlechtert. Das Problem der Zusammenfassung von Attributwerten kann leicht auf eine Attributreduktbildung zurückgeführt werden.

Die Bedeutung der Parameter *Roughness* und *Rule Precision Threshold*, deren Werte zwischen 0.00 und 1.00 variiert werden können, ist in der Dokumentation zu DataLogic/R nicht genau beschrieben. Man kann ihren Einfluß aber in etwa durch die Auswirkungen auf die Regeln beschreiben.

Mit dem Parameter *Rule Precision Threshold* legt man eine Mindestgenauigkeit der ausgegebenen Regeln fest. Die Genauigkeit einer Regel (*Precision*) ist definiert als das Verhältnis der richtig vorhersagten, abgedeckten Fälle zu allen von der Regel abgedeckten Fällen. In systematischen Voruntersuchungen habe ich festgestellt, daß der Einfluß dieses Parameters über weite Bereiche eher gering ist. Auch wenn man den *Rule Precision Threshold* auf 0.50 stellt, werden dennoch Regeln mit hoher Genauigkeit den weniger genauen Regeln vorgezogen. Es hat den Anschein, als ob der Algorithmus ohnehin versucht, möglichst genaue Regeln zu produzieren. M. E. ist ein hoher Schwellwert sinnvoll bei Anwendungen in denen eine Mißklassifikation besonders hohe Kosten verursacht. Bei Werten nahe 1.00 werden Regeln, die auch Beispiele aus ande-

ren Konzepten abdecken, verhindert. Man erwartet von Regeln, die mit einer solchen Parametereinstellung erzeugt werden, zwar hohe Vorhersagegüte, aber geringe Anwendbarkeit. Bei der Anwendung auf die phonologischen Daten sind allzu hohe Werte für den Rule Precision Treshold aufgrund des starken Noise nicht empfehlenswert, da sie die Erkennungsraten senken würden. In einer Spracherzeugungsanwendung könnte es dagegen besser sein, in einem unklaren Fall auf einen Wortakzent ganz zu verzichten, als möglicherweise gerade die falsche Silbe zu betonen.

Der Parameter Roughness legt die Detailliertheit der Regeln fest. Wenn man den Roughnessparameter erhöht, werden einfachere Regeln produziert. In Voruntersuchungen habe ich festgestellt, daß mit einem Roughnesswert von 1.00 für jedes Konzept nur noch eine Regel mit nur einem Attribut produziert wird. Eine solche einfache Regel induziert eine sehr grobe Äquivalenzrelation. Je mehr Attribute berücksichtigt werden, desto feiner wird die induzierte Äquivalenzrelation. Der Roughnessparameter verhält sich damit wie das Roughnessmaß der Rough Sets Theorie, das angibt, wie nahe die obere und untere Approximation an der zu beschreibenden Menge liegen (vgl. Abschnitt 2.1.2). Eine Möglichkeit, eine Konzeptbeschreibung zu erhalten, deren Roughness dem mit dem Roughnessparameter vorgegebenen Wert möglichst nahekommt, ist die β -Werte bei der Merkmalsreduktbildung und der Zusammenfassung der Blöcke entsprechend zu setzen.

Kapitel 3

Akustische Korrelate der Betonung des Deutschen

3.1 Untersuchungsgegenstand

Ziel der Untersuchungen in diesem phonetischen Teil der Arbeit ist, herauszufinden, welche akustisch–physikalischen Größen mit dem Wortakzent des Deutschen korrelieren, mit anderen Worten, es wird untersucht, wie sich betonte von unbetonten Silben eines Wortes akustisch unterscheiden. In der Literatur sind verschiedene Möglichkeiten beschrieben, mit denen ein Sprecher die Akzentuierung einer Silbe realisieren kann. So schreibt Bußmann:

Unter akustischem Aspekt kann Akzentuierung durch verschiedene Mittel bewirkt werden: Steigerung der Intensität bzw. Lautstärke, Veränderung der Tonhöhe (Intonation), Veränderung der Quantität (z. B. Dehnung von Lauten), Veränderung der Qualität (offene Vokale wirken stärker als geschlossene). [Bußm83]

Häufig wird dabei der Akzentuierung durch Tonhöhenänderung die wichtigste Rolle zuerkannt, wie beispielsweise im folgenden Zitat aus [Grew&al91].

Die [...] Gewichtung kann auf unterschiedliche Weise phonetisch realisiert werden:

- *durch eine größere Lautstärke der akzentuierten Silbe,*
- *durch eine längere Dauer der akzentuierten Silbe, insbesondere durch Längung des Vokals dieser Silbe, oder*
- *durch intonatorische Mittel wie Hebung oder Senkung der Tonhöhe auf oder unmittelbar nach der akzentuierten Silbe.*

Insbesondere die letzte Möglichkeit, der sog. «pitch accent», wird im Deutschen (und Englischen) zur Realisierung des Primärakzents benutzt. Den beiden anderen Möglichkeiten kommt eine geringere Be-

deutung zu, sie dienen im Deutschen oder Englischen meist zur phonetischen Realisierung von Nebenakzenten. [Grew&al91, S. 114]

Die letzte Bemerkung läßt sich so auffassen, daß der Wortakzent im Deutschen auf verschiedene Arten realisiert wird, und zwar so, daß sich der Akzent an hauptbetonter Stelle in einer Veränderung der meßbaren Grundfrequenz bemerkbar macht, während Nebenakzente auch durch Längung und/oder Lautstärkenänderung realisiert werden. Dieses Kapitel beschreibt eine Reihe von empirischen Untersuchungen über die Realisierung der Betonung im Deutschen. Die Grundlage der Untersuchung bilden Daten, die an der Universität Bielefeld experimentell erhoben wurden [Dogil95].

3.2 Datenmaterial und Datenaufbereitung

3.2.1 Gewinnung der Daten

Für die in Bielefeld durchgeführte Untersuchung wurde das Wort „Liliputaner“ ausgewählt. Für diese Wahl spricht, daß das Wort jedem Muttersprachler geläufig ist. So kann also angenommen werden kann, daß Versuchspersonen dieses Wort korrekt produzieren. Außerdem besteht das Wort aus fünf ähnlichen Silben. Der Onset (Anfangsrand) jeder Silbe besteht aus einem Konsonanten, die Coda (Koda) ist — sieht man von einer möglichen geschlossenen Artikulation der letzten Silbe ab — leer, und der Nukleus (Silbengipfel) besteht bei keiner Silbe aus einem Diphthong. Die ähnliche Struktur der Silben ermöglicht grobe Vergleiche etwa der Silbenlänge betonter und unbetonter Silben.

Aufgrund der intrinsischen Länge der Vokale sind bei dem Wort Liliputaner temporale Vergleiche nur eingeschränkt sinnvoll, denn ein tiefer Vokal wird normalerweise länger ausgesprochen als ein hoher. Auch werden gespannte Vokale im allgemeinen länger artikuliert als ungespannte. Außerdem sind auch Vergleiche der Silbenintensität problematisch wegen der unterschiedlichen intrinsischen Intensität der beteiligten Segmente, besonders der Konsonanten. Deshalb wurden die Versuchsreihen zusätzlich mit einem logatomisierten Wort durchgeführt. Dazu wurden die Sprecherinnen aufgefordert, das Wort „Liliputaner“ im Trägersatz durch das Kunstwort „Dadadadada“ zu ersetzen und dabei die Satzmelodie beizubehalten. Da die Silben gleich aufgebaut sind, ist bei der Logatomisierung ein genauerer Vergleich möglich.

Als Trägersatz diene „Wir haben den Liliputaner im Zirkus gesehen“, da dieser Satz auf natürliche Art ermöglicht, daß das untersuchte Wort einmal fokussiert und einmal nicht fokussiert wird. Dazu wurde er in die folgenden Kontexte eingebettet:

Kontext 1:

A: Wir haben einen Liliputaner im Zirkus gesehen.

B: Wen habt ihr im Zirkus gesehen?

A: Wir haben den *Liliputaner* im Zirkus gesehen.
(bzw.: Wir haben den *Dadadadada* im Zirkus gesehen.)

Kontext 2:

A: Wir haben einen Liliputaner im Zirkus gesehen.
B: Wo habt ihr den Liliputaner gesehen?
A: Wir haben den Liliputaner im *Zirkus* gesehen.
(bzw.: Wir haben den Dadadadada im *Zirkus* gesehen.)

Im Kontext 1 ist „Liliputaner“ bzw. „Dadadadada“ fokussiert und erhält klar die Hauptbetonung im Satz. Die vierte Silbe vereint so Satz- und Wortakzent. Dies wurde auch durch Hörexperimente von am Experiment nicht beteiligten Linguisten bestätigt. Im zweiten Kontext ist „Zirkus“ im Fokus und erhält die Hauptbetonung, d.h. den Satzakzent. Auf die vierte Silbe von Liliputaner fällt lediglich eine Nebenbetonung, der Wortakzent.

Für den Versuch standen fünf muttersprachliche Studentinnen als Sprecherinnen zur Verfügung. Jede Versuchsteilnehmerin sprach den Trägersatz zweimal in beiden Kontexten sowohl normal als auch logatomisiert, so daß das Datenmaterial insgesamt 40 Sätze umfaßt. Die Äußerungen wurden an der Universität Bielefeld mit einem DAT-Recorder aufgezeichnet.

3.2.2 Merkmalsextraktion

Ebenfalls an der Universität Bielefeld wurden mit dem digitalen Signalverarbeitungsprogramm S-Tools aus den Silben Merkmale berechnet. Für jede der fünf Silben des Wortes „Liliputaner“ bzw. dessen Logatomisierung wurden über die Dauer der Silbe folgende Merkmale bestimmt:

- der Mittelwert der Grundfrequenz in Hertz (**F0**),
- die Standardabweichung der Grundfrequenz in Hertz (**SD_F0**),
- der Mittelwert der Lautstärke in Dezibel (**dB**),
- die Standardabweichung der Lautstärke in Dezibel (**SD_dB**),
- die absolute Dauer in Sekunden (**Dur**) und
- die auf die Wortlänge normalisierte Zeit in Prozent der Länge des gesamten Worts (**Dur_Norm**).

Zusätzlich wurde in dem Merkmal **Focus** festgehalten, ob das Wort, aus dem die Silbe stammt, durch den Kontext fokussiert wurde oder nicht.

Bei der Bielefelder Datenerhebung wurden zwei Dinge versäumt:

- Die Werte des Grundfrequenzmerkmals sind nicht normalisiert. Wünschenswert wäre eine Normalisierung auf den Mittelwert der gesamten Äußerung. Nur so sind Vergleiche zwischen Messungen von verschiedenen Sprecherinnen sinnvoll.
- Auch die Lautstärkemessungen sind nicht normalisiert. Durch unterschiedlichen Mikrofonabstand von Aufnahme zu Aufnahme kann es zu unerwünschten Lautstärkeunterschieden kommen. Auch hier wäre eine Normalisierung auf den Mittelwert für die Äußerung des ganzen Satzes sinnvoll. Auf die Standardabweichung der Lautstärke hat eine solche Normalisierung keinen Einfluß.

Aus den vorhandenen Merkmalen können die erwünschten normalisierten Werte jedoch genähert werden. Man schätzt den Mittelwert von Grundfrequenz bzw. Lautstärke des ganzen Satzes durch den Mittelwert des gemessenen Wortes und erhält somit eine Normalisierung bezüglich dem Wort. Dadurch gehen allerdings Anhebungen/Absenkungen des ganzen Wortes gegenüber dem Satzniveau verloren. Auf diese Weise erhält man die beiden folgenden neuen Merkmale:

- geschätzte normalisierte Grundfrequenz (**F0_Norm**) und
- geschätzte normalisierte Lautstärke (**dB_Norm**).

In den folgenden Untersuchungen wurden diese Merkmale anstelle ihrer nicht normalisierten Pendanten verwendet.

3.3 Lernreihen und Ergebnisse

Mit diesen Daten wurden acht Lernreihen durchgeführt. Dabei wurde die Untersuchung einmal für die normal gesprochenen Wörter und einmal für die Logatome vorgenommen. Die Daten für die jeweils 100 Silben (5 Sprecherinnen, 5 Silben, 2 Kontexte, 2 Durchgänge) wurden pseudozufällig in fünf gleich große Teile zerlegt.

Vier Teile wurden als Trainingsmenge benutzt, der fünfte Teil bildete die Testmenge, sodaß sich ein *Learn* \neq *Test*-Szenario ergibt. In den einzelnen Lernreihen wurden dem Lernprogramm verschiedene Untermengen der Merkmalsmenge angeboten, um die Bedeutung der einzelnen Merkmale für die Vorhersage neuer Daten zu untersuchen. Die im Kapitel über die Rough Sets Theorie angesprochenen Redukte geben vergleichbare Aufschlüsse, beziehen sich jedoch immer auf die Lernmenge und nicht auf eine dazu disjunkte Testmenge. Die Parameter des Lernprogramms wurden konstant bei *Roughness* = 0.50, *PrecisionThreshold* = 0.95 gehalten. Stichprobenhaft durchgeführte Läufe mit kleineren Werten für *Precision Threshold* ergaben keine wesentlichen Abweichungen.

3.3.1 Erste Lernreihe: Alle Merkmale

Liliputaner			Dadadadada			Legende
Dur_Norm	FO_Norm	SD_FO	Dur_Norm	FO_Norm	SD_FO	angeboten
dB_Norm	SD_dB	Focus	dB_Norm	SD_dB	Focus	verwendet
Lauf A		100	Lauf A		100	<div>■ = richtig klassifiziert</div> <div>■ = keine Entscheidung</div> <div>■ = falsch klassifiziert</div>
Lauf B		95	Lauf B		100	
Lauf C		100	Lauf C		100	
Lauf D		100	Lauf D		100	
Lauf E		100	Lauf E		100	
Gemittelt		99	Gemittelt		100	

In der ersten Lernreihe wurden dem Lernprogramm alle Merkmale angeboten, also **Dur_Norm**, **FO_Norm**, **SD_FO**, **dB_Norm**, **SD_dB** und **Focus**. Die Ergebnisse dieser Reihe sind für beide Wörter sehr klar. Der Algorithmus wählte als einziges Merkmal **Dur_Norm** aus. Dieses Merkmal alleine reicht aus, um 99% bzw. 100% der Silben aus den Testmengen von Liliputaner- bzw. Dadadadada-Silben korrekt als betont oder unbetont zu klassifizieren.

Sämtliche fünf Läufe mit den Liliputaner-Silben ergaben im wesentlichen die gleichen zwei einfachen Regeln:

```
Decision :: Accent==> No
=====
1 |   |   | [Dur_Norm<=26.00]
```

```
Decision :: Accent==> Yes
=====
2 |   |   | [Dur_Norm>26.00]
```

Lediglich in Lauf B wählte der Algorithmus als Grenze für die Unterscheidung 28.85% der Wortlänge anstelle von 26.00%. Die Regeln bedeuten, daß betonte Silben länger sind als unbetonte. Die Grenze von 26% bzw. 28.85% ist plausibel: wären alle Silben gleich lang, hätten sie eine relative Länge von 20%, da unser Wort ja aus 5 Silben besteht.

Die Testmengen der Liliputaner-Silben wurden alleine mit dem Längenmerkmal **Dur_Norm** zu 99% korrekt klassifiziert. Eine einzige betonte Silbe wurde fälschlicherweise als unbetont klassifiziert. Es handelte sich um eine Silbe der Testmenge von Lauf B mit einer relativen Länge von 28.58%.

Bei den Dadadadada-Silben scheint die Länge der betonten Silben nicht so stark ausgeprägt zu sein wie bei den Liliputaner-Silben, aber immer noch ausreichend, um sie von den unbetonten zu unterscheiden. Alle fünf Läufe ergaben dieselben Regeln:

```
Decision :: Accent==> No
=====
1 |   |   | [Dur_Norm<=24.60]
```

```
Decision :: Accent==> Yes
=====
2 |   | [Dur_Norm>24.60]
```

Die Anforderungen an betonte Silben sind hier gelockert, es genügt nun eine relative Länge von knapp 25%, um eine Silbe als betont zu klassifizieren. Der Unterschied zu den obigen Grenzwerten könnte in der intrinsischen Länge der Vokale begründet sein, die durch die Logatomisierung ausgeglichen wurde.

Die Testmengen der Dadadadada-Silben wurden alleine mit dem Längenmerkmal `Dur_Norm` zu 100% korrekt klassifiziert.

3.3.2 Zweite Lernreihe: Ohne Zeitinformation

Liliputaner			Dadadadada			Legende
FO_Norm SD_F0			FO_Norm SD_F0			angeboten
dB_Norm	SD_dB	Focus	dB_Norm	SD_dB	Focus	verwendet
Lauf A		100	Lauf A		70	<div>■ = richtig klassifiziert</div> <div>■ = keine Entscheidung</div> <div>■ = falsch klassifiziert</div>
Lauf B		100	Lauf B		65	
Lauf C		100	Lauf C		75	
Lauf D		95	Lauf D		80	
Lauf E		100	Lauf E		90	
Gemittelt		99	Gemittelt		76	

In der zweiten Lernreihe wurde das Zeitmerkmal `Dur_Norm` aus der dem Lernalgorithmus zur Verfügung stehenden Merkmalsmenge entfernt. Mit dieser Reihe sollte untersucht werden, ob neben dem Zeitmerkmal andere Merkmale ausreichend sind, um betonte von unbetonten Silben zu unterscheiden.

Bei den Liliputaner-Silben ergibt sich mit durchschnittlich 99% eine gleich gute Vorhersagegenauigkeit wie in der ersten Lernreihe. Bei den logatomisierten Silben bricht jedoch die Güte der Vorhersage von 100% auf 76% ein. Der auf den ersten Blick immer noch hoch erscheinende Wert relativiert sich rasch, wenn man bedenkt, daß der triviale Klassifikator „jede Silbe ist unbetont“ auf 80% Vorhersagegenauigkeit kommen würde, da ja nur 20 der 100 Silben betonte Silben sind. Eine Balancierung der Daten, also eine zufällige Streichung von unbetonten Silben würde die Skalierung entzerren. Meist werden in Lernexperimenten wie diesem die Daten so balanciert, daß die zu lernenden Konzepte in der Trainingsmenge etwa gleich häufig vorkommen. Man kann aber auch eine in der Anwendung zu erwartende Verteilung anstreben. In Anhang A habe ich untersucht, aus wievielen Silben ein Wort im Deutschen durchschnittlich besteht. Wenn man realistische Verteilungen zwischen betonten und unbetonten Silben erhalten will, müßte man demnach dreisilbige Wörter untersuchen oder die Anzahl der unbetonten Silben bis zum Verhältnis 2 : 1 verringern.

Bei den Dadadadada-Silben wurde lediglich das `Focus`-Merkmal nicht verwendet. Sonst wurden alle restlichen Merkmale (`FO_Norm`, `SD_F0`, `dB_Norm`, `SD_dB`) in den Läufen verwendet. DataLogic/R gibt an, wie wichtig die Merkmale für die Entscheidung waren. Die Veränderung in der Lautstärke (`SD_dB`) war i. a. das wichtigste Merkmal, die normalisierte Grundfrequenz (`FO_Norm`) wurde i. a.

```

Decision :: Accent==> No
=====
1 |   |   | [SD_dB<=4.93 or SD_dB>9.51]
  | OR
2 |   |   | [dB_Norm<=-3.84 or dB_Norm>1.31] & [6.08< SD_dB<=9.51]

Decision :: Accent==> Yes
=====
3 |   |   | [-3.84< dB_Norm<=1.31] & [4.93< SD_dB<=9.51]
  | OR
4 |   |   | [4.93< SD_dB<=6.08]

```

Tafel 3.1: Die bei den Läufen A, B, C und E erzeugten Regeln für die Silben des Wortes Liliputaner, zweite Lernreihe (ohne Zeitinformation)

von den gewählten Merkmalen als am unwichtigsten eingeschätzt. Wie bereits erwähnt, wurde auch in dieser Reihe das Merkmal *Focus nicht* benutzt; wenn man annimmt, daß Haupt- und Nebentakzente unterschiedlich realisiert werden, wäre zu erwarten gewesen, daß dieses Merkmal in Betracht gezogen wird.

Ganz anders sieht die Situation bei den Liliputaner-Silben aus. Für sie konnten genauso gute Erkennungsraten erreicht werden wie in der ersten Lernreihe. In sämtlichen Läufen wählte der Algorithmus die beiden Lautstärkemerkmale *dB_Norm* und *SD_dB* aus, wobei er *SD_dB* etwas größere Wichtigkeit beimaß. Auf die Grundfrequenzmerkmale wurde nicht zurückgegriffen. Schaut man sich die Daten der Silben im *dB_Norm-SD_dB*-Raum an, wird klar, warum dem so ist: der Algorithmus hat nicht erkannt, ob eine Silbe betont ist oder nicht, sondern er hat erkannt, ob es sich bei der Silbe um eine *ta*-Silbe handelt. Betrachtet man die Intensität der einzelnen Silben etwas genauer, so sieht man einerseits, daß die Vokale (hier die beiden /*ɪ*/, das /*u*/, /*a*/ und den Schwa der letzten Silbe) in etwa gleich große Intensität aufweisen. Andererseits sind die Silbenonsets in ihrer Intensität stark unterschiedlich: Die Sonorantkonsonanten /*l*/ und /*n*/ sind deutlich lauter als die Plosive /*p*/ und /*t*/ mit ihren Verschlußphasen, in denen Stille vorliegt. Da die Lautheit der Sonorantkonsonanten nahezu an die der Vokale heranreicht, hingegen die Plosive deutlich leiser sind, wird klar, daß die Standardabweichung der Lautstärke bei der dritten und vierten Silbe größer und der Mittelwert der Lautstärke kleiner sein muß als bei den anderen Silben. Wie man sowohl Abb. 3.1 als auch den gelernten Regeln, exemplarisch eine davon in Tafel 3.1, entnehmen kann, macht sich der Lernalgorithmus genau diesen Effekt zunutze, den ich im folgenden Spracherkennungseffekt nennen werde. Die abgebildeten Regeln wurden genau gleich von den vier Läufen produziert, die alle Testfälle korrekt klassifizierten.

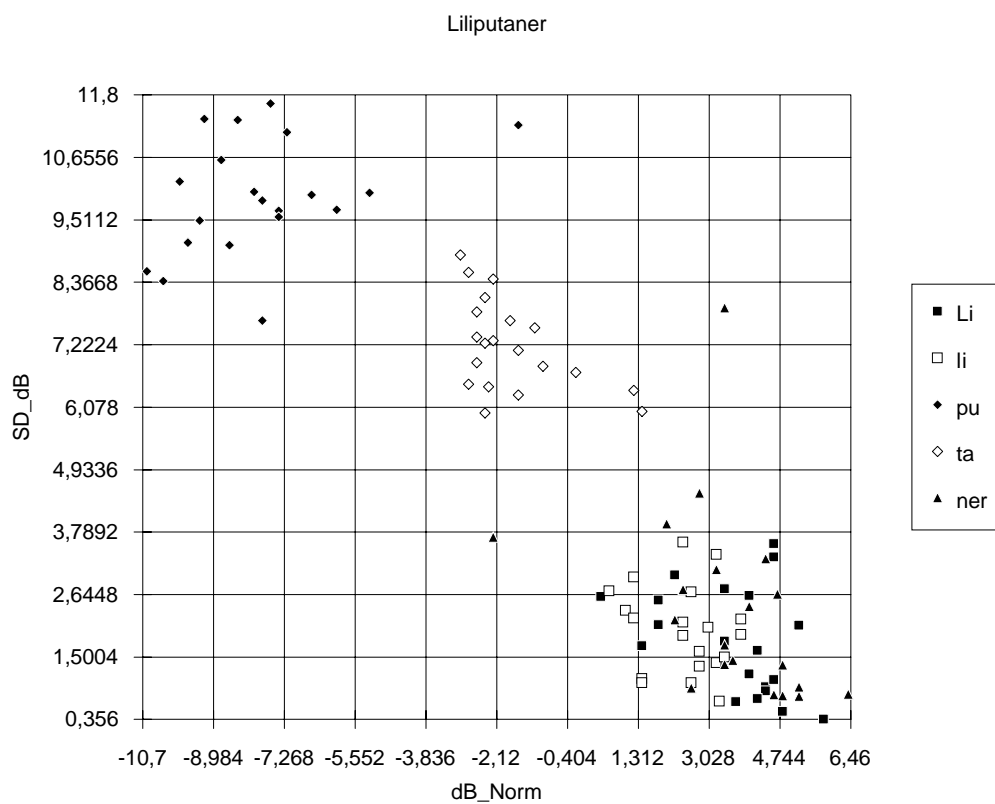


Abbildung 3.1: Der Merkmalsunterraum für die Merkmale dB_Norm und SD_dB. Eingezeichnet sind die Orte für die Silben des Wortes Liliputaner.

3.3.3 Dritte Lernreihe: nur Lautstärkemerkmale

Liliputaner		Dadadadada		Legende
dB_Norm	SD_dB	dB_Norm	SD_dB	angeboten verwendet
Lauf A	100	Lauf A	50	<div>■ = richtig klassifiziert</div> <div>■ = keine Entscheidung</div> <div>■ = falsch klassifiziert</div>
Lauf B	100	Lauf B	65	
Lauf C	100	Lauf C	70	
Lauf D	95	Lauf D	60	
Lauf E	100	Lauf E	70	
Gemittelt	99	Gemittelt	63	

In der dritten Lernreihe wurde untersucht, ob die guten Erkennungsraten bei den Liliputaner-Silben der vorangegangenen Reihe tatsächlich auf dem beschriebenen Spracherkennungseffekt zurückzuführen ist. Aus diesem Grund wurden dem Algorithmus nur die Lautstärkemerkmale dB_Norm und SD_dB angeboten.

Sowohl für die Liliputaner- als auch die Dadadadada-Silben wurden beide Merkmale in den gelernten Regeln benutzt. Für die Liliputaner-Silben haben sich weder die Regeln noch die Erkennungsraten gegenüber der zweiten Lernreihe geändert, da der Algorithmus dort nur Merkmale gewählt hatte, die auch hier bereitstehen.

Aus den Ergebnissen für die Dadadadada-Silben läßt sich ersehen, daß tatsächlich die Lautstärkeunterschiede der Konsonanten wesentlich zu den guten Erkennungsraten der Liliputaner-Silben beitrugen. Die Erkennungsraten für die Dadadadada-Silben sinken ohne die Grundfrequenzmerkmale erwartungsgemäß weiter. Sie liegen nun bei 63% korrekt vorhergesagten und 19% falsch vorhergesagten Fällen. In 18% der Fälle hat der Klassifikator nicht entschieden. Bei den gelernten Regeln ist über die fünf Läufe keine einheitliche Struktur erkennbar. Zum Vergleich mit den Liliputaner-Silben (Abbildung 3.1) sind die Daten für die Dadadadada-Silben im Merkmalsunterraum der beiden Lautstärkemerkmale in Abbildung 3.2 dargestellt. Man sieht deutlich, daß die Bereiche der einzelnen Silben sich nunmehr wesentlich stärker überschneiden. Man kann jedoch auch noch hier Tendenzen ausmachen, beispielsweise scheint die betonte vierte Silbe sich bevorzugt im oberen linken Viertel aufzuhalten, die erste und fünfte Silbe eher im unteren rechten Viertel. Unerwartet ist, daß die betonte Silbe sich hauptsächlich durch größere Standardabweichung (also vermutlich einem stärkeren Lautstärkekontrast zwischen Plosiv und Vokal) und keinesfalls durch eine besonders große mittlere Lautstärke auffällig verhält. Im Mittel sind beispielsweise die erste und letzte Silbe lauter als die betonte, wie Abbildung 3.3 belegt.

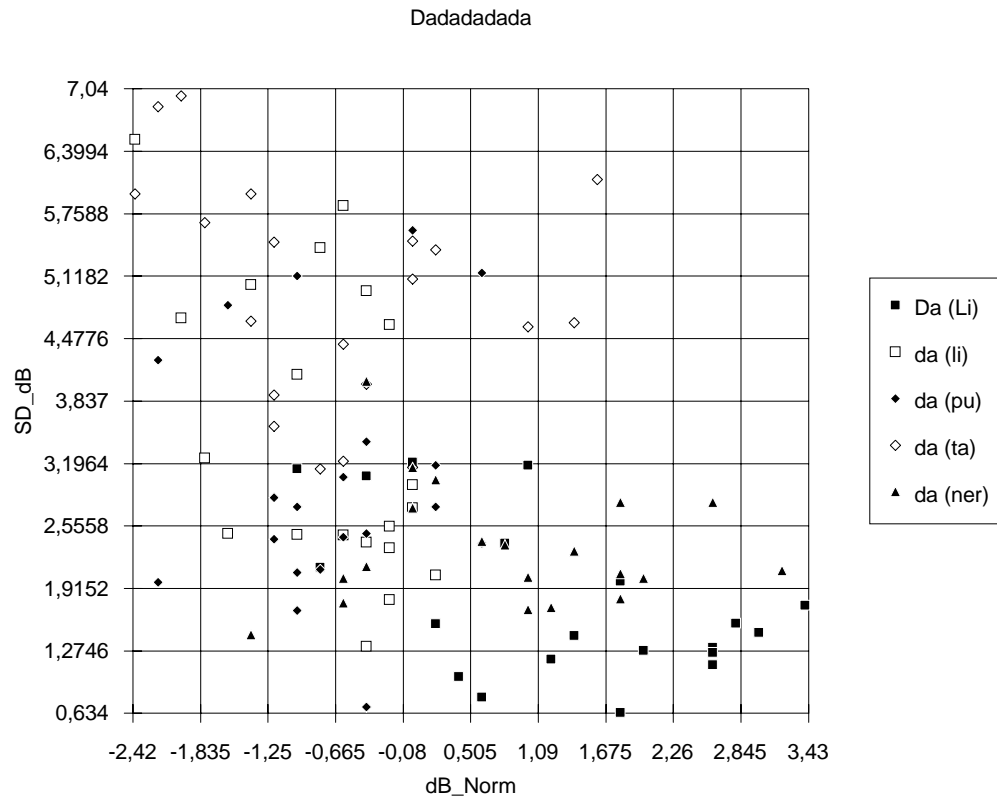


Abbildung 3.2: Der Merkmalsunterraum für die Merkmale `dB_Norm` und `SD_dB`. Eingezeichnet sind die Orte für die Silben des Wortes *Dadadadada*.

3.3.4 Vierte Lernreihe: Grundfrequenzmerkmale und Änderung der Lautstärke

Liliputaner			Dadadadada			Legende
	FO_Norm	SD_F0		FO_Norm	SD_F0	angeboten
	SD_dB			SD_dB		verwendet
Lauf A	100		Lauf A	75		<div>■ = richtig klassifiziert</div> <div>■ = keine Entscheidung</div> <div>■ = falsch klassifiziert</div>
Lauf B	85		Lauf B	65		
Lauf C	95		Lauf C	75		
Lauf D	90		Lauf D	80		
Lauf E	90		Lauf E	75		
Gemittelt	92		Gemittelt	74		

In dieser Lernreihe wurde dem Algorithmus das Lautstärkemerkmale vorenthalten. Dadurch sollte untersucht werden, ob stattdessen die Grundfrequenzmerkmale die Informationen zur Unterscheidung betonter und unbetonter Silben liefern können. Sowohl bei den normalen Silben als auch den Logatomisierungen wurden alle drei Merkmale in den Regeln verwendet.

Die erzielten Erkennungsraten sind für die Liliputaner-Silben schlechter als in den vorangegangenen Reihen. Es scheint der Fall zu sein, daß die Erkennung der Silbe /*ta*/ durch den Wegfall eines der Lautstärkemerkmale bereits gestört

Veränderung der Klassenmitten durch Logatomisierung

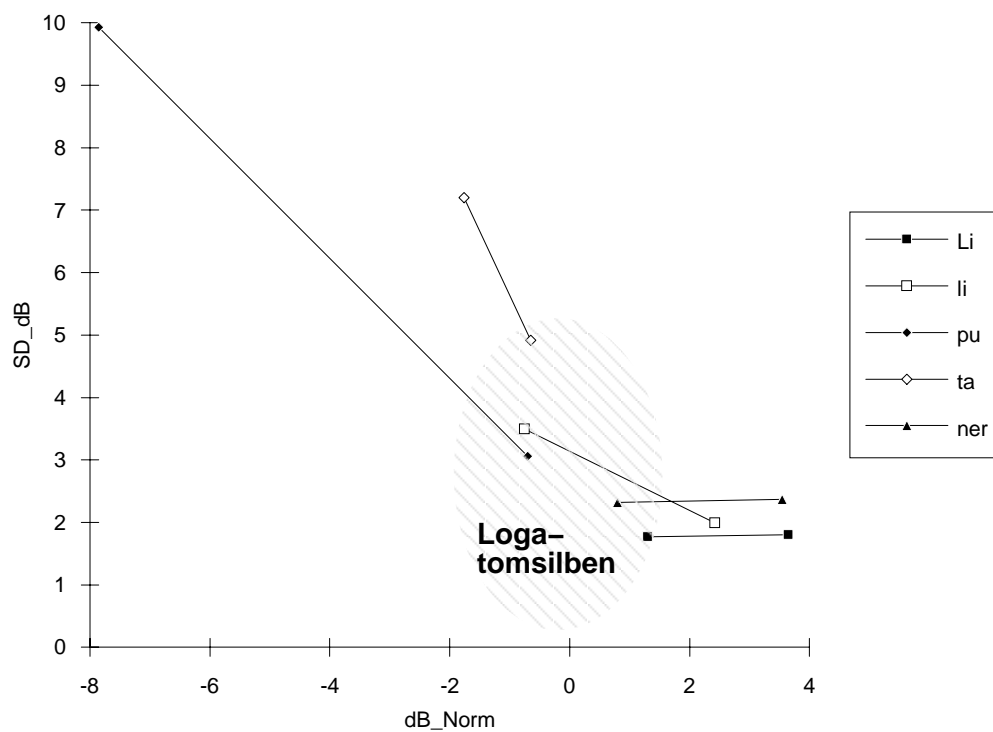


Abbildung 3.3: Der Merkmalsunterraum für die Merkmale `dB_Norm` und `SD_dB`. Eingezeichnet sind die Mittelwerte für die Silben der Worte Liliputaner und Dadadadada. Das Diagramm verdeutlicht, daß durch die Logatomisierung die Unterschiede zwischen den einzelnen Silben verringert werden. Die Orte der Silben von Dadadadada sind in der Mitte des Schaubilds, während die Liliputaner-Silben sich ringsum befinden.

```

Decision :: Accent==> No
=====
1 |   |   | [SD_dB<=2.56]
  | OR
2 |   |   | [SD_dB<=5.12] & [dB_Norm<=-1.25]
  | OR
3 |   |   | [SD_dB<=3.20] & [-0.66< dB_Norm<=-0.08]
  | OR
4 |   |   | [SD_dB<=4.48] & [dB_Norm>0.51]
  | OR
5 |   |   | [4.48< SD_dB<=5.12] & [dB_Norm<=0.51]
  | OR
6 |   |   | [SD_dB>5.12] & [dB_Norm>0.51]

Decision :: Accent==> Yes
=====
7 |   |   | [3.20< SD_dB<=3.84] & [-1.25< dB_Norm<=-0.66]
  | OR
8 |   |   | [5.12< SD_dB<=6.40] & [dB_Norm<=-1.25]
  | OR
9 |   |   | [3.20< SD_dB<=5.12] & [dB_Norm>0.51]
  | OR
10 |  |   | [SD_dB>5.12] & [-0.66< dB_Norm<=0.51]

```

Tafel 3.2: Die beim Lauf B erzeugten Regeln für die Silben des Wortes Dadadadada, dritte Lernreihe (nur Lautstärkemerkmale)

wird. Andererseits ist der Spracherkennungseffekt immer noch groß genug, um besser als Zufall zu klassifizieren (92%).

Bei den vom Spracherkennungseffekt nicht betroffenen Dadadadada-Silben konnten Erkennungsraten erreicht werden, die nahezu an die zweite Lernreihe heranreichen, auf alle Fälle jedoch besser als die Ergebnisse der dritten Reihe sind. Hier konnten die beiden Grundfrequenzmerkmale die vom dB_Norm-Merkmal gelieferte Information ersetzen. Es muß dabei jedoch immer bedacht werden, daß bereits in der zweiten Reihe, und daher umso mehr in den folgenden Reihen, die Erkennungsraten bereits inakzeptabel schlecht sind.

3.3.5 Fünfte Lernreihe: Grundfrequenzmerkmale und Lautstärke

Liliputaner			Dadadadada			Legende	
	FO_Norm	SD_F0		FO_Norm	SD_F0	angeboten	
	dB_Norm			dB_Norm		verwendet	
Lauf A	85		Lauf A	45		<div></div> = richtig klassifiziert <div></div> = keine Entscheidung <div></div> = falsch klassifiziert	
Lauf B	90		Lauf B	45			
Lauf C	95		Lauf C	70			
Lauf D	95		Lauf D	70			
Lauf E	95		Lauf E	70			
Gemittelt	92		Gemittelt	60			

Die fünfte Reihe ist gewissermaßen das Pendant zur vierten, denn sie untersucht, inwieweit das andere Lautstärkemerkmale, die Veränderung SD_{dB} , durch FO_{Norm} und SD_{FO} ersetzt werden kann.

Bei den Liliputaner-Silben ergibt sich im Mittel die gleiche Vorhersagegüte wie in der vorherigen Lernreihe, es gilt das oben gesagte. Bei den Logatomisierungen zeigt sich, daß die Erkennungsraten hinter der dritten Reihe zurückstehen. Man kann also sagen, daß das Merkmal SD_{dB} nicht durch die beiden Grundfrequenzmerkmale substituiert werden kann.

3.3.6 Sechste Lernreihe: nur Grundfrequenzmerkmale

Liliputaner			Dadadadada			Legende
FO_{Norm} SD_{FO}			FO_{Norm} SD_{FO}			angeboten verwendet
Lauf A	40		Lauf A	10		<div>■ = richtig klassifiziert</div> <div>■ = keine Entscheidung</div> <div>■ = falsch klassifiziert</div>
Lauf B	40		Lauf B	20		
Lauf C	50		Lauf C	25		
Lauf D	50		Lauf D	25		
Lauf E	40		Lauf E	40		
Gemittelt	44		Gemittelt	24		

Eine Lernreihe, in denen dem Algorithmus nur die Grundfrequenzmerkmale angeboten wurden, bestätigt, daß diese Merkmale zur Vorhersage der Betonung kaum geeignet sind. Die Erkennungsraten sind indiskutabel schlecht, bei den Liliputaner-Silben wurden 44% korrekt klassifiziert, bei den Dadadadada-Silben nur 24%.

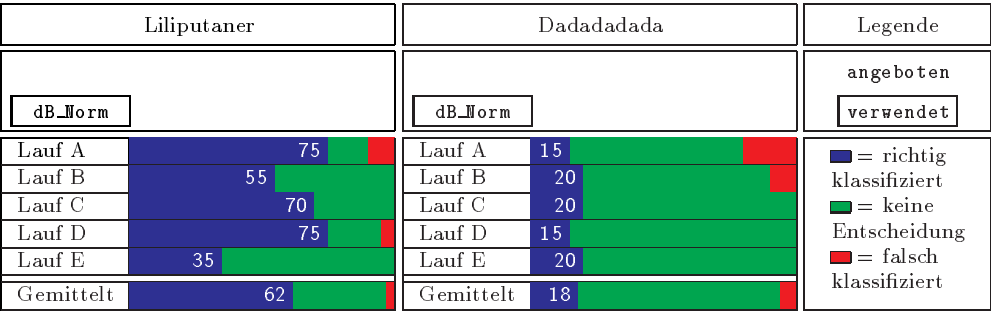
3.3.7 Siebte Lernreihe: nur Änderung der Lautstärke

Liliputaner			Dadadadada			Legende
SD_{dB}			SD_{dB}			angeboten verwendet
Lauf A	95		Lauf A	45		<div>■ = richtig klassifiziert</div> <div>■ = keine Entscheidung</div> <div>■ = falsch klassifiziert</div>
Lauf B	75		Lauf B	50		
Lauf C	90		Lauf C	55		
Lauf D	75		Lauf D	35		
Lauf E	80		Lauf E	60		
Gemittelt	83		Gemittelt	49		

In den letzten beiden Lernreihen wurde untersucht, inwiefern die beiden Lautstärkemerkmale alleine eine Vorhersage ermöglichen. In der siebten Reihe wurde dazu dem Lernprogramm nur das anscheinend informativere SD_{dB} zugänglich gemacht (vgl. Reihen 4 und 5). Die Erkennungsraten sind hier ebenfalls vergleichsweise schlecht, wenngleich der Klassifikator bei den Liliputaner-Silben sogar die 80% eines trivialen Klassifikators übertrifft (83%). Auf alle Fälle ist das Merkmal alleine besser als die Kombination der beiden Grundfrequenzmerkmale. Bemerkenswert ist jedoch der geringe Anteil an Mißklassifikationen.

Bei den Liliputaner-Silben wurden keine Silben falsch klassifiziert, die verbleibenden 17% wurden nicht entschieden. Bei den Dadadadada-Silben stehen 49% korrekt entschiedenen Fällen nur 4% Mißklassifikationen gegenüber.

3.3.8 Achte Lernreihe: nur Lautstärke



Das zweite Lautstärkemerkmale dB_Norm alleine kann, vermutlich aufgrund des Spracherkennungseffekts, bei den Liliputaner-Silben bessere Erkennungsraten erzielen als die beiden Grundfrequenzmerkmale, nämlich 62% gegenüber 44%. Bei den nur durch Betonung unterschiedlichen da-Silben liegt das einzelne Merkmal jedoch dahinter (18% gegenüber 24%). Auf alle Fälle sind die Raten schlechter als bei der Standardabweichung der Lautstärke, eine Tatsache, die wir bereits bei der Betrachtung der dB_Norm-SD_dB-Räume vermuteten.

3.4 Zwischenbilanz

Die Ergebnisse der Lernreihen ergeben ein klares Bild über die Nützlichkeit der untersuchten Merkmale für die Vorhersage der Betonung einer Silbe. Die Länge einer Silbe kann demzufolge eindeutig als bester Indikator für eine betonte Silbe betrachtet werden. Wie wir gesehen haben, ist es durch den Versuchsaufbau (es wurden nur zwei ‚Wörter‘ untersucht) nicht immer klar, ob die entdeckten Regelmäßigkeiten tatsächlich auf die Betonung zurückzuführen sind, da die beteiligten unterschiedlichen Phoneme einen Einfluß auf nahezu alle Merkmale besitzen. Es sei ausdrücklich darauf hingewiesen, daß auch das Längenmerkmal diesem Einfluß unterliegt. Da der Vokal der betonten Silbe im Wort Liliputaner sicherlich die größte intrinsische Länge besitzt, kann man die guten Erkennungsraten bei ausschließlicher Verwendung des Längenmerkmals auch auf diesen, zunächst nicht prosodisch verursachten Umstand zurückführen. Durch die parallelen Untersuchungen mit den Logatomisierungen wurde versucht, die Unterschiede der Silben nur auf die durch die Betonung verursachten Unterschiede zu reduzieren. Bei diesen Silben fällt es schwer, die Argumentation aufrechtzuerhalten, daß die Längung der betonten Silbe nur aufgrund der intrinsischen Länge des Vokals zustandekommt. Es könnte lediglich angeführt werden, daß für die unbetonten Silben ein ungespanntes a, für die betonte Silbe ein gespanntes a verwendet wurde, möglicherweise weil das a in Liliputaner gespannt gesprochen würde. Die Unterscheidung zwischen einem gespannten und

ungespanntem a ist jedoch nicht unumstritten. Beispielsweise wird in [Mang90] nur eine Sorte a transkribiert. Aber selbst wenn man von zwei unterschiedlichen a-Lauten ausgehen möchte, ist nicht klar, ob man daraus eine Herabsetzung der Wichtigkeit des Längenmerkmals ableiten kann. Wie ganz am Anfang des Kapitels gezeigt, nehmen manche Autoren an, daß auch die Veränderung der Qualität (und dadurch die sich daraus ergebende intrinsische Länge) eine Realisationsmöglichkeit für den Akzent ist.

Um die Wichtigkeit des Längenmerkmals für die Betonungserkennung losgelöst von der Diskussion um eine mögliche Projektion der intrinsischen Länge auf das Logatom zu untersuchen, könnte man denselben Versuch noch einmal mit anderen Wörtern wiederholen. Dabei sollten mehrere Punkte beachtet werden:

- Die betonte Silbe sollte als Nukleus einen Vokal mit verhältnismäßig kurzer intrinsischer Länge haben.
- Bei der Logatomisierung sollten Phoneme verwendet werden, die nicht schon im Wort auftreten. Die Verwendung von Dododododo als Logatomisierung von Liliputaner hätte der Argumentation mit den zwei verschiedenen Vokalvarianten von vornherein die Grundlage entzogen.
- Bei der Auswertung sollten feinere Merkmale extrahiert werden. Es wäre z. B. sehr sinnvoll, die Zeiten von Nukleus und Onset einzeln zu extrahieren, um zu untersuchen, in welchem Teil der Silbe die Längung realisiert wird.

Auch für die Behandlung der Frage, warum die mittlere Lautstärke der Silbe kein verlässliches Merkmal für die Erkennung von Betonung ist, wäre eine getrennte Erfassung von Nukleus und Onset wünschenswert. Damit könnte die folgende These überprüft werden, die erklärt, warum die Standardabweichung von betonten Silben auch bei den Dadadadada-Silben mehr Aufschluß über die Betonung gibt als die mittlere Lautstärke: Möglicherweise wird bei der betonten Silbe eine Längung vor allem in der Verschlußphase des Plosivs vorgenommen. Der dadurch vergrößerte Anteil an schwachen Intensitätswerten würde — zusammen mit einer stärkeren Intensität während des Vokals — eine große Standardabweichung zur Folge haben, während sich die Einflüsse (verlängertes schwaches und verstärktes starkes Segment) für die mittlere Lautstärke neutralisieren würden. Diese These steht jedoch im Widerspruch zu der anfänglich zitierten Annahme, die Längung werde hauptsächlich im Vokal vorgenommen. Andererseits werden in der Musik derartige „off beat“ Akzentuierungen eingesetzt, z. B. bei Instrumenten, die eine Akzentuierung nicht durch Lautstärke ausdrücken können (Cembalo, Orgel). Die These wird auch von aktuellen Untersuchungen, die am Lehrstuhl Experimentelle Phonetik durchgeführt wurden, unterstützt [JessMara95].

Ein weiteres interessantes Ergebnis der Untersuchungen ist, daß kein Einfluß des Fokus auf eine mögliche unterschiedliche Realisierung des Akzents beobachtet werden konnte. Für eine genauere Untersuchung dieses Problems ist jedoch die vorhandene Datenmenge zu klein. Sinnvoll wäre eine Untersuchung,

bei der dem Lernalgorithmus die Merkmale der betonten und den beiden ihr benachbarten Silben zur Verfügung stehen. Die vorherzusagende Kategorie ist das Fokus-Merkmal, das für die Lernmenge bekannt ist. Leider ist die Anzahl der vorhandenen Daten für eine solche Untersuchung recht gering: es stehen lediglich jeweils 20 Beispiele zur Verfügung, eine Zahl, die durch die Aufteilung in Lern- und Testmengen noch weiter vermindert wird. Aufgrund der geringen Anzahl käme allenfalls ein leave-one-out-Verfahren in Frage.

Das überraschendste Ergebnis der Untersuchungen ist, daß die Grundfrequenzmerkmale fast nichts zur Vorhersage beitragen. Ein Grund für die schlechten Vorhersagequalitäten könnte sein, daß die Merkmale ungeeignet für die Repräsentation eines Pitchakzents sind. Der Pitchakzent wird allgemein als silbenübergreifendes Phänomen [Grew&al91]. In diesem Fall ist es möglich, daß die silbenbezogenen Merkmale, die in dieser Untersuchung verwendet wurden, den Pitchakzent nicht ausreichend widerspiegeln, weil sich die Einflüsse des Akzents so auf die Merkmale der betonten und der auf sie folgenden Silbe aufteilen, daß sie sich bei beiden nicht mehr signifikant bemerkbar machen. Im nächsten Abschnitt wird versucht, die Grundfrequenzmerkmale der betonten und der auf sie folgenden Silbe gemeinsam zu behandeln, um so dem silbenübergreifenden Charakter des Pitchakzents besser gerecht zu werden.

3.5 Pitchakzent und Differenzmerkmale

Dieser Abschnitt befaßt sich mit dem Versuch, den Pitchakzent in den vorhandenen Daten zu entdecken, indem aus den ursprünglichen Merkmalen neue, silbenübergreifende Merkmale, gewonnen werden. Zu diesem Zweck werden aus den Merkmalen `FO_Norm` und `SD_FO` die beiden neuen 'Differenzmerkmale' `d_FO_Norm` bzw. `d_SD_FO` gebildet, die aussagen, wie sich die zugrundeliegenden Merkmale von Silbe zu Silbe ändern. Da keine Informationen über die Merkmalsausprägungen vor der ersten sowie nach der fünften Silbe vorliegen, gibt es nur vier Differenzmerkmale pro Wort. Die silbenübergreifenden Merkmale werden durch Subtraktion der silbenbezogenen Merkmalswerte gewonnen. Für die Standardabweichung wäre eine Quotientenbildung resp. prozentuale Änderungsangabe ebenfalls denkbar, für die bereits normalisierten F_0 -Werte ist dies jedoch nicht möglich.

Da der Pitchakzent im Falle einer silbenübergreifenden Realisierung auf der betonten und der darauf folgenden Silbe und nicht auf der vorausgehenden realisiert wird, sollte der Lernalgorithmus die Abweichung der vierten Differenzmerkmale von den übrigen ausmachen können. Damit auch eine vollständige Realisierung auf der betonten Silbe, die sich in einer vergrößerten F_0 -Standardabweichung derselben niederschlagen müßte, vom Algorithmus in Betracht gezogen werden kann, werden den Differenzmerkmalen auch die silbenbezogenen Merkmale der vorderen Silbe zur Seite gestellt. Damit stehen dem Lernalgorithmus für die folgende Untersuchung die vier Merkmale `FO_Norm`, `d_FO_Norm`, `SD_FO` und `d_SD_FO` für insgesamt jeweils 80 Fallbeispiele (4 Silbenpaare, 5 Sprecherinnen, 2 Kontexte, 2 Durchgänge) zur Verfügung. Auch hier werden die

Untersuchungen für die Wörter Liliputaner und Dadadadada getrennt durchgeführt. Um die im letzten Abschnitt angesprochene unterschiedliche Realisierung des Akzents an fokussierter und nicht fokussierter Stelle zu untersuchen, werden zudem die Beispiele nach dem Merkmal **Focus** in zwei Mengen aufgeteilt. Wenn man der Annahme folgen möchte, daß die Akzentuierung an hauptbetonter Stelle bevorzugt durch Pitchakzent erfolgt, müßten sich bei den fokussierten Silben bessere Erkennungsraten ergeben als bei den nicht fokussierten, da nur Grundfrequenzmerkmale und keine Längen- oder Intensitätsmerkmale zur Verfügung stehen.

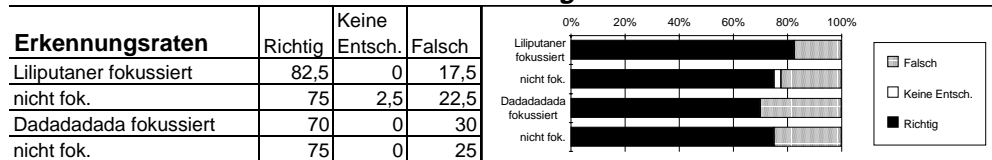
Wie für die oben beschriebenen Lernreihen werden auch hier die Daten in fünf Teile aufgeteilt, um getrennte Lern- und Testmengen zu erhalten. Zudem werden hier zwei verschiedene Zufallspartitionierungen untersucht, künftig Reihe 1 und Reihe 2 genannt, um auch den Einfluß der Partitionierung auf die Ergebnisse beobachten zu können.

Die Erkennungsraten für die Läufe sind in Abbildung 3.4 und Abbildung 3.5 abgedruckt. Für alle Läufe in beiden Reihen kann man sagen, daß das zusätzliche Anbieten der Differenzmerkmale wesentlich zur Verbesserung der Erkennungsraten beigetragen hat. Faßt man die Erkennungsraten von „fokussiert“ und „nicht fokussiert“ jeweils durch Mittelwertbildung zusammen und vergleicht sie mit den im vorigen Kapitel in der sechsten Lernreihe gezeigten Erkennungsraten, so stehen bei Liliputaner hier 78.75% (Reihe 1) bzw. 72.5% (Reihe 2) den dort erzielten 44% gegenüber. Bei den Dadadadada-Silben ist der Unterschied noch drastischer, denn hier stehen 72,5% bzw. 70% den dort erzielten 24% gegenüber. Dabei ist zu bedenken, daß sich durch die Betrachtung von nur vier Silbenpaaren gegenüber den fünf Silben im vorherigen Abschnitt, die durch den trivialen Klassifikator erzielbaren Erkennungsraten von 80% auf 75% verringert haben. Daraus folgt, daß die rein silbenbezogenen Grundfrequenzmerkmale alleine nicht geeignet sind, Betonung vorherzusagen, daß diese Merkmale zusammen mit den silbenübergreifenden Differenzmerkmalen jedoch immerhin bis auf die Erkennungsrate eines trivialen Klassifikators kommen.

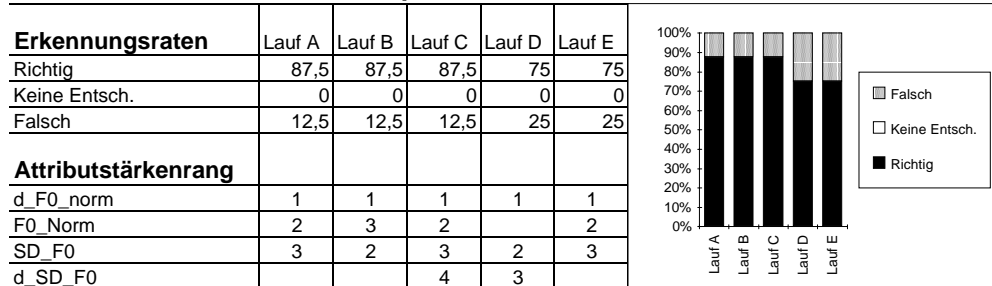
Wenn man die Attributstärke betrachtet, also die Wichtigkeit, die der Lernalgorithmus den einzelnen Attributen zuerkennt, fällt auf, daß das Merkmal **d_F0_Norm** bei den Liliputaner-Silben wohl eine sehr wichtige Rolle gespielt hat, während es bei den Dadadadada-Silben weniger bedeutend ist. Auch bei den Erkennungsraten zeigen sich größere Unterschiede zwischen Liliputaner- und Dadadadada-Läufen als zwischen den Läufen, bei denen das untersuchte Wort an fokussierter bzw. nicht fokussierter Stelle stand. So sind die Erkennungsraten bei den Liliputaner-Läufen an hauptbetonter Stelle tatsächlich im Mittel besser als an nicht fokussierter Stelle, bei den Dadadadada-Läufen wird dagegen die Betonung an nicht fokussierter Stelle besser erkannt als an fokussierter Stelle, bei der zweiten Reihe sogar sehr deutlich. Für die These, daß der Akzent an Fokusposition meist durch Pitchakzent, an Nichtfokusposition bevorzugt durch Längung (und/oder Lautstärkeänderung) realisiert wird, können also in den Reihen keine unterstützenden Belege gefunden werden.

Da die Unterschiede in den Erkennungsraten nicht sehr groß sind, wurde die-

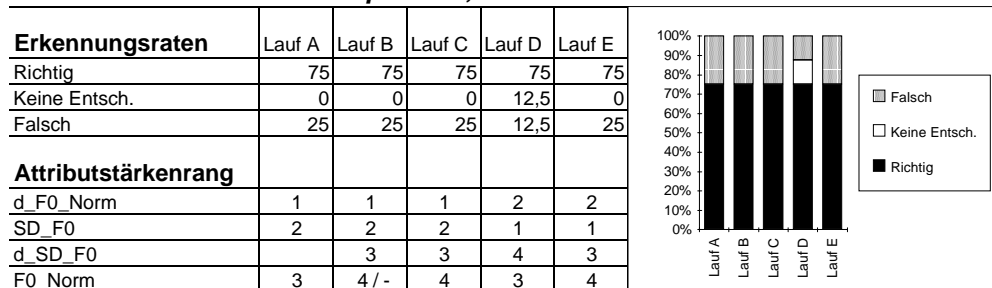
Zusammenfassung Reihe 1



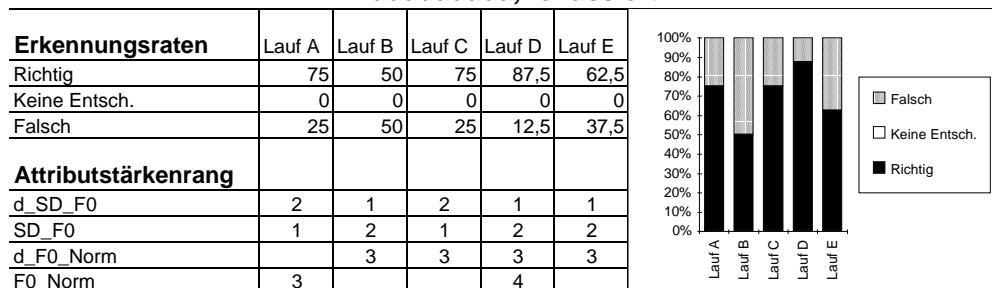
Liliputaner, fokussiert



Liliputaner, nicht fokussiert



Dadadadada, fokussiert



Dadadadada, nicht fokussiert

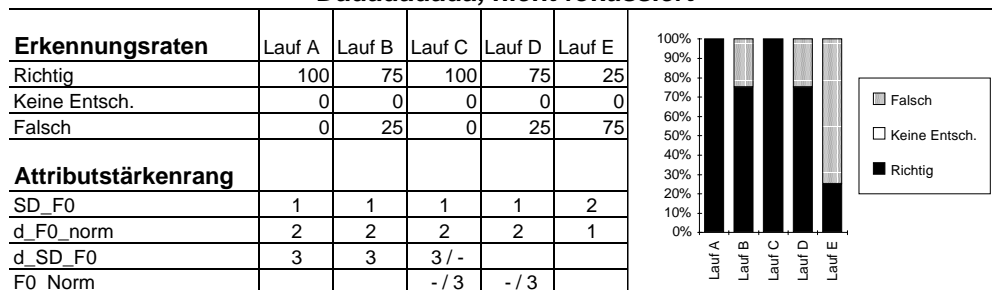


Abbildung 3.4: Erkennungsraten und Attributstärkenrang für die silbenbezogenen und silbenübergreifenden Grundfrequenzmerkmale (F0_Norm, SD_F0, d_F0_Norm und d_SD_F0) in der ersten der beiden durchgeführten Lernreihen.

Zusammenfassung Reihe 2

Erkennungsraten	Richtig	Keine Entsch.	Falsch
Liliputaner fokussiert	75	0	25
nicht fok.	70	0	30
Dadadadada fokussiert	55	2,5	42,5
nicht fok.	85	0	15

Liliputaner, fokussiert

Erkennungsraten	Lauf A	Lauf B	Lauf C	Lauf D	Lauf E
Richtig	75	75	87,5	75	62,5
Keine Entsch.	0	0	0	0	0
Falsch	25	25	12,5	25	37,5

Attributstärkenrang	Lauf A	Lauf B	Lauf C	Lauf D	Lauf E
d_F0_norm	1	1	1	2	1
F0_Norm	2	1	2		2
d_SD_F0	3	4	4	3	3
SD_F0		3	3	1	

Liliputaner, nicht fokussiert

Erkennungsraten	Lauf A	Lauf B	Lauf C	Lauf D	Lauf E
Richtig	87,5	50	75	62,5	75
Keine Entsch.	0	0	0	0	0
Falsch	12,5	50	25	37,5	25

Attributstärkenrang	Lauf A	Lauf B	Lauf C	Lauf D	Lauf E
SD_F0	1	1	2	2 / 1	1
d_F0_Norm	2	3	1	1 / 2	1
d_SD_F0	3	4	3	3	4
F0_Norm	4	1	4 / -	4	3

Dadadadada, fokussiert

Erkennungsraten	Lauf A	Lauf B	Lauf C	Lauf D	Lauf E
Richtig	50	75	50	37,5	62,5
Keine Entsch.	12,5	0	0	0	0
Falsch	37,5	25	50	62,5	37,5

Attributstärkenrang	Lauf A	Lauf B	Lauf C	Lauf D	Lauf E
d_SD_F0	2	1	1	2	1
SD_F0	1	2	2	1	2
d_F0_Norm	3	3	3		3
F0_Norm		4			

Dadadadada, nicht fokussiert

Erkennungsraten	Lauf A	Lauf B	Lauf C	Lauf D	Lauf E
Richtig	62,5	100	87,5	75	100
Keine Entsch.	0	0	0	0	0
Falsch	37,5	0	12,5	25	0

Attributstärkenrang	Lauf A	Lauf B	Lauf C	Lauf D	Lauf E
SD_F0	2	1	1	1	1
d_F0_norm	1	2	2	- / 2	2
d_SD_F0	3	3	3		3
F0_Norm				- / 3	

Abbildung 3.5: Erkennungsraten und Attributstärkenrang für die silbenbezogenen und silbenübergreifenden Grundfrequenzmerkmale (F0_Norm, SD_F0, d_F0_Norm und d_SD_F0) in der ersten der beiden durchgeführten Lernreihen.

ser Aspekt mit Hilfe der beurteilenden Statistik genauer untersucht. Man kann das Testen eines Beispiels aus der Testmenge als ein Zufallsexperiment mit den zwei Ausgängen *richtig vorhergesagt* und *nicht richtig vorhergesagt* auffassen (Bernoulliexperiment). Führt man das Experiment unabhängig mehrmals hintereinander aus, so ist die Anzahl der richtig vorhergesagten Ausgänge binomialverteilt und damit näherungsweise normalverteilt. Daher können die in den Läufen ermittelten Erkennungsraten aufgefaßt werden als Realisierungen einer näherungsweise normalverteilten Zufallsvariable. Man kann damit testen, ob beispielsweise die Erkennungsraten bei den Silben von Liliputaner an fokussierter Stelle signifikant besser sind als an nicht fokussierter Stelle, oder ob die Unterschiede nur im Zufallsbereich liegen.

Die Ergebnisse der durchgeführten Analysen, die in Abb. 3.6 abgedruckt sind, lassen sich wie folgt zusammenfassen: Bei der Reihe 1 kann mit einer Irrtumswahrscheinlichkeit von 0.05 die Hypothese verworfen werden, daß die Silben von Liliputaner an nicht fokussierter Stelle besser oder gleich gut vorhergesagt werden können als an fokussierter Stelle. Für Reihe 2 kann die Hypothese mit dieser Irrtumswahrscheinlichkeit nicht verworfen werden (lediglich mit einer Irrtumswahrscheinlichkeit von 0.30). Betrachtet man die Daten der beiden Reihen gemeinsam, so kann man die Hypothese mit einer Irrtumswahrscheinlichkeit von 0.10 ablehnen (Abb. 3.7), für eine Irrtumswahrscheinlichkeit von 0.05 darf sie jedoch nicht abgelehnt werden (das 0.95%-Quantil der t-Verteilung mit 18 Freiheitsgraden beträgt 1.734).

Für die Silben der Logatomisierung kann man in Reihe 1 die Hypothese, an Fokusposition werde besser oder gleichgut vorhergesagt als an Nichtfokusposition, mit der Irrtumswahrscheinlichkeit von 0.05 nicht verwerfen. Für Reihe 2 gelingt dies jedoch; auch bei der Betrachtung beider Reihen gleichzeitig kann man die Hypothese mit der Irrtumswahrscheinlichkeit 0.05 ablehnen.

Das Ergebnis bleibt also auch nach der statistischen Analyse widersprüchlich: Die Liliputansilben können zwar mit den hier untersuchten Grundfrequenzmerkmalen geringfügig besser an Fokusposition vorhergesagt werden, für die Silben der Logatomisierung kann jedoch das Gegenteil gezeigt werden.

3.6 Zusammenfassung und Bewertung

Die ermittelten Erkennungsraten der Untersuchungen aus diesem Kapitel sind in Abbildung 3.8 noch einmal zusammengefaßt. Dargestellt sind die über die Läufe gemittelten Erkennungsraten. Bei den Untersuchungen mit den Differenzmerkmalen wurde zusätzlich über die jeweils vier Ergebnisse (fokussiert / nicht fokussiert, zwei Durchgänge) gemittelt. Die Untersuchungen haben gezeigt, daß eindeutig die Silbenlänge der beste Indikator für betonte Silben ist, unabhängig davon, ob die Silbe in Fokusposition steht oder nicht. Außerdem hat sich gezeigt, daß die Einflüsse der beteiligten Segmente, speziell bei den Lautstärkemerkmale sehr groß sind und die Erkennung der rein prosodischen Unterschiede erschweren. Durch parallele Untersuchungen mit einem Logatom,

Vergleich der Erwartungswerte

Liliputaner, Reihe 1

	n_i	Daten					Mittelwert	Varianz
Liliputaner, fokussiert	5	87,5	87,5	87,5	75	75	$\mu_1 = 82,5$	46,875
Liliputaner, nicht fokussiert	5	75	75	75	75	75	$\mu_2 = 75$	0

berechnetes t: 2,44948974
 Anzahl Freiheitsgrade: 8
 nachgeschlagenes 0,95-Quantil dazu: 1,86
 Irrtumswahrschk: := 0,05
H0 kann abgelehnt werden

H2: $\mu_1 > \mu_2$
 H3: $\mu_1 < \mu_2$
 Irrtumswahrschk: := 0,05
H2 darf nicht abgelehnt werden

Liliputaner, Reihe 2

	n_i	Daten					Mittelwert	Varianz
Liliputaner, fokussiert	5	75	75	87,5	75	62,5	$\mu_1 = 75$	78,125
Liliputaner, nicht fokussiert	5	87,5	50	75	62,5	75	$\mu_2 = 70$	203,125

berechnetes t: 0,66666667
 Anzahl Freiheitsgrade: 8
 nachgeschlagenes 0,95-Quantil dazu: 1,86
 Irrtumswahrschk: := 0,05
H0 darf nicht abgelehnt werden

H2: $\mu_1 > \mu_2$
 H3: $\mu_1 < \mu_2$
 Irrtumswahrschk: := 0,05
H2 darf nicht abgelehnt werden

Dadadadada, Reihe 1

	n_i	Daten					Mittelwert	Varianz
Dadadadada, fokussiert	5	75	50	75	87,5	62,5	$\mu_1 = 70$	203,125
Dadadadada, nicht fokussiert	5	100	75	100	75	25	$\mu_2 = 75$	937,5

berechnetes t: -0,33104236
 Anzahl Freiheitsgrade: 8
 nachgeschlagenes 0,95-Quantil dazu: 1,86
 Irrtumswahrschk: := 0,05
H0 darf nicht abgelehnt werden

H2: $\mu_1 > \mu_2$
 H3: $\mu_1 < \mu_2$
 Irrtumswahrschk: := 0,05
H2 darf nicht abgelehnt werden

Dadadadada, Reihe 2

	n_i	Daten					Mittelwert	Varianz
Dadadadada, fokussiert	5	50	75	50	37,5	62,5	$\mu_1 = 55$	203,125
Dadadadada, nicht fokussiert	5	62,5	100	87,5	75	100	$\mu_2 = 85$	265,625

berechnetes t: -3,09838668
 Anzahl Freiheitsgrade: 8
 nachgeschlagenes 0,95-Quantil dazu: 1,86
 Irrtumswahrschk: := 0,05
H0 darf nicht abgelehnt werden

H2: $\mu_1 > \mu_2$
 H3: $\mu_1 < \mu_2$
 Irrtumswahrschk: := 0,05
H2 kann abgelehnt werden

Abbildung 3.6: Signifikanztests für die Erkennungsraten der silbenbezogenen und silbenübergreifenden Grundfrequenzmerkmale. Die Erkennungsraten sind bei den Silben des Wortes Liliputaner an Fokusposition in einer der beiden Reihen signifikant besser (Reihe 1). Bei den Silben von Dadadadada sind die Erkennungsraten an Fokusposition jedoch in einer Reihe signifikant schlechter (Reihe 2). Für die jeweils verbleibenden Reihen sind die Unterschiede nicht signifikant. Diese Aussagen gelten für eine Irrtumswahrscheinlichkeit von 0,05 und 0,10.

Liliputaner, Reihe 1 und Reihe 2

	n	i	Daten Reihe 1					Daten Reihe 2					Mittelwert	Varianz
Liliputaner, fokussiert	10	87,5	87,5	87,5	75	75	75	75	87,5	75	62,5	$\mu_1 =$	78,75	71,1805556
Liliputaner, nicht fokussiert	10	75	75	75	75	75	75	87,5	50	75	62,5	$\mu_2 =$	72,5	97,2222222

$$H_0: \mu_1 \leq \mu_2$$

H1: $\mu_1 > \mu_2$

Irrtumswahrschkt := 0,10

berechnetes t: 1,52301925

Anzahl Freiheitsgrade: 18

nachgeschlagenes 0,90-Quantil dazu: 1,33

H0 kann abgelehnt werden

$$H_2: \mu_1 \geq \mu_2$$
$$H_3: \mu_1 < \mu_2$$

Irrtumswahrschkt := 0,10

H2 darf nicht abgelehnt werden

Dadadadada, Reihe 1 und Reihe 2

	n _i	Daten Reihe 1				Daten Reihe 2				Mittelwert	Varianz	
Dadadadada, fokussiert	10	75	50	75	87,5	62,5	50	75	50	37,5	62,5	243,055556
Dadadadada, nicht fokussiert	10	100	75	100	75	25	62,5	100	87,5	75	100	562,5

$$H_0: \mu_1 \leq \mu_2$$
$$H_1: \mu_1 > \mu_2$$

Irrtumswahrschkt := 0,10

berechnetes t: -1,94980105

Anzahl Freiheitsgrade: 18

nachgeschlagenes 0,90-Quantil dazu: 1,33

H0 darf nicht abgelehnt werden

$$H_2: \mu_1 \geq \mu_2$$

H3: $\mu_1 < \mu_2$

Irrtumswahrschkt := 0,10

H2 kann abgelehnt werden

Abbildung 3.7: Signifikanztests für die Erkennungsraten der silbenbezogenen und silbenübergreifenden Grundfrequenzmerkmale beider Lernreihen zusammen. Die Erkennungsraten bei den Silben des Wortes Liliputaner sind an Fokusposition signifikant besser als an Nichtfokusposition. Beim Wort Dadadadada sind die Erkennungsraten an Fokusposition jedoch signifikant schlechter. Die zugrundeliegende Irrtumswahrscheinlichkeit beträgt 0,1. Für eine Irrtumswahrscheinlichkeit von 0,05 sind lediglich die Unterschiede bei den Dadadadada-Silben signifikant.

Liliputaner								Dadadadada							
1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
⊕	+	+	+	+	+			⊕	+	+	+	+	+		
	+	+	⊕	⊕	+				⊕	⊕	⊕	⊕	+		
			⊕	⊕							⊕	⊕			
	⊕	⊕		⊕					⊕	⊕		⊕			
	⊕	⊕	⊕						⊕	⊕	⊕				
	⊕	⊕							⊕	⊕					
				⊕								⊕			
			⊕								⊕				
⊕	⊕							⊕	⊕						
⊕	⊕				⊕	⊕		⊕	⊕					⊕	⊕

1 = Dur_Norm, 2 = F0_Norm, 3 = SD_F0, 4 = dB_Norm, 5 = SD_dB, 6 = Focus, 7 = d_F0_Norm, 8 = d_SD_F0.

+ = Merkmal angeboten, ⊕ = Merkmal verwendet

■ = richtig vorhergesagt, ■ = keine Entscheidung, ■ = falsch vorhergesagt.

Abbildung 3.8: Die ermittelten Erkennungsraten im Überblick.

bei dem alle Silben die gleiche segmentale Struktur aufweisen, konnten die unerwünschten Einflüsse der Segmente weitgehend ausgeschaltet werden. Bei den Logatomsilben ermöglicht allein die Verwendung des Längenmerkmals einen Klassifikator zwischen betonten und unbetonten Silben, der besser als der triviale Klassifikator „jede Silbe ist unbetont“ abschneidet. Akzeptabel sind für die oberen acht Reihen Erkennungsraten über 80%, für die unterste Reihe über 75%. Die demnach ebenfalls akzeptablen Erkennungsraten einiger Lernreihen bei den Liliputansilben sind aber wie oben beschrieben auf das Erkennen der Segmente und nicht auf das Erkennen einer Betonung zurückzuführen. Die Grundfrequenzmerkmale ermöglichen nur sehr schlechte Klassifikatoren. Lediglich die in der untersten Reihe abgebildeten Ergebnisse der Lernreihen mit den Grundfrequenz-Differenzmerkmalen sind mit den Leistungen eines trivialen Klassifikators vergleichbar. In der Erkennungsleistung stehen sie somit zwar deutlich hinter dem Längenmerkmal zurück, liefern jedoch tatsächlich die zweitbesten Erkennungsraten, da sämtliche Lernreihen der durch den Spracherkennungseffekt nicht betroffenen Da-Silben nicht an die Leistungen eines trivialen Klassifikators (dort 80%) heranreichen.

Das schlechte Abschneiden der Grundfrequenzmerkmale kann zum Teil durch die Art der verwendeten Merkmale begründet werden. Wenn man die Pitchakzente adäquat repräsentieren möchte, scheint mir eine bessere zeitliche Strukturierung als durch silbenbezogene Merkmale unumgänglich. Unter Umständen könnte hier schon die bereits erwähnte getrennte Erfassung der Merkmale für Onset und Rhyme eine Verbesserung erbringen. Auch die Aufnahme von Merkmalen der ersten (und evtl. der zweiten) Ableitung des geglätteten Grundfrequenzverlaufs scheint mir ein geeignetes Mittel zu sein, den kontinuierlichen Charakter des Pitchakzents angemessener zu repräsentieren als durch Mittelwert und Standardabweichung. Auch die Extraktion von Minimal- und Maximalwerten oder deren Differenz könnten bessere Merkmale ergeben, wenngleich natürlich auch Verbindungen zwischen überstrichenem Wertebereich und Standardabweichung bestehen. Wenn man den Rahmen einer merkmalsbasierten Untersuchung erweitern möchte, bietet sich auch eine Modellierung durch Hidden-Markov-Modelle (HMMs) an [Rabi89]. HMMs haben sich beispielsweise bei der Erkennung kontinuierlich gesprochener Sprache in den letzten Jahren als sehr nützlich erwiesen beim Übergang von der kontinuierlichen Signal- zur kategorialen Symbolebene. Als Merkmalswert könnte in eine merkmalsbasierte Untersuchung die Wahrscheinlichkeit eingehen, mit welcher ein HMM für betonte Silben den Grundfrequenzverlauf der unmittelbaren Umgebung der Silbe emittiert.

Kapitel 4

Phonologische Betonungsregeln des Deutschen

Im bisherigen Verlauf dieser Arbeit wurde die Frage untersucht, wie Wortbetonung im Deutschen phonetisch realisiert wird, also, inwiefern sich eine betonte Silbe von einer unbetonten Silbe unterscheidet. Dieses Kapitel befaßt sich nun mit der Fragestellung, auf welcher Stelle innerhalb eines Wortes die Betonung liegt. Da eine allgemeine Untersuchung für eine solche Arbeit zu umfangreich ist, wird zunächst das Untersuchungsgebiet näher eingegrenzt. Danach wird erläutert, welche Daten für die Untersuchung herangezogen, und wie sie dem gewählten Lernverfahren zugänglich gemacht werden können. Anschließend werden die in den Lernreihen gewonnenen Ergebnisse vorgestellt und diskutiert. In einem abschließenden Abschnitt werden die durch das Lernverfahren erzeugten Regeln mit verbreiteten Annahmen von Phonologen in Beziehung gesetzt.

4.1 Untersuchungsgebiet

Untersucht werden morphologisch einfache Wörter. Darunter versteht man nach [Jess94] Wörter, die entweder nur aus einem Morphem bestehen oder aus monomorphemischen Wörtern durch nicht-native Suffixe abgeleitet sind. Die Aufgabe ist, aus Merkmalen, die den Aufbau der Silben beschreiben, die Stelle des Wortakzents vorherzusagen. Instrument der Untersuchung ist dabei wieder das Rough Sets basierte Lernverfahren. Durch Betrachtung der produzierten Regeln sowie Vergleich der Vorhersagegenauigkeit verschiedener Codierungen können Rückschlüsse auf phonologisch interessante Fragestellungen gezogen werden.

Wenn man sich mit Untersuchungen über den deutschen Wortakzent morphologisch einfacher Wörter befaßt, muß man sich im klaren sein, daß es für die überwiegende Zahl von Wörtern eigentlich nichts zu untersuchen gibt. Ein beträchtlicher Teil der morphologisch einfachen Wörter besteht entweder aus einer Silbe oder aus zwei Silben, wovon eine einen Schwa als Nukleus besitzt. Da

Schwasilben niemals betont werden, steht in solchen Wörtern von vornherein fest, welche Silbe die Wortbetonung erhält. In der in Anhang A beschriebenen Abschätzung habe ich festgestellt, das etwa 45% der Wörter aus einem Zeitungstext „wahrscheinlich einsilbig“ sind. Etwa 74% der Wörter besitzen „wahrscheinlich höchstens zwei Silben“. Außerdem muß man bedenken, daß sich im untersuchten Text auch viele zusammengesetzte Wörter befinden, die zudem verhältnismäßig lang sind. Es ist deshalb klar, daß morphologisch einfache Wörter, für die die Betonungszuweisung ein Problem darstellt, eher selten auftreten.

4.2 Datenmaterial und -aufbereitung

4.2.1 Auswahl eines Korpus

Bei Beginn der Arbeit standen dem Lehrstuhl keine geeigneten maschinenlesbaren und phonetisch transkribierten Daten zur Verfügung. Da eine Erfassung der Trainings- und Testdaten von Hand unumgänglich erschien, mußten Einschränkungen bezüglich des abgedeckten Bereichs getroffen werden. Nicht abgedeckt sind aus diesem Grund die zwar uninteressanten, aber für eine gute Vorhersagegenauigkeit aufgrund ihrer hohen Frequenz wichtigen Wörter, die weniger als zwei volle, d. h. nicht-Schwa-Vokale aufweisen. In dieser Untersuchung ist man jedoch nicht in erster Linie an besonders guten Erkennungsraten interessiert, sondern mehr an der Frage der Anwendbarkeit von Algorithmen des Maschinellen Lernens auf phonetisch/phonologische Fragestellungen. Wenn man maschinell gewonnene Regeln, z. B. bei Text-to-Speech-Anwendungen, einsetzen möchte, ist auch eine zweistufige Architektur denkbar: In einer ersten Stufe werden die trivialen Fälle abgehandelt, also morphologisch einfache Wörter, die nur einen vollen Vokal aufweisen. An die zweite Stufe werden die morphologisch einfachen Wörter weitergeleitet, die auch Gegenstand dieser Untersuchung sind. Durch dieses Vorgehen wird der Lernalgorithmus gewissermaßen gezwungen, auch die seltener auftretenden, schwierigen Fälle adäquat zu behandeln. Für eine praktisch einsetzbare Anwendung fehlt schließlich noch eine morphologische Komponente, die zusammengesetzte Wörter zerlegt, sowie Regeln, die die Gewichtung der gefundenen Akzente der Teile für das ganze Wort festlegen.

Eine weitere Einschränkung betrifft die syntaktische Kategorie der in das Korpus aufgenommenen Wörter. Um die Kandidatenanzahl einzuschränken wurden nur Nomen untersucht. Diese Einschränkung auf Nomen ist sinnvoll, da für diese Wortklasse die größte Varianz von Betonungsmustern berichtet wird [Eise91, Benw80].

Als Ausgangspunkt für das Korpus wurde die Liste der benutzten Wörter eines Buches über den deutschen Wortakzent verwendet [Gieg85]. Die dort aufgeführten Einträge werden als Beispiele und Gegenbeispiele für Regeln zur Betonungsvorhersage verwendet. Man kann demnach davon ausgehen, daß die Daten relevant und prototypisch für die gestellte Aufgabe sind. Da Giegerich auch viele Ausnahmen seiner Regeln aufführt, ist eine zu starke Orientierung der Daten

zugunsten Giegerichs Analysen nicht zu vermuten. Die Wortliste umfaßt 668 Wörter, von denen 242 den oben erwähnten Kriterien genügen.

4.2.2 Transkription der Wörter

Die phonetische Transkription der ausgewählten Wörter wurde in [Mang90] nachgeschlagen. Darin waren drei Wörter aus der Wortliste, Epitaxis, Kanut und Platonik, nicht verzeichnet. Sie wurden deshalb aus dem Korpus entfernt. In [Mang90] sind gelegentlich Aussprachevarianten aufgeführt. Diese wurden aufgenommen, außer wenn sie als fremdsprachlich gekennzeichnet waren. Dabei ist anzumerken, daß die Unterscheidung in fremdsprachlich oder nicht fremdsprachlich in [Mang90] m. E. zumindest fragwürdig scheint. Beispielsweise ist einerseits „Establishment“ als englischsprachig gekennzeichnet, während andererseits bei „Arrangement“ eine Kennzeichnung fehlt, obwohl das Wort in französischer Aussprache mit Nasalvokalen aufgeführt ist. Da eine von [Mang90] abweichende Unterscheidung eine gewissenhafte Untersuchung aller Einzelfälle erfordern würde, habe ich die Eintragung von Mangold als Kriterium für die Auswahl belassen.

Als Notation für die Weiterverarbeitung wurde eine leicht abgeänderte Version von SAM-PA [Four&al89] verwendet. SAM-PA ist eine Abbildung des internationalen phonetischen Alphabets (IPA) auf 7-Bit ASCII-Zeichen. SAM-PA wurde ausgewählt, da es problemlos weiterzuverarbeiten ist. Die Repräsentation von Diphthongen, Nasalvokalen, silbischen Konsonanten und Affrikaten als 2-Zeichensequenzen stellt kein Problem dar, da sie eindeutig analysiert werden können. Für unsilbische Vokale ist in SAM-PA keine Schreibweise festgelegt, jedoch von Mangold transkribiert. Ich habe darum die Notation um das Prefix /~/ erweitert, da dieses Zeichen in SAM-PA noch nicht belegt zu sein scheint und es an die in [Mang90] verwendete Schreibweise erinnert. Beispiele dafür sind: Altar – /a1|'ta:~6/, Chiasmus – /'C~ias|mUs/, Fraktion – /frak|'ts~io:n/ und Pointe – /'p~oE~:|t@/.

Wie die angeführten Beispiele zeigen, wurde auch Information über Silbengrenzen in die Transkription aufgenommen. Silbengrenzen sind in [Mang90] nur vor betonten Silben indirekt wiedergegeben. Das Betonungszeichen /'/ steht immer vor der betonten Silbe, dadurch ist die vordere Grenze einer betonten Silbe also bekannt. Die restlichen Grenzen wurden von mir eingefügt, bei unklaren Fällen wurde zusammen mit Michael Jessen entschieden. Besondere Beachtung fanden dabei Segmente, die in der phonologischen Literatur als „ambisyllabisch“ [ClemKeys83, S.58] oder „scharf geschnitten“ [Venn91] bezeichnet sind. Bei diesen Segmenten geht man davon aus, daß ein zwischen zwei Nuklei liegender Konsonant sowohl zur Coda der vorangehenden Silbe als auch zum Onset der folgenden Silbe gerechnet werden muß, obwohl er tatsächlich meßbar nur einmal realisiert wird. Beispiele dafür sind /k/ in „Akku“ und /m/ in „Kommando“. Bei der Betrachtung des Korpus konnte für die untersuchten Daten ein einheitliches Kriterium für die Ambisyllabizität gefunden werden, das jedoch keinen universalen Anspruch hat. Ich bezeichne im folgenden eine Silbe, deren Coda nur aus einem ambisyllabischen Segment besteht,

als ambivalent. Als ambivalent betrachtet werden Silben, wenn die darauffolgende Silbe orthographisch mit *bb*, *ck*, *ff*, *kk*, *ll*, *mm*, *nn*, *pp*, *rr*, *ss*, *tt* oder *x* beginnt. In der SAM-PA-Notation habe ich dies repräsentiert, indem vor der Silbengrenze ein *_* und nach der Silbengrenze das Segment selbst steht, beispielsweise /'a_*_*ku/, /k0_*_*'man|do/. Der Unterstrich ist dabei ein weiteres Symbol, das nicht in der ursprünglichen Notation von SAM-PA enthalten ist. Beachtenswert ist, daß außer bei *x* möglicherweise auch die orthographischen Trennungsregeln einen Hinweis auf die Ambisyllabizität geben können (vgl. [Rame92]). Innerhalb von /*ks*/, den Segmenten, die dem orthographischen *x* entsprechen, kann keine Silbengrenze gesetzt werden, so daß also das Segment /*k*/ der vorangehenden Silbe zugerechnet würde und /*s*/ der nachfolgenden. Solch ein Vorgehen würde klar der Transkription von [Mang90] widersprechen, wie beispielsweise „Alexandra“ /a*l*E'*ks*andra/ zeigt. In fremdsprachlichen Einträgen hingegen (bes. Russisch und Rumänisch) verwendet Mangold die Trennung zwischen den Segmenten, z. B. „Alexandr“ /a*l*Ik'*s*andr/. Man kann daraus keine zwingende Notwendigkeit für Ambisyllabizität ableiten. In dem hier verfolgten Rahmen schadet es aber nicht, diese möglicherweise zu feine Untergliederung von Merkmalen dem Lernalgorithmus anzubieten. Sollte die Ambisyllabizität im Deutschen irrelevant sein (s. z. B. [Hall92]), ist es dem Algorithmus unbenommen, die ambivalenten Silben in den Regeln den offenen Silben zuzuschlagen. Wie wir jedoch noch sehen werden, benutzte der Algorithmus für die Coda-Merkmale die Ausprägung *ambiv.* weitaus häufiger zusammen mit den Ausprägungen *geschl.* und *m.geschl.* als mit *offen.* Dies ist eine von den Ergebnissen ableitbare Rechtfertigung für das gewählte Vorgehen.

4.2.3 Umsetzung in Merkmale

Das symbolische Lernverfahren erwartet als Eingabe Trainingsbeispiele, die durch Attribute beschrieben sind. Die Attribute sollen die Struktur der Silben beschreiben. Die Auswahl der Attribute orientiert sich dabei an den Kriterien, die auch in der phonologischen Literatur allgemein als für die Betonungszuweisung wichtig angesehen werden.

Eines dieser Merkmale ist der Aufbau der Coda der Silbe, also die Anzahl der Konsonanten, die auf den Nukleusvokal folgen. Ein weiteres relevantes Beschreibungsmerkmal ist die Qualität des Nukleusvokals, also im wesentlichen die Unterscheidung zwischen gespannten und ungespannten Vokalen. Auch die Unterscheidung zwischen Schwa und den anderen Vokalen ist relevant, da Schwas keine Betonung tragen können. Schließlich scheint auch die Information, ob der Nukleus aus einem Diphtong besteht, relevant zu sein. Die Höhe des Nukleus wird häufig nicht als relevant eingestuft, andere sehen die Höhe in bestimmten Situationen als unterscheidend an [Dael&al93, Féry86, Benw80]. Für Schwa-Silben wird manchmal ein unterschiedlicher Einfluß angenommen, je nach dem, ob die Silbe über einen Onset verfügt oder nicht [Venn92]. Zusammenfassend kann man sagen, daß für die Betonungszuweisung folgende Merkmale als relevant betrachtet werden: Vorhandensein eines Onsets, Gespanntheit des Nu-

kleusvokals, Kennzeichnung von Schwa und Diphtongen sowie die Anzahl der Segmente in der Coda.

Die Länge des Nukleusvokals ist als Beschreibungsmerkmal nicht unproblematisch, obwohl sie prinzipiell geeignet ist, einen wichtigen Aspekt der Silbenstruktur zu beschreiben. Wie im Kapitel über die akustischen Korrelate der Betonung gezeigt, ist die phonetisch realisierte Länge der Silbe unter den untersuchten Merkmalen der beste Indikator für die Unterscheidung zwischen betonten und unbetonten Silben.

Die angesprochenen Merkmale werden in der Phonologie jedoch meist nicht unabhängig voneinander betrachtet. Stattdessen werden Silben zu bestimmten Silbenklassen zusammengefaßt. In diese Klassenbildung gehen dabei unter Umständen mehrere der oben beschriebenen Merkmale ein.

Die Extraktion der Merkmale aus der Lautschrift erfolgt maschinell durch ein PROLOG-Program. Dabei werden die Merkmale in einem zweistufigen Prozeß extrahiert. Zunächst werden für jede Silbe die Merkmale **Onset**, **Hoehe**, **Gespanntheit**, **Laenge** und **Coda** bestimmt, anschließend wird aus diesen das Merkmal **Typ** berechnet. Die Merkmale und ihre möglichen Werte sind in Tafel 4.1 wiedergegeben. Die Liste der Wörter und ihrer Transkriptionen wird von einer Definite Clause Grammar abgearbeitet, und die Merkmale der ersten Phase werden durch eingebettete semantische Aktionen extrahiert. In der zweiten Phase erfolgt dann die Abbildung der unabhängigen Merkmale auf die phonologischen Klassen.

Die Anwendung auf die Umschrift des gewählten Korpus ergibt für jedes Wort 36 Merkmale, da das längste Wort des Korpus 6 Silben umfaßt und pro Silbe 6 Merkmale extrahiert werden. Die Merkmale werden von rechts nach links ausgerichtet. Dies entspricht der in der Phonologie verbreiteten Betrachtungsweise. Demnach sind im ersten Merkmal jeder Merkmalsart die Eigenschaften der letzten Silbe festgehalten, im zweiten Merkmal die der vorletzten Silbe und so fort. Da das Lernprogramm in der mir vorliegenden Version nicht mit fehlenden Merkmalswerten umgehen kann, wird für die Merkmale der über die Wortlänge hinausgehenden Silben der Wert - eingeführt. Dadurch ist es dem Algorithmus möglich, eine Codierung der Silbenzahl in die Regeln einzubauen.

Als Entscheidungsattribut wird die Stelle der Betonung aus der Transkription extrahiert. Es kann die Werte **final**, **penult** oder **antepenult** annehmen, die einer Betonung der letzten, vorletzten bzw. vorvorletzten Silbe entsprechen. Die numerische Verteilung der zu lernenden Konzepte ist wie folgt: 113 final betonte Fälle, 107 penult betonte Fälle und 40 antepenult betonte Fälle.

4.3 Lernreihen

Mit den Korpusdaten wurden drei Lernreihen durchgeführt. Die ersten beiden Reihen dienen zur Überprüfung zweier Annahmen, die die meisten Phonologen in ihren Analysen unterstellen. In der Literatur herrscht große Übereinstimmung darüber, daß für die Betonungszuweisung in morphologisch einfachen

1. Typ	Phonologischer Typ der Silbe
@	Schwa-Silbe mit leerem Onset
C@	Schwa-Silbe mit vorhandenem Onset
Vu	Silbe mit ungespanntem Vokal und leerer Coda
Vu(C)	Silbe mit ungespanntem Vokal und ambisyllabischer Coda
VuC	Silbe mit ungespanntem Vokal und einem Segment in der Coda
VuCC1	Silbe mit ungespanntem Vokal und mehr als einem Segment in der Coda
V_V	Silbe mit Diphtong als Nukleus
Vg	Silbe mit gespanntem Vokal und leerer Coda
VgC1	Silbe mit gespanntem Vokal und mindestens einem Segment in der Coda
konson.	Silbe mit silbischem Konsonanten als Nukleus
2. Onset	Vorhandensein eines Onsets
vorh.	vorhandener Onset
n.vorh.	leerer Onset
3. Hoehe	Höhe des Nukleusvokals
hoch	hoher Vokal (/I/, /U/, /Y/, /i/, /u/, /y/)
mittel	mittelhoher Vokal (/E/, /O/, /9/, /e/, /o/, /2/ sowie die beiden Schwa /@/ and /6/)
tief	tiefer Vokal (/a/)
steigend	Diphtong (/ai/, /au/, /0y/)
konson.	silbischer Konsonant als Nukleus
4. Laenge	Länge des Nukleusvokals
lang	in der Lautschrift mit : markiert oder Diphtong
kurz	sonst
5. Gespanntheit	Gespanntheit des Nukleusvokals
gespannt	gespannter Vokal (/e/, /o/, /2/, /i/, /u/, /y/)
ungesp.	ungespannter Vokal (/a/, /E/, /O/, /9/, /I/, /U/, /Y/ und die Diphtonge /ai/, /au/, /0y/)
Schwa	Schwa (/@/, /6/)
konson.	silbischer Konsonant als Nukleus
6. Coda	Anzahl der Segmente in der Coda
offen	leere Coda
ambiv.	ambisyllabische Coda
einf.g.	Coda besteht aus einem Segment
mehrf.g.	Coda besteht aus mehreren Segmenten

Tafel 4.1: Merkmale und mögliche Werte

Wörtern nur die Struktur der letzten drei bzw. letzten drei betonbaren Silben relevant sei. Demnach muß man erwarten, daß das Hinzufügen von Information der weiter vorne stehenden Silben die Qualität der erzeugten Regeln nicht wesentlich verbessert. Die zweite Annahme betrifft die Auswahl der Silbentypen. Möglicherweise geht durch diese Generalisierung wertvolle Information verloren. In einer vergleichbaren Studie für das Holländische [Dael&al93], wurde diese Möglichkeit bereits erwähnt.

In der zweiten Lernreihe werden die zu erwartenden Erkennungsraten für zwei verschiedene Codierungen untersucht. Zum einen wird dem Algorithmus nur die Typ-Information, also nur Merkmal 1, angeboten, zum anderen die detailliertere Information der Merkmale 2–6. Es ist nicht zu erwarten, daß die zweite Codierung schlechtere Erkennungsraten erreicht, da die durch diese Merkmale induzierte Äquivalenzrelation sehr viel feinkörniger ist und dadurch bessere Approximationen durch die Rough Sets erlaubt. Jedoch wäre es eine Rechtfertigung der in der Phonologie üblichen Typeneinteilung, wenn die Codierung in Typen nur vernachlässigbar schlechter wäre als die aus den Einzelmerkmalen bestehende Codierung.

Aufgrund der Ergebnisse der zweiten Lernreihe wird die dritte Lernreihe durchgeführt, um festzustellen, inwiefern die Vorteile der Codierung ein Einzelmerkmale gegenüber der Codierung in Typen auf der methodologisch problematischen Längeninformation beruhen.

Anders als bei den phonetischen Untersuchungen wird der Parameter *Roughness* des Lernalgorithmus nicht konstant gehalten, sondern systematisch variiert. Mit diesem Parameter wird der Algorithmus zur Generalisierung gezwungen. Da die Typ-Codierung bereits eine Generalisierung gegenüber der Codierung in Einzelmerkmale darstellt, ist anzunehmen, daß für die verschiedenen Codierungen unterschiedliche Werte für diesen Parameter optimal sind. Als Vergleichskriterium wird die beste erzielbare Erkennungsrate verwendet.

4.3.1 Erste Lernreihe

In der ersten Lernreihe wird untersucht, wie gut die maschinell erzeugten Regeln die Daten des Korpus abdecken. Dazu werden für unterschiedliche Roughnesswerte und für vier verschiedene Untermengen der Merkmale aus allen Beispielen des Korpus Regeln generiert. Dann werden die Regeln wiederum auf alle Beispiele angewendet und so die Abdeckung ermittelt. In dieser Lernreihe liegt also ein *Lern = Test*-Szenario vor. Die Ergebnisse sind in Abbildung 4.1 zusammengefaßt. Auf der linken Seite der Abbildung, in den Teilen (a) und (b), standen dem Lernalgorithmus nur die Typ-Merkmale zur Verfügung, auf der rechten Seite, Teile (c) und (d), die Einzelmerkmale (*Onset*, *Hoehe*, *Laenge*, *Gespanntheit*, *Coda*). Bei den beiden oberen Teilen (a) und (c) waren die Merkmale aller Silben zugänglich, in den unteren Teilen nur die der letzten drei Silben.

Die Abbildung zeigt, daß sich mit Zunahme des Roughnessparameters die Anzahl der produzierten Regeln verringert. Wenn die Roughness den Wert 1 an-

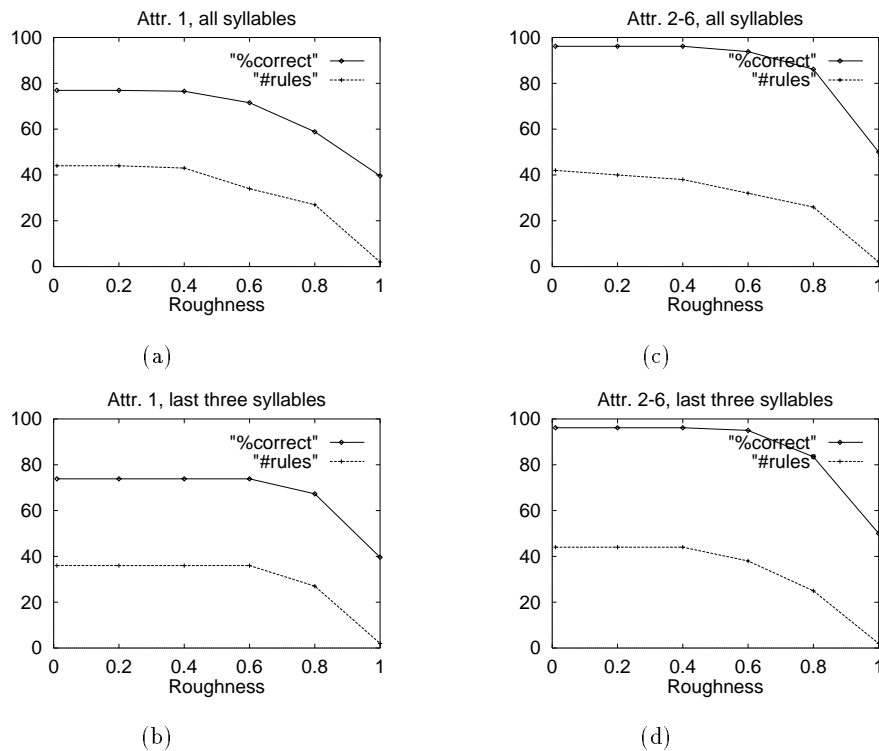


Abbildung 4.1: Ergebnisse der ersten Lernreihe. Abgebildet sind die Anzahl der generierten Regeln und der Prozentsatz der davon abgedeckten Fälle des Korpus für verschiedene Roughnesswerte (Lernmenge = Testmenge).

nimmt, wird für jedes Konzept nur noch eine einzige Regel erzeugt. Da in dieser Lernreihe Lern- und Testmenge identisch sind, nimmt der von den Regeln abgedeckte Bereich mit zunehmender Generalisierung ab. Wie wir noch sehen werden, muß dieses Verhalten für disjunkte Lern- und Testmengen nicht so sein. Im allgemeinen erwarten wir dort eine Verbesserung der Erkennungsraten durch die stärkere Generalisierung, die der Algorithmus bei höheren Roughnesswerten vornimmt. Ausreißer werden dann von den Regeln nicht mehr so gut abgedeckt. Dies ist sinnvoll für die Vorhersage neuer Fälle, denn wenn neue Daten klassifiziert werden sollen, ist es notwendig, von Ausreißern in der Lernmenge zu abstrahieren, damit keine Überanpassung an die Lernmenge eintritt.

Der Vergleich der Messungen auf der linken Seite mit den Messungen auf der rechten Seite von Abb. 4.1 zeigt, daß die Abdeckung mit der ersten Codierung, den in der Phonologie gebräuchlichen Typen, deutlich schlechter ist, als die Codierung in Einzelmerkmale. Anders ausgedrückt zeigt ein Vergleich der nicht richtig klassifizierten Fälle in den Graphiken, daß die Codierung in Typen mehr Rauschen in den Daten zur Folge hat als die andere Codierung. Die nächste Lernreihe wird zeigen, inwiefern sich dieser Vorteil der zweiten Codierung bei der Vorhersage neuer Fälle auswirkt.

Beim Vergleich der jeweils oberen Abbildungen mit den unteren erkennt man, daß die Information von weiter vorne liegenden Silben tatsächlich kaum eine

Betonung==> antep.		Betonung==> penult		Betonung==> final	
=====		=====		=====	
Attribute	Max.Loc.Str	Attribute	Max.Loc.Str	Attribute	Max.Loc.Str
=====	=====	=====	=====	=====	=====
Laenge_3	0.54	Laenge_2	0.55	Laenge_1	0.47
Gespanntheit_2	0.40	Laenge_1	0.33	Laenge_2	0.42
Hoehe_3	0.39	Coda_2	0.25	Coda_1	0.25
Coda_2	0.38	Onset_3	0.23	Laenge_3	0.14
Coda_3	0.33	Coda_1	0.22	Onset_1	0.13
Onset_3	0.32	Hoehe_3	0.18	Gespanntheit_1	0.12
Laenge_1	0.27	Hoehe_2	0.16	Hoehe_4	0.09
Coda_4	0.23	Gespanntheit_1	0.15	Hoehe_3	0.09
Laenge_2	0.22	Onset_1	0.14	Hoehe_2	0.08
Hoehe_2	0.22	Coda_3	0.14	Hoehe_1	0.07
		Hoehe_1	0.13	Gespanntheit_3	0.07

Tafel 4.2: Attributstärke für Lernreihe 1, zweite Codierung, alle Silben, Roughness = 0.60.

Verbesserung der Abdeckung erbringt. Dies bestätigt die „three syllable window“ These der Phonologen, die besagt, daß nur Information der letzten drei (betonbaren) Silben für die Betonungszuweisung relevant ist. Weitere Rechtfertigung dieser Ansicht kommt von der ebenfalls vom Algorithmus protokollierten Attributstärke, die angibt, wie wichtig das betreffende Attribut für die Klassifikation ist. Sie kann einen Wert zwischen 0 und 1 annehmen, wobei hohe Werte große Bedeutung ausdrücken. In Tafel 4.2 sind die Attributstärken aus einem Lauf abgedruckt, in dem die Merkmale aller Silben zur Verfügung standen. Darin gibt die an die Attribute mit einem Unterstrich angefügte Zahl die Silbe an, zu der das Atribut gehört. **Laenge_1** bezeichnet also die Nukleusvokallänge der letzten Silbe, **Laenge_2** die Nukleusvokallänge der vorletzten Silbe und so fort. Tafel 4.2 zeigt, daß der Algorithmus den Merkmalen der weiter als drei Silben vom Wortende entfernten Silben fast keine Bedeutung zumißt. Lediglich zwei Merkmale der viertletzten Silbe sind in den Listen aufgeführt, von den noch weiter vorne stehenden Silben gar keine. Die Merkmale der letzten drei Silben werden hingegen fast alle verwendet. Auch in den anderen, nicht abgedruckten Lernläufen hat der Algorithmus den Attributen der letzten drei Silben deutlich größere Wichtigkeit zugeschrieben als den Merkmalen der weiter vorne stehenden Silben. Aufgrund der in dieser Reihe gewonnenen Erkenntnisse werden in den folgenden Lernreihen nur noch Merkmale der letzten drei Silben verwendet.

4.3.2 Zweite Lernreihe

In dieser Lernreihe wird abgeschätzt, welche Erkennungsraten für neue, dem Algorithmus nicht bekannte Daten zu erwarten sind. Auch diese Reihe wird für beide Codierungen parallel durchgeführt. Analog zum Vorgehen bei der Untersuchung der akustischen Korrelate, ist auch hier das Korpus pseudozufällig in fünf Teile zerlegt. Jeweils vier Teile werden zum Lernen verwendet, der verbleibende fünfte Teil wird mit den gelernten Regeln klassifiziert. Die gemittelten

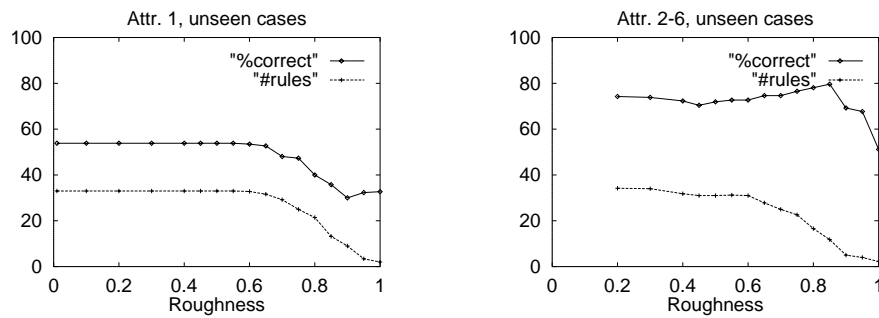


Abbildung 4.2: Ergebnisse der zweiten Lernreihe. Die Bilder zeigen die über die jeweils fünf Läufe gemittelten Erkennungsraten (korrekt vorhergesagte Testmuster) sowie die generierte Regelanzahl für verschiedene Roughnesswerte (Lernmenge \neq Testmenge).

Erkennungsraten der fünf Läufe, also die jeweiligen prozentualen Anteile korrekt vorhergesagter Betonungen, dienen als Schätzung der zu erwartenden Erkennungsrate. Um die Vergleichbarkeit der Ergebnisse zu ermöglichen, wird die Aufteilung der Daten für alle Läufe dieser und der folgenden dritten Lernreihe beibehalten.

In Abb 4.2 sind die gemittelten Erkennungsraten sowie die gemittelte Anzahl an produzierten Regeln für verschiedene Roughnesswerte abgetragen. Der linke Teil der Abbildung zeigt die Ergebnisse für die erste Codierung, der rechte Teil die Ergebnisse der zweiten Codierung. Obwohl bei beiden die Prozentzahlen gegenüber der Abdeckung der letzten Lernreihe gesunken sind, bleibt der beträchtliche Vorteil der Codierung in Einzelmerkmale gegenüber der Codierung in phonologische Typen erhalten. Auffällig bei der Betrachtung der beiden Bilder ist die Ausbildung eines Gipfels bei den Erkennungsraten der Codierung in Einzelmerkmale bei einer Roughness von 0,85. Ein solcher Gipfel ist bei der traditionellen Codierung nicht erkennbar. Dieser Unterschied bedeutet, daß sich die Erkennungsrate für die zweite Codierung verbessert, wenn der Algorithmus zu einer stärkeren Generalisierung gezwungen wird, während sie bei der ersten Codierung stetig abnimmt. Da für die erste Codierung dieses Verhalten nicht beobachtbar ist, könnte man vermuten, daß die durch die Typeinteilung vorgenommene Generalisierung möglicherweise bereits zu stark ist. Jedenfalls werden die Erkennungsraten mit diesem Algorithmus nicht mehr weiter verbessert, wenn man ihn zu stärkerer Generalisierung zwingt. In der nächsten Lernreihe wird näher untersucht werden, was die Ursache für das wesentlich bessere Abschneiden der zweiten Codierung ist.

Die Betrachtung der einzelnen Läufe in Abb. 4.3 gibt Aufschluß darüber, welche Roughnesswerte für die Codierungen sinnvoll sind. Wie man sieht, sind für die erste Codierung kleine Roughnesswerte bis zu etwa 0,5 sinnvoll. Lediglich für einen der fünf Läufe konnte für Roughness 0,75 eine minimale Verbesserung der Erkennungsrate erzielt werden, für alle anderen jedoch nicht. Bei der zweiten Codierung hingegen ist 0,85 ein geeigneter Wert, der auch schon durch Betrachtung der gemittelten Kurve nahegelegt wird. Wie man den einzelnen

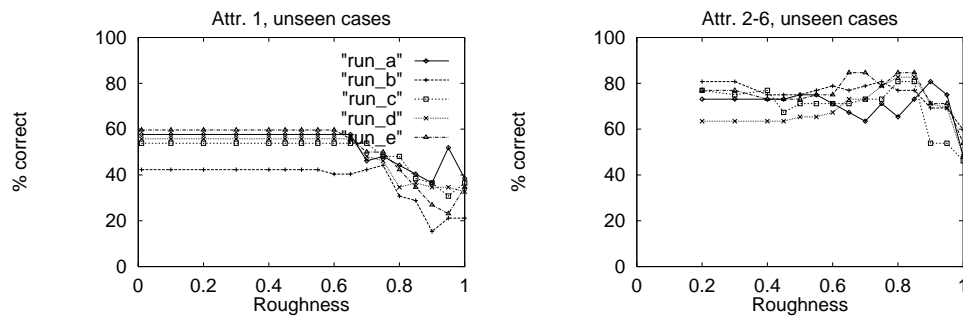


Abbildung 4.3: Ergebnisse der zweiten Lernreihe. Die Bilder zeigen die Erkennungsraten für die fünf einzelnen Läufe (Lernmenge \neq Testmenge).

Läufen entnimmt, wird bei allen Läufen die beste Leistung in etwa bei Roughness 0,85 erbracht. Die besten Erkennungsraten liegen immer über 80% für einen Roughnesswert im Bereich zwischen 0,75 und 0,90.

Erwähnenswert ist die Tatsache, daß die bei der zweiten Codierung sinnvolle stärkere Generalisierung sich auch recht deutlich in der Verminderung der Regelanzahl bemerkbar macht. Während für niedrige Roughnesswerte über 30 Regeln für eine Erkennungsrate von durchschnittlich 74,23% notwendig sind, werden bei Roughness 0,85 mit nur 9-15 Regeln 79,616% erreicht, also mit etwa vier Regeln für jedes zu lernende Konzept. Die Erkennungsraten bei diesem Roughnesswert liegen bei den einzelnen Läufen zwischen 73,08% und 84,62%. In Abschnitt 4.4 werden die gelernten Regeln der verschiedenen Codierungen vorgestellt und anschließend diskutiert. Bei der Codierung in phonologische Typen sind für die bestmöglichen Erkennungsraten über 30 Regeln notwendig. Dies erschwert, wie wir noch sehen werden, einen Vergleich mit den von Phonologen aufgestellten Regeln. Die phonologische Codierung wurde in die Beschreibung der Wörter aufgenommen, um einen einfacheren Vergleich zu ermöglichen.

Das nachträgliche Festlegen von Parametern wie die hier durchgeführte nachträgliche Bestimmung der Roughness ist nicht unproblematisch, da die Gefahr besteht, sich damit indirekt an die Testdaten zu adaptieren. Ein ideales Vorgehen wäre, in Vorversuchen mit kleinen Teilen der Daten alle Parameter des Lernalgorithmus zu fixieren und dann die Bestimmung der zu erwartenden Erkennungsrate mit noch nicht gesehenen Daten durchzuführen. Dieses Vorgehen kann hier jedoch wegen der geringen Anzahl von Daten nicht durchgeführt werden. Das gewählte Vorgehen ist hier vertretbar, da in dieser Arbeit nicht eine möglichst genaue Abschätzung der erzielbaren Vorhersagegenauigkeit von Interesse ist, sondern die durch ML-Verfahren mögliche Untersuchung phonologisch interessanter Fragestellungen. Dazu gehört sicherlich die Frage, ob die verbreitete Codierung in Silbentypen zu rechtfertigen ist gegenüber einer Codierung in Einzelmerkmale. Die in dieser Lernreihe erzielten Ergebnisse legen nahe, daß die traditionelle Typeneinteilung zumindest verbesserungsfähig sein könnte.

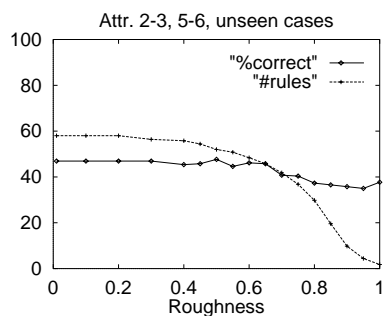


Abbildung 4.4: Ergebnisse der dritten Lernreihe. Die Abbildung zeigt die gemittelten Erkennungsraten über verschiedene Roughnesswerte sowie die dazu im Mittel notwendige Regelanzahl (Lernmenge \neq Testmenge).

4.3.3 Dritte Lernreihe

Die Frage, die sich durch die Ergebnisse der zweiten Lernreihe stellt, ist, warum die Codierung in Einzelmerkmale so deutlich besser abschneidet als die traditionelle, über Jahre gebräuchliche Codierung der Phonologen. Der wesentliche Unterschied zwischen den beiden Codierungen liegt in der unterschiedlichen Berücksichtigung der Nukleuslänge. In den Typen ist die Länge des Nukleusvokals nur indirekt enthalten. Die einzigen Implikationen über die Länge des Nukleusvokals, die man aus dem Typ der Silbe entnehmen kann, sind folgende: Wenn es sich um den Typ einer Schwa-Silbe, also \textcircled{a} oder $\textcircled{C}\textcircled{a}$ handelt, ist der Nukleus kurz, wenn es sich um eine Diphthong-Silbe V_V handelt, ist der Nukleus lang.

Die Information über die Vokallänge, die in die Silbentypen codiert ist, ist auch in den Einzelmerkmalen ohne das Längenmerkmal enthalten. In dieser dritten Lernreihe wird die Vorhersagefähigkeit der Einzelmerkmale ohne das problematische Längenmerkmal untersucht. Für die in dieser Reihe verwendete dritte Codierung stehen also dem Lernalgorithmus die Merkmale **Onset**, **Höhe**, **Gespanntheit** und **Coda** zur Verfügung. Wie schon in der zweiten Lernreihe werden nur Merkmale der letzten drei Silben berücksichtigt. Die Aufteilung in Lern- und Testmengen der zweiten Lernreihe sowie die sich daraus ergebenden fünf Läufe je Roughnesswert werden beibehalten, damit die erzielten Erkennungsraten direkt vergleichbar sind. Abbildung 4.4 zeigt die gemittelten Werte von Erkennungsrate und Anzahl der gelernten Regeln für unterschiedliche Werte des Generalisierungsparameters.

Für die dritte Codierung produziert der Lernalgorithmus mit Abstand die meisten Regeln. Mit beiden anderen Codierungen erzeugte der Lernalgorithmus für niedrige Roughnesswerte zwischen 31 und 36 Regeln, während er für die dritte Codierung zwischen 54 und 62 Regeln erzeugt. Die Anzahl an Regeln, für die mit der zweiten Codierung die besten Erkennungsraten zu erwarten sind, ist mit 9 bis 15 Regeln sogar noch deutlich niedriger. Bei der dritten Codierung, also den Einzelmerkmalen ohne Längenmerkmal, erniedrigt sich die Regelanzahl nur geringfügig auf etwa 52 Regeln für den Roughnesswert 0,5, der die besten

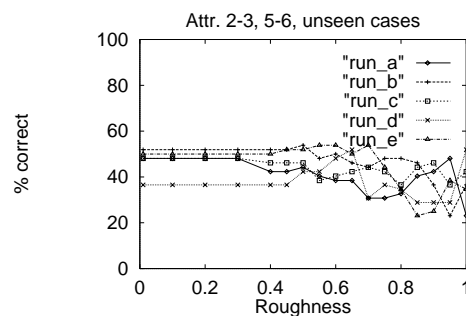


Abbildung 4.5: Ergebnisse der dritten Lernreihe. Die Abbildung zeigt die Erkennungsraten für die fünf einzelnen Läufe (Lernmenge \neq Testmenge).

Erkennungsraten verspricht. Damit wird durchschnittlich eine Erkennungsrate von 47,692% erreicht. Der Gipfel ist hier nicht so stark ausgeprägt wie bei der zweiten Codierung. Wie aus Abb. 4.5 ersichtlich ist, sind auch die Maxima der einzelnen Läufe weiter gestreut. Sie liegen zwischen 48,08% und 53,85% für Roughnesswerte zwischen 0,30 und 0,65.

4.4 Regeln

Die im letzten Abschnitt gefundenen optimalen Werte für den Roughnessparameter werden nun eingesetzt, um aus allen zur Verfügung stehenden Beispielfällen des Korpus Regeln zu generieren. Diese werden hier zunächst undisputiert vorgestellt. Im anschließenden Abschnitt werden sie dann mit entsprechenden Aussagen aus der phonologischen Literatur in Beziehung gesetzt.

4.4.1 Erste Codierung: nur Typinformation

Die im folgenden abgedruckten Regeln wurden aus allen 260 Fällen des Korpus bei einem Roughnesswert von 0,50 extrahiert. Der *rule precision threshold* betrug 0,50. Als Merkmale standen dem Algorithmus die Merkmale **Typ** der letzten drei Silben zur Verfügung. Die Parameter des Lernalgorithmus entsprechen den Parametern der zweiten Lernreihe, bei denen die besten Erkennungsraten zu erwarten sind. Die Regeln sind aus allen Fällen des Korpus berechnet und nicht nur aus 4/5 der Fälle. Unterstellt man einen repräsentativen Aufbau des Korpus, so sind mit diesen Regeln vergleichbare Erkennungsraten zu erwarten, wie für die in der zweiten Lernreihe erzeugten Regeln ermittelt wurden, also etwa 54%.

Attributstärke

Betonung==> antep.		Betonung==> penult		Betonung==> final	
Attribute	Max.Loc.Str	Attribute	Max.Loc.Str	Attribute	Max.Loc.Str
Typ_3	0.50	Typ_2	0.25	Typ_1	0.18
Typ_2	0.46	Typ_3	0.24	Typ_2	0.09
Typ_1	0.28	Typ_1	0.16	Typ_3	0.08
Coverage ==> 90.00%		Coverage ==> 78.50%		Coverage ==> 61.06%	

Regeln

Betonung antepenult				Typ
Nr.	Fälle	Precision	Regel	
1	15	73.3%	$\text{Typ}_1 = (\text{Vu} \vee \text{Vg} \vee \text{VuC} \vee \text{konson.}) \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{C@} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{VgC1})$	
2	16	68.8%	$\text{Typ}_1 = \text{VuC} \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{Vu} \vee \text{VgC1})$	
3	7	85.7%	$\text{Typ}_1 = (\text{Vu} \vee \text{konson.}) \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{V_V} \vee \text{Vu} \vee \text{VuCC1} \vee \text{C@} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{Vu}(\text{C}) \vee \text{Vg})$	
4	2	100.0%	$\text{Typ}_1 = \text{VuC} \wedge$ $\text{Typ}_2 = (\text{V_V} \vee \text{Vu} \vee \text{VuCC1}) \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{Vu} \vee \text{VgC1})$	
5	4	75.0%	$\text{Typ}_1 = (\text{Vu} \vee \text{Vg} \vee \text{VgC1} \vee \text{konson.}) \wedge$ $\text{Typ}_2 = (\text{C@} \vee \text{VgC1})$	
6	11	72.7%	$\text{Typ}_1 = \text{VuC} \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{C@} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{Vu}(\text{C}) \vee \text{Vg})$	
7	1	100.0%	$\text{Typ}_1 = \text{VgC1} \wedge$ $\text{Typ}_2 = (\text{V_V} \vee \text{Vu} \vee \text{VuCC1} \vee \text{C@} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{VgC1})$	
8	1	100.0%	$\text{Typ}_2 = \text{V_V} \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{Vu} \vee \text{VgC1})$	

Betonung penult				Typ
Nr.	Fälle	Precision	Regel	
9	21	71.4%	$\text{Typ}_1 = (\text{Vu} \vee \text{Vg} \vee \text{C@} \vee \text{VuCC1} \vee \text{V_V} \vee \text{konson.}) \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{Vu} \vee -)$	
10	9	100.0%	$\text{Typ}_1 = \text{C@} \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{Vu} \vee \text{VuCC1} \vee \text{C@} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{Vu} \vee - \vee \text{V_V} \vee \text{VuCC1} \vee \text{VgC1})$	
11	23	60.9%	$\text{Typ}_1 = \text{VuC} \wedge$ $\text{Typ}_2 = \text{VuC}$	

Betonung penult (Fortsetzung)				Typ
Nr.	Fälle	Precision	Regel	
12	22	81.8%	$\text{Typ}_2 = (\text{Vu}(\text{C}) \vee \text{V}_\text{V}) \wedge$ $\text{Typ}_3 = -$	
13	9	100.0%	$\text{Typ}_1 = \text{C@} \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{VuC} \vee \text{C@} \vee \text{VgC1})$	
14	6	100.0%	$\text{Typ}_1 = (\text{Vu} \vee \text{Vg} \vee \text{VuC} \vee \text{C@} \vee \text{VuCC1} \vee \text{V}_\text{V} \vee \text{konson.}) \wedge$ $\text{Typ}_2 = (\text{VuC} \vee \text{C@} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{Vu}(\text{C}) \vee \text{Vg} \vee \text{V}_\text{V} \vee \text{VuCC1} \vee \text{VgC1})$	
15	17	64.7%	$\text{Typ}_1 = \text{Vg} \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{VuC} \vee \text{C@} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{Vu} \vee - \vee \text{V}_\text{V} \vee \text{VuCC1} \vee \text{VgC1})$	
16	36	61.1%	$\text{Typ}_1 = \text{VuC} \wedge$ $\text{Typ}_2 = (\text{VuC} \vee \text{Vu}(\text{C}) \vee \text{V}_\text{V} \vee \text{Vu} \vee \text{VuCC1} \vee \text{C@} \vee \text{VgC1}) \wedge$ $\text{Typ}_3 = -$	
17	2	100.0%	$\text{Typ}_1 = (\text{Vu} \vee \text{VuCC1} \vee \text{V}_\text{V} \vee \text{konson.}) \wedge$ $\text{Typ}_2 = (\text{VuC} \vee \text{Vu}(\text{C}) \vee \text{V}_\text{V} \vee \text{Vu}) \wedge$ $\text{Typ}_3 = \text{Vu}$	
18	2	100.0%	$\text{Typ}_1 = (\text{Vu} \vee \text{C@} \vee \text{VuCC1} \vee \text{V}_\text{V} \vee \text{konson.}) \wedge$ $\text{Typ}_3 = (\text{V}_\text{V} \vee \text{VuCC1} \vee \text{VgC1})$	
19	2	100.0%	$\text{Typ}_2 = (\text{Vu}(\text{C}) \vee \text{V}_\text{V}) \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{V}_\text{V} \vee \text{VuCC1} \vee \text{VgC1})$	
20	3	100.0%	strut $\text{Typ}_2 = \text{VuCC1}$	
21	2	100.0%	$\text{Typ}_2 = \text{VgC1}$	
22	1	100.0%	$\text{Typ}_1 = \text{Vg} \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{VgC1})$	
23	3	66.7%	$\text{Typ}_1 = \text{VgC1} \wedge$ $\text{Typ}_2 = \text{Vu} \wedge$ $\text{Typ}_3 = (\text{Vu}(\text{C}) \vee \text{Vg} \vee \text{V}_\text{V} \vee \text{VuCC1} \vee \text{VgC1})$	

Betonung final				Typ
Nr.	Fälle	Precision	Regel	
24	15	100.0%	$\text{Typ}_1 = (\text{VuCC1} \vee \text{V}_\text{V}) \wedge$ $\text{Typ}_3 = (\text{VuC} \vee \text{Vu} \vee \text{Vu}(\text{C}) \vee \text{Vg} \vee \text{V}_\text{V} \vee \text{VuCC1} \vee \text{VgC1})$	
25	16	75.0%	$\text{Typ}_1 = \text{VgC1} \wedge$ $\text{Typ}_2 = (\text{Vg} \vee \text{VgC1})$	
26	18	77.8%	$\text{Typ}_1 = (\text{VgC1} \vee \text{C@} \vee \text{VuCC1}) \wedge$ $\text{Typ}_2 = \text{VuC} \wedge$ $\text{Typ}_3 = (- \vee \text{Vu}(\text{C}) \vee \text{Vg} \vee \text{V}_\text{V} \vee \text{VuCC1} \vee \text{VgC1})$	
27	4	100.0%	$\text{Typ}_1 = (\text{Vg} \vee \text{VuC} \vee \text{V}_\text{V}) \wedge$ $\text{Typ}_2 = (\text{Vu} \vee \text{VuCC1} \vee \text{C@}) \wedge$ $\text{Typ}_3 = (\text{Vu}(\text{C}) \vee \text{Vg} \vee \text{V}_\text{V} \vee \text{VuCC1} \vee \text{VgC1})$	

Betonung final (Fortsetzung)				Typ
Nr.	Fälle	Precision	Regel	
28	5	100.0%	Typ_1 = (VuCC1 ∨ V_V) ∧ Typ_2 = Vu	
29	2	100.0%	Typ_1 = (Vu ∨ Vg ∨ VuC ∨ V_V ∨ konson.) ∧ Typ_2 = (Vu(C) ∨ V_V) ∧ Typ_3 = (VuC ∨ Vu ∨ Vu(C) ∨ Vg ∨ V_V ∨ VuCC1 ∨ VgC1)	
30	3	100.0%	Typ_1 = Vg ∧ Typ_2 = (Vu ∨ VuCC1 ∨ C@) ∧ Typ_3 = (Vu ∨ Vu(C) ∨ Vg ∨ V_V ∨ VuCC1)	
31	2	100.0%	Typ_1 = (Vg ∨ VuC ∨ C@ ∨ VuCC1 ∨ V_V) ∧ Typ_2 = (VuCC1 ∨ C@) ∧ Typ_3 = (VuC ∨ Vu ∨ Vu(C) ∨ Vg ∨ V_V ∨ VuCC1 ∨ VgC1)	
32	17	64.7%	Typ_1 = (VuC ∨ C@) ∧ Typ_2 = (Vg ∨ VuCC1 ∨ C@ ∨ VgC1) ∧ Typ_3 = –	
33	1	100.0%	Typ_3 = VuCC1	
34	2	100.0%	Typ_1 = VgC1 ∧ Typ_3 = (Vu ∨ VuCC1)	
35	2	100.0%	Typ_1 = VgC1 ∧ Typ_2 = Vu ∧ Typ_3 = –	

4.4.2 Zweite Codierung: Einzelmerkmale

Die hier abgedruckten Regeln sind aus allen 260 Fällen des Korpus bei einem Roughnesswert von 0,85 berechnet. Der *rule precision threshold* lag bei 0,50. Als Merkmale standen dem Algorithmus *Onset*, *Hoehe*, *Laenge*, *Gespanntheit* und *Coda* der letzten drei Silben zur Verfügung. Unterstellt man einen repräsentativen Aufbau des Korpus, so sind für diesen Regeln ähnliche Erkennungsraten zu erwarten wie in der zweiten Lernreihe, also ungefähr 80%.

Attributstärke

Betonung==> antep.		Betonung==> penult		Betonung==> final	
Attribute	Max.Loc.Str	Attribute	Max.Loc.Str	Attribute	Max.Loc.Str
Coda_3		Laenge_2		Laenge_1	
Laenge_3		Laenge_1		Laenge_2	
Gespanntheit_2		Coda_2		Coda_1	
		Coda_3		Laenge_3	
		Coda_1		Hoehe_2	
				Hoehe_3	
				Coda_3	
Coverage ==> 55.00%		Coverage ==> 93.46%		Coverage ==> 92.92%	

Regeln

Betonung antepenult			Onset Hoehe Laenge Gespanntheit Coda
Nr.	Fälle	Precision	Regel
1	20	90.0%	Laenge_3 = lang \wedge Gespanntheit_2 = (gespannt \vee Schwa)
2	3	100.0%	Laenge_3 = lang \wedge Gespanntheit_2 = ungesp.
3	1	100.0%	Coda_3 = ($- \vee$ ambiv. \vee mehrf.g.) \wedge Gespanntheit_2 = Schwa

Betonung penult			Onset Hoehe Laenge Gespanntheit Coda
Nr.	Fälle	Precision	Regel
4	62	96.8%	Laenge_2 = lang
5	54	75.9%	Laenge_1 = kurz \wedge Coda_2 = (einf.g. \vee ambiv. \vee mehrf.g.)
6	24	100.0%	Laenge_1 = kurz \wedge Coda_3 = $- \wedge$ Coda_1 = offen

Betonung final			Onset Hoehe Laenge Gespanntheit Coda
Nr.	Fälle	Precision	Regel
7	49	87.8%	Laenge_1 = lang \wedge Laenge_2 = kurz \wedge Hoehe_3 = ($- \vee$ hoch \vee steigend)
8	37	86.5%	Laenge_1 = lang \wedge Laenge_2 = kurz \wedge Laenge_3 = (kurz $\vee -$) \wedge Coda_3 = (einf.g. \vee offen)
9	16	93.8%	Laenge_1 = lang \wedge Laenge_2 = kurz \wedge Hoehe_2 = (hoch \vee steigend)
10	12	100.0%	Coda_3 = (einf.g. \vee offen) \wedge Coda_1 = mehrf.g.
11	9	100.0%	Hoehe_3 = (tief \vee mittel) \wedge Coda_1 = mehrf.g.
12	56	71.4%	Laenge_2 = kurz \wedge Hoehe_2 = (mittel \vee tief) \wedge Hoehe_3 = ($- \vee$ hoch \vee steigend) \wedge Coda_3 = ($- \vee$ ambiv. \vee mehrf.g.) \wedge Coda_1 = (einf.g. \vee mehrf.g.)

4.4.3 Dritte Codierung: Einzelmerkmale ohne Länge

Die folgenden Regeln wurden aus allen 260 Fällen des Korpus bei einem Roughnesswert von 0,50 gewonnen. Der *rule precision threshold* betrug 0,50. Als Merkmale standen dem Algorithmus **Onset**, **Hoehe**, **Gespanntheit** und **Coda** der letzten drei Silben zur Verfügung. Wie bei der dritten Lernreihe sind für diese Regeln Erkennungsraten von ca. 48% zu erwarten.

Attributstärke

Betonung==> antep.		Betonung==> penult		Betonung==> final	
Attribute	Max.Loc.Str	Attribute	Max.Loc.Str	Attribute	Max.Loc.Str
Coda_3	0.56	Coda_2	0.25	Coda_1	0.25
Gespanntheit_2	0.46	Coda_3	0.24	Gespanntheit_1	0.12
Onset_3	0.40	Gespanntheit_2	0.22	Hoehe_2	0.09
Hoehe_3	0.39	Hoehe_3	0.18	Hoehe_3	0.09
Coda_2	0.38	Gespanntheit_1	0.16	Coda_3	0.09
Gespanntheit_1	0.29	Hoehe_2	0.16	Hoehe_1	0.07
Hoehe_2	0.22	Gespanntheit_3	0.13	Onset_3	0.06
Hoehe_1	0.20	Hoehe_1	0.13	Coda_2	0.05
Coda_1	0.09	Coda_1	0.13	Gespanntheit_2	0.01
Onset_2	0.07				
Coverage ==> 92.50%		Coverage ==> 79.44%		Coverage ==> 84.07%	

Betonung antepenult			Onset Hoehe gespanntheit Coda
Nr.	Fälle	Precision	Regel
1	10	100.0%	Coda_2 = offen \wedge Coda_1 = einf.g. \wedge Coda_3 = (einf.g. \vee offen \vee mehrf.g.) \wedge Gespanntheit_1 = (ungesp. \vee Schwa \vee konson.) \wedge Hoehe_1 = hoch
2	8	100.0%	Coda_2 = offen \wedge Hoehe_2 = (mittel \vee tief \vee steigend) \wedge Coda_1 = einf.g. \wedge Coda_3 = (einf.g. \vee offen \vee mehrf.g.) \wedge Gespanntheit_1 = (ungesp. \vee Schwa \vee konson.) \wedge Hoehe_1 = (hoch \vee mittel \vee steigend \vee konson.)
3	4	100.0%	Coda_2 = offen \wedge Coda_1 = einf.g. \wedge Coda_3 = (einf.g. \vee - \vee ambiv. \vee mehrf.g.) \wedge Hoehe_3 = (tief \vee mittel \vee steigend) \wedge Gespanntheit_1 = (ungesp. \vee Schwa \vee konson.) \wedge Hoehe_1 = (hoch \vee mittel \vee steigend \vee konson.)
4	5	100.0%	Gespanntheit_2 = (gespannt \vee Schwa) \wedge Onset_3 = vorh. \wedge Coda_2 = offen \wedge Hoehe_2 = (mittel \vee tief \vee steigend) \wedge Coda_1 = einf.g. \wedge Hoehe_3 = (tief \vee mittel \vee steigend) \wedge Hoehe_1 = (hoch \vee mittel \vee steigend \vee konson.)
5	2	100.0%	Gespanntheit_2 = (gespannt \vee Schwa) \wedge Hoehe_2 = (mittel \vee tief \vee steigend) \wedge Coda_3 = (einf.g. \vee mehrf.g.) \wedge Hoehe_3 = (tief \vee mittel \vee steigend) \wedge Gespanntheit_1 = (ungesp. \vee konson.)
6	4	75.0%	Hoehe_2 = hoch \wedge Coda_1 = (offen \vee mehrf.g.) \wedge Coda_3 = (einf.g. \vee - \vee ambiv. \vee mehrf.g.) \wedge Hoehe_3 = (tief \vee hoch \vee steigend)
7	2	100.0%	Gespanntheit_2 = (gespannt \vee Schwa) \wedge Hoehe_2 = (mittel \vee tief \vee steigend) \wedge Coda_1 = einf.g. \wedge Hoehe_3 = (tief \vee steigend) \wedge Hoehe_1 = (hoch \vee mittel \vee steigend \vee konson.)

Betonung antepenult (Fortsetzung)			Onset Hoehe Gespanntheit Coda
Nr.	Fälle	Precision	Regel
8	2	100.0%	Onset_3 = (n.vorh. \vee -) \wedge Coda_1 = einf.g. \wedge Coda_3 = offen \wedge Hoehe_3 = (tief \vee hoch \vee steigend)
9	4	75.0%	Onset_3 = vorh. \wedge Hoehe_2 = (mittel \vee tief \vee steigend) \wedge Coda_1 = (offen \vee mehrf.g.) \wedge Hoehe_3 = (tief \vee hoch \vee steigend) \wedge Gespanntheit_1 = (ungesp. \vee Schwa \vee konson.) \wedge Hoehe_1 = (tief \vee hoch \vee konson.)
10	2	100.0%	Gespanntheit_2 = ungesp. \wedge Coda_2 = offen \wedge Coda_1 = einf.g. \wedge Coda_3 = (einf.g. \vee mehrf.g.)
11	3	100.0%	Hoehe_2 = hoch \wedge Coda_3 = offen \wedge Hoehe_3 = (tief \vee mittel \vee steigend) \wedge Hoehe_1 = (tief \vee konson.)
12	1	100.0%	Onset_3 = (n.vorh. \vee -) \wedge Hoehe_2 = hoch \wedge Coda_3 = offen
13	1	100.0%	Hoehe_2 = hoch \wedge Coda_1 = (offen \vee mehrf.g.) \wedge Coda_3 = (einf.g. \vee offen \vee mehrf.g.) \wedge Gespanntheit_1 = (gespannt \vee Schwa) \wedge Hoehe_1 = (tief \vee hoch \vee konson.)
14	1	100.0%	Gespanntheit_2 = ungesp. \wedge Coda_2 = offen \wedge Coda_1 = (offen \vee mehrf.g.) \wedge Gespanntheit_1 = (ungesp. \vee Schwa \vee konson.) \wedge Hoehe_1 = (tief \vee hoch \vee konson.)
15	1	100.0%	Onset_3 = (n.vorh. \vee -) \wedge Coda_3 = (einf.g. \vee offen \vee mehrf.g.) \wedge Gespanntheit_1 = Schwa
16	1	100.0%	Gespanntheit_2 = (gespannt \vee Schwa) \wedge Coda_1 = (offen \vee mehrf.g.) \wedge Coda_3 = (einf.g. \vee offen \vee mehrf.g.) \wedge Hoehe_3 = (- \vee hoch \vee steigend) \wedge Gespanntheit_1 = gespannt
17	1	100.0%	Gespanntheit_2 = ungesp. \wedge Onset_2 = vorh. \wedge Coda_3 = offen \wedge Hoehe_3 = (mittel \vee -) \wedge Gespanntheit_1 = (ungesp. \vee Schwa \vee konson.) \wedge Hoehe_1 = (hoch \vee mittel \vee steigend \vee konson.)

Betonung penult			Onset Hoehe Gespanntheit Coda
Nr.	Fälle	Precision	Regel
18	18	94.4%	Gespanntheit_2 = (ungesp. \vee Schwa) \wedge Gespanntheit_1 = (ungesp. \vee konson.) \wedge Hoehe_1 = hoch
19	16	93.8%	Coda_3 = - \wedge Gespanntheit_1 = (ungesp. \vee konson.) \wedge Hoehe_1 = hoch
20	10	100.0%	Gespanntheit_1 = (Schwa \vee konson.) \wedge Hoehe_2 = (hoch \vee mittel \vee steigend)
21	8	100.0%	Gespanntheit_2 = (ungesp. \vee Schwa) \wedge Coda_1 = einf.g. \wedge Coda_2 = (offen \vee ambiv. \vee mehrf.g.) \wedge Coda_3 = - \wedge Hoehe_2 = (hoch \vee mittel \vee steigend)

Betonung penult (Fortsetzung)			Onset Hoehe Gespanntheit Coda
Nr.	Fälle	Precision	Regel
22	8	100.0%	Gespanntheit ₁ = (ungesp. \vee gespannt \vee konson.) \wedge Hoehe ₂ = steigend
23	7	100.0%	Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₃ = (einf.g. \vee ambiv. \vee mehrf.g.) \wedge Gespanntheit ₁ = (gespannt \vee Schwa \vee konson.) \wedge Hoehe ₁ = (tief \vee mittel \vee steigend \vee konson.)
24	11	81.8%	Coda ₂ = (offen \vee ambiv. \vee mehrf.g.) \wedge Coda ₃ = – \wedge Hoehe ₂ = (mittel \vee steigend) \wedge Hoehe ₁ = hoch
25	11	81.8%	Gespanntheit ₂ = (ungesp. \vee Schwa) \wedge Hoehe ₃ = (mittel \vee –) \wedge Gespanntheit ₃ = (ungesp. \vee –) \wedge Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₂ = (offen \vee ambiv. \vee mehrf.g.) \wedge Hoehe ₁ = hoch
26	5	100.0%	Gespanntheit ₃ = (ungesp. \vee –) \wedge Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₂ = (offen \vee mehrf.g.) \wedge Gespanntheit ₁ = (gespannt \vee Schwa \vee konson.) \wedge Hoehe ₂ = mittel \wedge Hoehe ₁ = (tief \vee mittel \vee steigend \vee konson.)
27	6	100.0%	Gespanntheit ₃ = (ungesp. \vee –) \wedge Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₃ = (einf.g. \vee offen \vee ambiv. \vee mehrf.g.) \wedge Gespanntheit ₁ = (gespannt \vee Schwa \vee konson.) \wedge Hoehe ₂ = tief \wedge Hoehe ₁ = (tief \vee mittel \vee steigend \vee konson.)
28	5	100.0%	Hoehe ₃ = (mittel \vee –) \wedge Coda ₂ = (einf.g. \vee ambiv. \vee mehrf.g.) \wedge Coda ₃ = (einf.g. \vee offen \vee ambiv. \vee mehrf.g.) \wedge Hoehe ₂ = (tief \vee steigend)
29	6	100.0%	Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₂ = einf.g. \wedge Gespanntheit ₁ = (gespannt \vee Schwa \vee konson.) \wedge Hoehe ₁ = (tief \vee mittel \vee steigend \vee konson.)
30	6	100.0%	Gespanntheit ₂ = gespannt \wedge Gespanntheit ₃ = (ungesp. \vee –) \wedge Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₃ = offen \wedge Hoehe ₁ = (tief \vee mittel \vee steigend \vee konson.)
31	4	100.0%	Hoehe ₃ = (mittel \vee –) \wedge Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₂ = (einf.g. \vee ambiv. \vee mehrf.g.) \wedge Coda ₃ = (einf.g. \vee offen \vee ambiv. \vee mehrf.g.) \wedge Gespanntheit ₁ = (gespannt \vee Schwa \vee konson.)
32	6	83.3%	Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₃ = – \wedge Gespanntheit ₁ = (ungesp. \vee konson.) \wedge Hoehe ₂ = (hoch \vee steigend) \wedge Hoehe ₁ = (hoch \vee mittel \vee steigend \vee konson.)
33	6	83.3%	Gespanntheit ₂ = (ungesp. \vee Schwa) \wedge Hoehe ₃ = (mittel \vee –) \wedge Gespanntheit ₁ = (ungesp. \vee konson.) \wedge Hoehe ₂ = (hoch \vee steigend) \wedge Hoehe ₁ = (hoch \vee mittel \vee steigend \vee konson.)

Betonung penult (Fortsetzung)			Onset Hoehe Gespanntheit Coda
Nr.	Fälle	Precision	Regel
34	5	100.0%	Gespanntheit ₂ = (ungesp. \vee Schwa) \wedge Hoehe ₃ = (mittel \vee -) \wedge Coda ₂ = (offen \vee ambiv. \vee mehrf.g.) \wedge Hoehe ₂ = (hoch \vee steigend)
35	3	100.0%	Gespanntheit ₃ = (ungesp. \vee -) \wedge Coda ₂ = (einf.g. \vee ambiv. \vee mehrf.g.) \wedge Coda ₃ = offen \wedge Hoehe ₂ = (mittel \vee tief \vee steigend)
36	4	100.0%	Coda ₁ = einf.g. \wedge Coda ₂ = einf.g. \wedge Gespanntheit ₁ = (ungesp. \vee konson.) \wedge Hoehe ₂ = (tief \vee steigend) \wedge Hoehe ₁ = (hoch \vee mittel \vee steigend \vee konson.)
37	3	100.0%	Coda ₁ = (offen \vee mehrf.g.) \wedge Coda ₂ = (ambiv. \vee mehrf.g.) \wedge Gespanntheit ₁ = (ungesp. \vee konson.)
38	2	100.0%	Hoehe ₃ = (mittel \vee -) \wedge Coda ₁ = (offen \vee mehrf.g.) \wedge Hoehe ₂ = (mittel \vee steigend) \wedge Hoehe ₁ = (tief \vee konson.)
39	2	100.0%	Hoehe ₃ = (mittel \vee -) \wedge Coda ₃ = (einf.g. \vee offen \vee ambiv. \vee mehrf.g.) \wedge Gespanntheit ₁ = (gespannt \vee Schwa \vee konson.) \wedge Hoehe ₂ = (hoch \vee steigend) \wedge Hoehe ₁ = (tief \vee mittel \vee steigend \vee konson.)
40	2	100.0%	Gespanntheit ₂ = gespannt \wedge Coda ₂ = (einf.g. \vee ambiv. \vee mehrf.g.)
41	2	100.0%	Coda ₂ = (offen \vee mehrf.g.) \wedge Coda ₃ = offen \wedge Gespanntheit ₁ = (ungesp. \vee gespannt \vee konson.) \wedge Hoehe ₂ = (tief \vee steigend) \wedge Hoehe ₁ = (mittel \vee steigend \vee konson.)
42	2	100.0%	Coda ₁ = einf.g. \wedge Coda ₂ = (ambiv. \vee mehrf.g.) \wedge Gespanntheit ₁ = (gespannt \vee Schwa \vee konson.)
43	1	100.0%	Coda ₂ = (ambiv. \vee mehrf.g.) \wedge Hoehe ₁ = (tief \vee konson.)

Betonung final			Onset Hoehe Gespanntheit Coda
Nr.	Fälle	Precision	Regel
44	14	100.0%	Coda ₁ = mehrf.g. \wedge Hoehe ₂ = (mittel \vee tief) \wedge Coda ₂ = (offen \vee einf.g. \vee mehrf.g.)
45	12	100.0%	Coda ₁ = mehrf.g. \wedge Coda ₃ = (einf.g. \vee offen \vee mehrf.g.)
46	12	83.3%	Gespanntheit ₁ = gespannt \wedge Onset ₃ = (n.vorh. \vee -) \wedge Gespanntheit ₂ = ungesp. \wedge Coda ₁ = (einf.g. \vee mehrf.g.) \wedge Hoehe ₂ = (hoch \vee mittel \vee tief \vee steigend) \wedge Coda ₃ = (offen \vee - \vee ambiv. \vee mehrf.g.) \wedge Coda ₂ = (offen \vee einf.g. \vee mehrf.g.)

Betonung final (Fortsetzung)			Onset Hoehe Gespanntheit Coda
Nr.	Fälle	Precision	Regel
47	6	100.0%	Gespanntheit ₁ = gespannt ∧ Gespanntheit ₂ = ungesp. ∧ Coda ₃ = (einf.g. ∨ offen ∨ mehrf.g.) ∧ Hoehe ₁ = (tief ∨ hoch ∨ konson.) ∧ Coda ₂ = (offen ∨ mehrf.g.)
48	15	86.7%	Gespanntheit ₁ = (ungesp. ∨ Schwa ∨ konson.) ∧ Onset ₃ = (n.vorh. ∨ -) ∧ Gespanntheit ₂ = (gespannt ∨ Schwa) ∧ Coda ₁ = (einf.g. ∨ mehrf.g.) ∧ Hoehe ₂ = (mittel ∨ tief) ∧ Hoehe ₁ = (tief ∨ mittel ∨ steigend ∨ konson.)
49	5	100.0%	Coda ₁ = (einf.g. ∨ mehrf.g.) ∧ Hoehe ₃ = (tief ∨ steigend) ∧ Coda ₃ = (offen ∨ - ∨ ambiv. ∨ mehrf.g.) ∧ Hoehe ₁ = (mittel ∨ steigend)
50	5	100.0%	Onset ₃ = (n.vorh. ∨ -) ∧ Gespanntheit ₂ = (gespannt ∨ Schwa) ∧ Coda ₁ = (einf.g. ∨ mehrf.g.) ∧ Hoehe ₃ = (tief ∨ mittel ∨ steigend) ∧ Coda ₃ = (offen ∨ - ∨ ambiv. ∨ mehrf.g.)
51	6	100.0%	Gespanntheit ₁ = gespannt ∧ Coda ₁ = (einf.g. ∨ mehrf.g.) ∧ Coda ₃ = (einf.g. ∨ offen ∨ mehrf.g.) ∧ Hoehe ₁ = (tief ∨ hoch ∨ konson.)
52	14	78.6%	Gespanntheit ₂ = ungesp. ∧ Hoehe ₂ = (hoch ∨ mittel ∨ tief ∨ steigend) ∧ Hoehe ₃ = (- ∨ hoch ∨ steigend) ∧ Hoehe ₁ = (tief ∨ konson.) ∧ Coda ₂ = (offen ∨ einf.g. ∨ mehrf.g.)
53	7	85.7%	Onset ₃ = vorh. ∧ Coda ₁ = (einf.g. ∨ mehrf.g.) ∧ Hoehe ₃ = (tief ∨ - ∨ hoch ∨ steigend) ∧ Coda ₃ = (offen ∨ - ∨ ambiv. ∨ mehrf.g.) ∧ Hoehe ₁ = (tief ∨ konson.)
54	5	100.0%	Gespanntheit ₁ = gespannt ∧ Gespanntheit ₂ = (gespannt ∨ Schwa) ∧ Hoehe ₂ = (hoch ∨ tief ∨ steigend) ∧ Hoehe ₃ = (tief ∨ - ∨ hoch ∨ steigend)
55	11	81.8%	Coda ₁ = (einf.g. ∨ mehrf.g.) ∧ Hoehe ₂ = mittel ∧ Coda ₃ = (offen ∨ - ∨ ambiv. ∨ mehrf.g.) ∧ Hoehe ₁ = (tief ∨ mittel ∨ steigend ∨ konson.) ∧ Coda ₂ = (einf.g. ∨ ambiv. ∨ mehrf.g.)
56	3	100.0%	Onset ₃ = (n.vorh. ∨ -) ∧ Gespanntheit ₂ = (gespannt ∨ Schwa) ∧ Coda ₁ = (einf.g. ∨ mehrf.g.) ∧ Hoehe ₂ = (hoch ∨ tief ∨ steigend) ∧ Coda ₃ = (einf.g. ∨ offen ∨ mehrf.g.)
57	4	100.0%	Gespanntheit ₁ = gespannt ∧ Hoehe ₂ = (mittel ∨ tief) ∧ Hoehe ₃ = mittel ∧ Coda ₃ = (einf.g. ∨ offen ∨ mehrf.g.) ∧ Hoehe ₁ = (tief ∨ hoch ∨ konson.) ∧ Coda ₂ = (offen ∨ mehrf.g.)

Betonung final (Fortsetzung)			Onset Hoehe Gespanntheit Coda
Nr.	Fälle	Precision	Regel
58	2	100.0%	Coda_1 = (einf.g. \vee mehrf.g.) \wedge Hoehe_2 = (hoch \vee tief \vee steigend) \wedge Hoehe_3 = mittel \wedge Hoehe_1 = (tief \vee konson.)
59	2	100.0%	Onset_3 = (n.vorh. \vee -) \wedge Gespanntheit_2 = (gespannt \vee Schwa) \wedge Coda_1 = (einf.g. \vee mehrf.g.) \wedge Hoehe_3 = (- \vee hoch \vee steigend) \wedge Coda_3 = (einf.g. \vee offen \vee mehrf.g.) \wedge Hoehe_1 = (tief \vee mittel \vee steigend \vee konson.)
60	5	80.0%	Gespanntheit_1 = gespannt \wedge Onset_3 = (n.vorh. \vee -) \wedge Hoehe_2 = tief \wedge Coda_3 = (offen \vee - \vee ambiv. \vee mehrf.g.) \wedge Hoehe_1 = (tief \vee mittel \vee steigend \vee konson.) \wedge Coda_2 = (offen \vee mehrf.g.)
61	4	75.0%	Coda_1 = (offen \vee mehrf.g.) \wedge Hoehe_2 = tief \wedge Hoehe_1 = (tief \vee mittel \vee steigend \vee konson.) \wedge Coda_2 = (ambiv. \vee mehrf.g.)
62	2	100.0%	Gespanntheit_1 = (ungesp. \vee Schwa \vee konson.) \wedge Coda_1 = einf.g. \wedge Hoehe_1 = (mittel \vee steigend) \wedge Coda_2 = (ambiv. \vee mehrf.g.)
63	1	100.0%	Hoehe_2 = tief \wedge Hoehe_3 = (tief \vee steigend) \wedge Coda_3 = (einf.g. \vee mehrf.g.)
64	1	100.0%	Onset_3 = vorh. \wedge Coda_1 = (einf.g. \vee mehrf.g.) \wedge Hoehe_2 = (hoch \vee tief \vee steigend) \wedge Coda_3 = (- \vee ambiv. \vee mehrf.g.) \wedge Hoehe_1 = (tief \vee hoch \vee konson.)
65	2	100.0%	Gespanntheit_1 = gespannt \wedge Gespanntheit_2 = (gespannt \vee Schwa) \wedge Coda_1 = (einf.g. \vee mehrf.g.) \wedge Hoehe_3 = (tief \vee - \vee hoch \vee steigend) \wedge Coda_3 = offen
66	1	100.0%	Onset_3 = vorh. \wedge Hoehe_3 = (tief \vee - \vee hoch \vee steigend) \wedge Coda_3 = (- \vee ambiv. \vee mehrf.g.)
67	1	100.0%	Gespanntheit_1 = (ungesp. \vee Schwa \vee konson.) \wedge Coda_1 = (offen \vee mehrf.g.) \wedge Hoehe_2 = tief \wedge Hoehe_3 = (- \vee hoch \vee steigend) \wedge Coda_3 = (offen \vee - \vee ambiv. \vee mehrf.g.)

4.5 Beziehungen zu verbreiteten phonologischen Regeln

In diesem Abschnitt werden die gelernten Regeln in Beziehung gebracht mit Aussagen, die in der phonologischen Literatur zu diesem Untersuchungsgebiet getroffen wurden. Die Aussagen sind [Jess94] entnommen, wo verschiedene in

der phonologischen Literatur vertretene Ansichten aufgezeigt sind. Da die Aussagen von verschiedenen Autoren stammen, ist die Terminologie nicht immer einheitlich. Beispiele dafür sind, daß die Unterscheidung zwischen „leichten“ und „schweren“ Silben nicht von allen Autoren gleich vorgenommen wird. Mitunter werden auch Silbengrenzen nicht berücksichtigt und somit Coda der vorangehenden und Onset der nachfolgenden Silbe als ein zwischen zwei Nuklei liegender Konsonantenkomplex betrachtet. Ein weiteres Problem ist die von verschiedenen Autoren unterschiedlich gehandhabte Extrametrikalität, also die Nichtberücksichtigung des letzten Segmentes oder der letzten Segmente bei der Betonungszuweisung. Schließlich gründen manche Autoren ihre Analysen auf andere Transkriptionen. Hier wird versucht, den Aussagen der Phonologen bei der Betrachtung möglichst nahezukommen. Dabei ist jedoch zu bedenken, daß dies aufgrund der unterschiedlichen Ausgangsposition nicht immer überzeugend gelingen kann. Zudem sollten die Ergebnisse wegen des verwendeten kleinen Korpus nicht überbewertet werden. Die nicht wörtlich zitierten Aussagen sind jeweils typographisch abgehoben. Daran schließt sich eine Betrachtung der Beziehung der Aussage zu den gelernten Regeln und gelegentlich zu den Fällen des Korpus an.

„Drei Silben Regel: Nur die letzten drei vollen Silben können betont werden.“

Diese Aussage steht mit den Daten nicht im Widerspruch. Alle Fälle des Korpus sind auf einer der letzten drei Silben betont und somit auf einer der letzten drei vollen Silben.

„Die Betonungszuweisung findet von rechts nach links statt.“

Diese Annahme wird durch die Ergebnisse der ersten Lernreihe teilweise unterstützt. Darin wurde gezeigt, daß die Information von weiter als drei Silben vom Ende entfernten Silben nur vernachlässigbar zur Verbesserung der Regeln beiträgt. Allerdings ging der Lernalgorithmus nicht strikt linear vor, z. B. stützte sich die Entscheidung zugunsten einer finalen Betonung auf Merkmale von weiter links stehenden Silben (s.o.). Dies liegt im gewählten Lernverfahren begründet. Man könnte diese These näher untersuchen, indem man die Entscheidungsregeln getrennt für die einzelnen zu lernenden Konzepte durchführt und dabei „von links“ stammende Information explizit unzugänglich macht. Ist die Qualität der Regeln durch diese Einschränkung wesentlich schlechter, wäre dies ein Argument gegen die gemachte Aussage.

„Wenn nicht aufgrund anderer Prinzipien eine Betonungszuweisung auf die finale oder penult Silbe erfolgt, so erhält die antepenult Silbe die Wortbetonung.“

Der gewählte Algorithmus erzeugt keine „Default–Werte“, so daß eine solche Regel nicht direkt erwartet werden kann. Interessanterweise ergab sich bei einem von mir durchgeführten Experiment mit dem ID3-Algorithmus¹ [Quin86] eine Verbesserung der Vorhersagegenauigkeit, wenn in aufgrund der Lerndaten unentschiedenen Fällen antepenult vor penult vor finaler Betonung bevor-

¹Dazu wurde eine Prolog-Implementation von Chris Mellish verwendet, die ich um die Fähigkeit, Rauschen in den Trainingsdaten zu verarbeiten, erweitert habe.

zugt wird. Dies entspricht eher der folgenden Regel: Wenn zwei Prinzipien zwei verschiedene Betonungen gleichgut rechtfertigen, erhält bevorzugt die weiter vorn stehende Silbe die Betonung. In Fällen, für die aus den Trainingsdaten keine Entscheidungen ableitbar sind, gab das blinde Raten einer penulten Betonung die besten Ergebnisse und nicht das Raten der höherfrequenten finalen Betonung. Der oben genannte Default einer antepenulten Betonung schnitt am schlechtesten ab. Die gemachte Aussage hat aber in den phonologischen Theorien durchaus ihre Berechtigung, denn viele machen sonst nur Aussagen über finale oder penulte Betonung. Würde in den maschinell gelernten Regeln versucht, nur die Konzepte „final“ und „penult“ abzudecken, wäre sicherlich ein der gemachten Aussage entsprechendes Vorgehen am erfolgreichsten.

„Die Betonung fällt auf die am weitesten rechts stehende schwere Silbe“

In der phonologischen Literatur finden sich unterschiedliche Ansichten darüber, was eine schwere und was eine leichte Silbe ist. Einigkeit besteht allenfalls darin, Silben mit einem Schwa oder silbischen Konsonanten im Nukleus als leicht zu bezeichnen. Geschlossene Silben und Silben mit Diphthong werden immer als schwer betrachtet. Unterschiedliche Ansichten bestehen hauptsächlich bei Silben vom Typ Vg und bei Silben mit ambisyllabischer Coda. Vennemann rechnet Vg Silben zu den schweren Silben [Venn92], viele andere würden Silben diesen Typs als leicht bezeichnen. Ambisyllabizität wird von manchen Autoren für das Deutsche generell abgelehnt [Hall92]. Andere betrachten ambivalente Silben als geschlossen und damit schwer. Mitunter werden Silben vom Typ VuCC1 oder VgC1 auch als „superschwer“ bezeichnet, die Schwasilben als „superleicht“.

Die These ist in den Regeln nicht klar erkennbar. Es treten jedoch häufig Tendenzen zugunsten der Aussage auf, z.B. in den Regeln für die Betonung **final** bei der ersten Codierung. Alle Regeln mit großer Abdeckung verlangen, daß die letzte Silbe superschwer oder schwer sein muß.²

Gelegentlich wird für den Nukleusvokal auch Länge anstatt Gespanntheit als unterscheidendes Merkmal für die Typenzuordnung unterstellt, siehe [Jess94]. Versteht man darunter nur dieselben Vokalqualitätsunterschiede wie bei der Unterscheidung gespannt/ungespannt, gilt das oben gesagte auch hier. Möchte man jedoch weitergehen und die hier vorgenommene Längenunterscheidung ebenfalls in die Beschreibung von Gewicht aufnehmen, ist es sinnvoll anzunehmen, daß die Betonung auf die letzte lange und damit schwere Silbe fällt. Beispielsweise könnte man schwere Silben als solche festlegen, die lang und/oder mehrfach geschlossen³ sind. Wie die Regeln der zweiten Codierung zeigen, wären mit einem solchen Gewichtssystem sehr zuverlässige Aussagen möglich. Dieser

²Das unverständliche Einfügen von C0 als möglichen Wert für den Typ der letzten Silbe muß als Artefakt des Algorithmus angesehen werden. Im Korpus sind natürlich keine Wörter vorhanden, die final betont sind und bei denen der letzte Nukleus aus einem Schwa gebildet wird. Da es sich um einen Algorithmus handelt, der kein Hintergrundwissen wie „eine Schwasilbe kann niemals betont werden“ besitzt, er andererseits aber auch zu Generalisierungen gezwungen wird, sind solche vom linguistischen Standpunkt unverständlichen Zusammenfassungen nie ausgeschlossen.

³Mehrere Segmente in der Coda tragen wie verlängerte Nuklei auch zu einer insgesamt langen Silbe bei.

Gewichtsdefinition stehen jedoch zumindest zwei methodische Schwächen entgegen: erstens der zyklische Gebrauch der Länge als Auslöser und phonetische Realisierung der Betonung und zweitens, daß sich ein solches Gewichtssystem für das Deutsche von Gewichtssystemen anderer Sprachen unterscheiden würde.

„Penult Regel: Die Betonung fällt nicht vor eine schwere vorletzte Silbe.“

Diese Aussage müßte sich idealerweise in den Regeln bemerkbar machen, die antepenulte Betonung vorhersagen, etwa durch Einschränkung von **Typ_2** auf leichte Silben. In den gelernten Regeln zeigt sich auch hier wieder keine klare Bestätigung. Allenfalls können die Regeln 20 und 21 der Typcodierung als Unterstützung für die Aussage angeführt werden. Sie sagen penulte Betonung für superschwere penulte Silben voraus. Deren Abdeckung ist jedoch mit 5 Fällen sehr gering.

„Wenn nicht schon finale Betonung feststeht, impliziert eine geschlossene penulte Silbe penulte Betonung.“

Diese Aussage hat eine schöne Entsprechung in der Regel 5 der Codierung in Einzelmerkmale, die lautet:

$$(\text{Laenge_1} = \text{kurz}) \wedge (\text{Coda_2} = (\text{einf.g.} \vee \text{ambiv.} \vee \text{mehrf.g.})) \rightarrow (\text{Betonung} = \text{penult})$$

Mit dem zusätzlichen Ausschluß langer finaler Silben greift diese Regel bei 54 der 260 Fälle, von denen sind 75,9% tatsächlich penult betont. Wie man sieht, schlägt der Algorithmus ambivalente Silben, also Silben mit ambisyllabischer Coda, den geschlossenen Silben zu.

Für penulte Betonung scheint die Struktur der Coda der penulten Silbe in der Tat relevant zu sein: Bei der dritten Codierung mißt der Algorithmus diesem Merkmal die größte Wichtigkeit bei, bei der zweiten Codierung stuft er lediglich die Längenmerkmale der beiden letzten Silben als wichtiger ein.

Die mitunter in der Literatur angetroffene Aussage, daß eine offene penulte Silbe antepenulte Betonung nach sich zieht, kann in einigen Regeln der dritten Codierung nachvollzogen werden. Von den 17 Regeln enthalten 6 den Term **Coda_2 = offen**, andere Werte für dieses Merkmal treten nicht auf. Die von diesen 6 Regeln abgedeckten Fälle sind zu 100% antepenult betont.

„Reduzierte-Silbe-Regel: Eine reduzierte letzte Silbe mit Onset legt die Betonung auf die letzte volle Silbe fest.“

Im Korpus sind keine Fälle enthalten, die mit zwei reduzierten Silben enden. Deshalb ist penulte Betonung zu erwarten, wenn die letzte Silbe eine durch einen Onset eingeleitete Schwasilbe ist. Neben den 16 Fällen, die dieser Aussage entsprechen, enthält das Korpus auch zwei Ausnahmen, nämlich „Ameise“ und „Brosame“. Diese werden vom Algorithmus erkannt und durch Einschränkung der erlaubten Typen der zweit- und drittletzten Silbe ausgeschlossen. Die Regeln

10 und 13 der Codierung in Typen lauten:

$$\begin{aligned}(\text{Typ_1} = \text{C@}) \quad \wedge \\ (\text{Typ_2} = (\text{Vg} \vee \text{Vu} \vee \text{VuCC1} \vee \text{C@} \vee \text{VgC1})) \quad \wedge \\ (\text{Typ_3} = (\text{VuC} \vee \text{Vu} \vee - \vee \text{V_V} \vee \text{VuCC1} \vee \text{VgC1})) \rightarrow (\text{Betonung} = \text{penult})\end{aligned}$$

Die Aussage wird also im wesentlichen von den im Korpus vorhandenen Fällen und den gelernten Regeln bestätigt.

$$\begin{aligned}(\text{Typ_1} = \text{C@}) \quad \wedge \\ (\text{Typ_2} = (\text{Vg} \vee \text{VuC} \vee \text{C@} \vee \text{VgC1})) \rightarrow (\text{Betonung} = \text{penult})\end{aligned}$$

„Die Silbenonsets beeinflussen nicht die Betonungszuweisung.“

Diese Aussage wird im wesentlichen von der Untersuchung bestätigt. Die Onsetmerkmale spielen bei der Typcodierung keine Rolle, da der Typ *@* nicht im Korpus auftritt.⁴ Bei der zweiten Codierung in Einzelmerkmale wird keines der Onsetmerkmale verwendet. Lediglich bei der am schlechtesten abschneidenden dritten Codierung trägt der Onset der drittletzten Silbe zur Vorhersage antepenulter Betonung bei. Dabei tritt in den Regeln keine Präferenz für vorhandenen oder nicht vorhandenen Onset auf. M. E. modelliert der Algorithmus mit diesem Merkmal nur einzelne Fälle, die mit anderen Merkmalen genauso gut oder schlecht modellierbar wären. Dies äußert sich auch in der geringen Abdeckung der betroffenen Regeln. Eine separate Lernreihe, bei der die Onsetattribute generell vorenthalten werden, könnte Klarheit über die Fragestellung erbringen.

„Superschwere-letzte-Silbe-Regel: Wenn die letzte Silbe entweder einen gespannten Vokal hat und von mindestens einem Konsonant geschlossen ist oder wenn sie einen ungespannten Vokal besitzt und von mindestens zwei Konsonanten geschlossen ist, wird final betont.“

Diese Aussage, die sich auf manchmal auch als superschwer bezeichnete Silben bezieht, findet keine unmittelbare Entsprechung in den bisher gezeigten Regeln. Sie wird jedoch identisch vom Algorithmus produziert, wenn man die Roughness auf 0,90 einstellt und die Typcodierung verwendet. Als einzige Regeln, die finale Betonung vorhersagen, werden produziert:

$$(\text{Typ_1} = \text{VgC1}) \rightarrow (\text{Betonung} = \text{final})$$

$$(\text{Typ_1} = \text{VuCC1}) \rightarrow (\text{Betonung} = \text{final})$$

Die Regeln greifen bei 42 resp. 30 Korpusfällen, von denen 64,3% resp. 76,7% final betont sind. Die stärkere Neigung zur Betonung der *VuCC1*-Silben ist in Übereinstimmung mit [Venn90].

⁴Dies ist zum Teil auch durch die verwendete Transkription begründet. Mangold transkribiert beispielsweise „Otilie“ oder „Familie“ mit einem unsilbischen /i/, das dadurch zu einem Onset für den Schwa wird und eine Teilung des Wortes in drei und nicht vier Silben sowie penulte und nicht antepenulte Betonung zur Folge hat. Es existieren auch von dieser Sicht abweichende Transkriptionen, die auch Auswirkungen auf die Reduzierte-Silbe-Regel hat.

Auch bei den Regeln, die mit Roughness 0,50 erzeugt wurden, kann man Entsprechungen finden. Bei den auf viele Fälle anwendbaren Regeln, die gute Vorhersagen versprechen, Regeln 24–26, werden fast nur superschwere Silbentypen für die letzte Silbe gefordert. Allerdings bestehen bei ihnen ebenfalls Einschränkungen für die zweit- und drittletzte Silben. Auffällig sind allenfalls noch die Diphthongsilben, die möglicherweise auch als superschwer angesehen werden können.

„Eine finale VuC-Silbe erhält die Betonung vs. eine finale VuC-Silbe erhält die Betonung nicht.“

Weder für die eine noch die andere Position sind aus den Regeln stichhaltige Belege zu finden. Für finale Betonung kann Regel 32 der Typcodierung angeführt werden, für die Gegenposition Regel 16. Beide Regeln verlangen explizit Zweisilbigkeit indem sie `Typ_3 = –` fordern. Eine weitere Gemeinsamkeit der beiden Regeln ist die recht große Abdeckung bei geringer Genauigkeit. Die Regeln greifen bei 17 bzw. 36 Fällen, darunter sind aber nur 64,7% finale bzw. 61,1% penulte Betonungen. Betrachtet man die Verteilung im Korpus, so ist dies wie auch die in der phonologischen Literatur angetroffene Uneinigkeit nicht verwunderlich. Unter den auf VuC endenden Einträgen sind 39 final, 34 penult und 21 antepenult betont. M. E. ist die Uneinigkeit letztendlich in der Annahme begründet, daß die Betonungszuweisung von rechts nach links verläuft. Ohne diese Annahme entsteht kein unmittelbarer Zwang, sich für eine der beiden Positionen zu entscheiden und die jeweils anders betonten Fälle als Ausnahme zu betrachten. Wie man an den gelernten Regeln sieht, ist durch differenzierende zusätzliche Einschränkung der Struktur weiter vorne stehender Silben möglicherweise eine bessere Repräsentation zu erzielen.

„Eine finale Vg-Silbe erhält die Betonung vs. Eine finale Vg-Silbe erhält die Betonung nicht.“

Die gelernten Regeln sprechen eher dafür, daß finale Vg-Silben nicht betont sind. Allerdings ist diese Tendenz nicht deutlich ausgeprägt. Die Regeln für finale Betonung, die eine finale Vg zulassen, die Regeln 27, 29, 30 und 33, decken nur sehr wenige Fälle ab, während von den entsprechend aufgebauten Regeln, die penulte Betonung prognostizieren, mehr Fälle abgedeckt werden, unter anderen die Regeln 9, 12, 14 und 15. Auch die Verteilung im Korpus legt eine Präferenz penulter Betonung nahe, wenngleich hier die Asymmetrie nicht so stark ausgeprägt ist. Die Verteilung beträgt 21 finale zu 28 penulte zu 2 antepenulte Fälle.

4.6 Zusammenfassung und Bewertung

In der ersten Lernreihe wurde untersucht, wie gut die vom Lernalgorithmus erzeugten Regeln die Fälle des Korpus abdecken. Dabei zeigte sich, daß die Information über die weiter als drei Silben vom Wortende entfernten Silben keine wesentliche Verbesserung der Abdeckung erbringt. Außerdem konnte festgestellt werden, daß die Codierung in Einzelmerkmale eine bessere Abdeckung

ermöglicht als eine Codierung in Typen. Dieses zweite Ergebnis wurde in den nachfolgenden Lernreihen näher analysiert.

In der zweiten und dritten Lernreihe wurden für drei unterschiedliche Codierungen Regeln generiert und die zu erwartenden Erkennungsraten für noch nicht gesehene Testfälle bestimmt. Diese sind in Tafel 4.3 zusammengefaßt.

	Erste Codierung	Zweite Codierung	Dritte Codierung
Typ	+		
Onset		+	+
Hoehe		+	+
Laenge		+	
Gespanntheit		+	+
Coda		+	+
Erkennungsrate	53,462%	79,616%	47,692%

Tafel 4.3: Zu erwartende Erkennungsraten der drei untersuchten Codierungen.

Das in der ersten Codierung verwendete Merkmal **Typ** ist aus den Merkmalen der dritten Codierung abgeleitet. Die erste Codierung stellt also eine Generalisierung der dritten Codierung dar. Die dritte Codierung wiederum ist eine Generalisierung der zweiten Codierung, da die Merkmale der dritten Codierung eine echte Teilmenge der Merkmale der zweiten Codierung bilden. Diese Abstufung spiegelt sich in der Feinheit der durch die Merkmale induzierten Äquivalenzrelation wider. Seien R , S , T die von den Codierungen 1–3 induzierten Äquivalenzrelationen und Q die vom Entscheidungsattribut induzierte Äquivalenzrelation. Dann gilt:

$$\gamma(S, Q, 0) = 0,9385 > \gamma(T, Q, 0) = 0,6962 > \gamma(R, Q, 0) = 0,4115.$$

Aufgrund dieser Beziehung ist zu erwarten, daß die obere und untere Approximation (siehe Kapitel 2.1) die zu lernenden Konzepte bei der zweiten Codierung am besten beschreiben. Bei der dritten Codierung ist wegen des Wegfalls des Längenmerkmals eine schlechtere Approximation und bei der Codierung durch Typen eine noch schlechtere Approximation zu erwarten.

Bei der Betrachtung der erzielten Erkennungsraten erkennt man jedoch, daß mit der ersten Codierung bessere Erkennungsraten erzielt werden als mit der dritten Codierung. Dieses unerwartete Ergebnis zeigt aber nur, daß Aussagen über die Lernmenge nicht notwendigerweise auch für die noch nicht gesehene Testfälle zutreffen müssen, denn die Rough Set Theorie macht lediglich Aussagen über die betrachteten Fälle und nicht über die noch nicht gesehene Testfälle. Dieses Problem betrifft alle maschinellen Lernverfahren gleichermaßen und unterstreicht die Wichtigkeit von Untersuchungen mit getrennten Lern- und Testmengen. Bei einer kleinen Zahl von Trainingsfällen ist es schwieriger, unter vielen Merkmalen die relevanten herauszusuchen als aus wenigen Merkmalen. Es ist dann möglich, daß sich die Vorhersage auf Merkmale stützt,

die zwar zufällig die Trainingsfälle gut beschreiben, aber für Voraussagen wenig relevant sind. Wenn man größere Lernmengen verwendet, werden solche irreführende Merkmale immer unwahrscheinlicher, es erscheinen nur noch die tatsächlich relevanten Merkmale erfolgversprechend. M. E. ist das Korpus für die dritte Codierung zu klein. Die vom Algorithmus mit den wenigen Trainingsdaten erzielten Generalisierungen stehen hinter den Generalisierungen zurück, die durch die Abbildung der Einzelmerkmale auf die Silbentypen erbracht werden. Ein weiteres Indiz für die unzureichende Anzahl von Trainingsbeispielen für die dritte Codierung ist, daß die gewonnenen Regeln meist deutlich weniger Fälle abdecken als die der beiden anderen Codierungen. Daraus resultiert auch die deutlich höhere Regelanzahl. Es ist möglich, daß die Erkennungsraten für die dritte Codierung über denen der ersten liegen werden, wenn zum Lernen ein größeres Korpus verwendet wird. Klarheit über diese Vermutung kann erst eine Untersuchung mit deutlich mehr Daten erbringen.

Man könnte nun vermuten, daß die zusätzliche Betrachtung des Längenmerkmals bei der zweiten gegenüber der dritten Codierung die Auswahl erfolgreicher Regeln noch mehr erschwert. Die mit Abstand besten Erkennungsraten der zweiten Codierung von nahezu 80% zeigen, daß das Vorhandensein der Längeninformation die Betonungszuweisung wesentlich erleichtert. Dies zeigt sich auch in der wesentlich geringeren Regelanzahl, die bei dieser Codierung nötig ist. Zudem sind die Regeln sehr einfach aufgebaut. Die aus den Regeln erkennbare Tendenz ist, daß bevorzugt lange Silben betont sind. Unter den Regeln, die antepenulte Betonung voraussagen, wird nahezu ausschließlich ein langer Nukleusvokal der vorvorletzten Silbe vorausgesetzt. Die penulte Betonung voraussagenden Regeln verlangen sämtlich einen langen penulten oder einen kurzen finalen Nukleusvokal. Die Regeln für finale Betonung setzen fast alle voraus, daß der Nukleusvokal der penulten Silbe kurz und der der letzten Silbe lang ist. Auf einen langen Nukleusvokal kann bei den letztgenannten Regeln nur verzichtet werden, wenn die Coda der finalen Silbe mehrfach geschlossen ist, was wiederum auch einen Beitrag zur Längung der betonten Silbe darstellt.⁵

Die Verwendung des Längenmerkmals bei den Regeln ist wegen der starken Korrelation von Länge und Betonung aus methodologischer Sicht nicht unproblematisch. In der Phonologie wird meist die Ansicht vertreten, daß zunächst die Betonungszuweisung erfolgt, dann die Betonung geeignet phonetisch realisiert wird. Wie im Kapitel über die akustischen Korrelate dargelegt wurde, scheint die Längung dabei die wichtigste Rolle zu spielen, wenngleich in der Literatur häufig andere Ansichten vertreten werden. Wenn wir die Länge als phonetische Realisierung der Betonung ansehen, können wir sie nicht gleichzeitig als wichtigstes Merkmal für die Betonungszuweisung verstehen.

Die Auswahl der besten Codierung hängt nach dem bis jetzt gesagten ganz klar von der Aufgabenstellung ab, die damit verfolgt wird. Wenn man versucht, phonologische Regeln zur Betonungszuweisung zu finden, scheidet die zweite Codierung aus. Die durchgeführten Untersuchungen rechtfertigen nicht

⁵Lediglich bei Regel 12 genügt bereits einfache Geschlossenheit. Sie weist aber auch die geringste Genauigkeit unter den Regeln dieser Codierung auf.

die Verwendung der Einzelmerkmale ohne die Länge gegenüber der traditionellen Verwendung von Silbentypen. Möglicherweise muß diese Ansicht jedoch nach Anwendung des Lernverfahrens auf ein größeres Korpus revidiert werden. Wenn man andererseits auf linguistisch fundierte Regeln keinen Wert legt, wäre man töricht, auf die Information aus dem Längenmerkmal zu verzichten, z. B. bei Text-to-Speech-Anwendungen. In der deutschen Orthographie kann man mitunter die Längung des Nukleus erkennen, beispielsweise an einem auf den Vokal folgenden „h“, oder bei manchen aus dem Französischen stammenden Wörtern, z. B. „Plateau“ vs. „Plato“.

Die generierten Regeln weisen bei allen Codierungen interessante Verbindungen zu in der phonologischen Literatur angetroffenen Ansichten auf. Einige Ansichten haben mehr oder weniger direkte Entsprechungen in den gelernten Regeln gefunden. Dabei hat sich herausgestellt, daß die maschinell gefundenen Regeln häufig komplizierter aufgebaut sind als die von Phonologen gemachten Aussagen. Meist weisen die maschinell gefundenen Regeln zusätzliche Antezedenzen auf. Die vorgestellten Regeln wurden mit den Parametereinstellungen erzeugt, mit denen die besten Erkennungsraten zu erwarten sind. Man kann auch die Parameter des Algorithmus so variieren, daß die Regeln in ihrer Komplexität mit den zu vergleichenden Aussagen übereinstimmen. Dieser Trade-off zwischen guter Vorhersagegenauigkeit und Einfachheit der Regeln ist bei dem Rough-Set-basierten Lernverfahren durch Verändern des Roughnessparameters gut nachzuvollziehen, wie die Untersuchung der Superschwere-letzte-Silbe-Regel gezeigt hat. Auch manche andere Lernverfahren besitzen vergleichbare Eigenschaften. M. E. ist das Vorhandensein einer solchen Eigenschaft wichtig für die Auswahl von Verfahren des Maschinellen Lernens für die linguistische Forschung. Bei der Untersuchung linguistischer Probleme ist man fast immer mit Ausnahmen konfrontiert, die man möglicherweise gar nicht genauer abdecken möchte zugunsten einer einfacheren und eleganteren Theorie. Zudem haben die Untersuchungen gezeigt, daß symbolische Lernverfahren für die theorieorientierte Forschung große Vorteile besitzen. Sie erlauben die Untersuchung größerer Datenmengen mit einem vertretbaren Zeitaufwand. Das Lernen eines Klassifikators dieser Arbeit dauert nie länger als wenige Minuten auf einem gewöhnlichen PC (80286 12Mhz). Der wichtigste Vorteil für Untersuchungen dieser Art ist die Interpretierbarkeit der Resultate. Diese können mit bereits existierenden Überlegungen in Beziehung gesetzt werden. Gegenüber der Überprüfung von Hypothesen durch Formulierung von Datenbankabfragen und der statistischen Bewertung der Ausgabe hat das Verfahren natürlich den Vorteil, daß neue Gesetzmäßigkeiten aufgedeckt werden können. Die vorhandene Datenbasis ist m. E. jedoch etwas zu klein, um solche neuen Thesen zu rechtfertigen. Da nun die erforderlichen Werkzeuge zur Aufbereitung bereitstehen, steht einer Untersuchung einer großen Datenmenge nichts entgegen, sobald entsprechende Rohdaten, also Transkriptionen mit Silbentrennungen, verfügbar sind.

Kapitel 5

Abschließende Bemerkungen und Ausblick

Thema dieser Arbeit war die Anwendung von Verfahren des Maschinellen Lernens auf Fragestellungen aus der linguistischen Forschung. Damit wurden zwei Ziele verfolgt: Zum einen sollte untersucht werden, ob und wie der Einsatz von automatischen Lernverfahren auf offene oder umstrittene Forschungsprobleme sinnvoll sein kann, zum anderen bestand auch ein Interesse an den inhaltlichen Ergebnissen der Untersuchungen. Die Ergebnisse aus dem phonetischen und dem phonologischen Teil der Arbeit sind in den Kapiteln 3.4, 3.6 und 4.6 zusammengefaßt und diskutiert worden.

Zusammenfassend kann gesagt werden, daß der Einsatz maschineller Lernverfahren gut geeignet ist, linguistische Hypothesen empirisch zu überprüfen. Die phonetische Untersuchung hat unter anderem erbracht, daß die Grundfrequenz weniger zur Unterscheidung von Betonung beiträgt als allgemein angenommen wird. Dies gilt sogar bei Vorliegen eines Fokusakzents.¹ Empirische Untersuchungen können also Anstöße zum Überdenken linguistischer Positionen erbringen. Speziell bei dieser Untersuchung ist allerdings festzustellen, daß die gefundenen Ergebnisse auch mit Mitteln der Statistik oder Visualisierung erbracht werden können, da die untersuchten Daten sehr eindeutig sind. Bei den phonologischen Untersuchungen ist es aufgrund verschiedener Faktoren wie starkem Rauschen in den Daten oder hoher Dimensionalität des Problems bedeutend schwieriger, mit solchen Mitteln zu den gezeigten Ergebnissen zu gelangen. Für viele der in der phonologischen Literatur verbreiteten Regeln zur Wortbetonung fanden sich Entsprechungen in den maschinell erzeugten Regeln. So wurde beispielsweise die sog. Superschwere-letzte-Silbe-Regel oder die Verwendung von Silbentypen durch die Untersuchungen bestätigt. Vergleichbare Resultate hätten allenfalls durch entsprechende Datenbankabfragen beziehungsweise durch Implementation von Regeln und Anwendung auf ein Korpus erzielt werden können. Eine solche Vorgehensweise ermöglicht jedoch nur die Überprüfung

¹In der Literatur wird nicht nur die Meinung vertreten, die Grundfrequenz sei das wichtigste Korrelat der Betonung. Es gibt auch Autoren, die zeitliche Strukturierung oder Intensität als wichtigste Faktoren ansehen, s. [Nöth91].

von Überlegungen dessen, der das Verfahren anwendet. Maschinelles Lernen hat dagegen die Potenz, neue Einsichten zu erbringen.

In diesem Rahmen wurde auf die Formulierung neuer Hypothesen größtenteils verzichtet, da die Datenbasis relativ gering ist. Bei beiden Untersuchungsgebieten gab es das Problem zu kleiner Lernmengen, also das Grundproblem empirischer Untersuchungen, das weitere interessante Untersuchungen verhinderte. Bei den phonetischen Untersuchungen erscheint die Frage interessant, ob es möglich ist, Unterschiede zwischen einem Fokusakzent und einem Wortakzent an Nichtfokusposition festzustellen. Aufgrund der wenigen zur Verfügung stehenden Daten konnte eine Untersuchung nur indirekt erfolgen, indem untersucht wurde, ob nur mit den Grundfrequenzmerkmalen die in Fokusposition stehenden Wortakzente besser vorhersagbar sind als nicht in Fokusposition stehende. Eine direkte Untersuchung, bei der aus Merkmalen aus der Umgebung des Akzents das Fokusmerkmal vorhergesagt werden soll wäre sicherlich sinnvoller. Allerdings können von einer solchen Untersuchung aufgrund der widersprüchlichen Ergebnisse der Untersuchung mit den Differenzmerkmalen keine sehr klaren Ergebnisse erwartet werden.

Wenn man die Untersuchung mit größeren Datenmengen wiederholt, sollten bei der Erhebung verschiedene Punkte beachtet werden.

- Die betonte Silbe sollte als Nukleus einen Vokal mit verhältnismäßig kurzer intrinsischer Länge haben.
- Bei der Logatomisierung sollten Phoneme verwendet werden, die nicht schon im Wort auftreten.
- Bei der Auswertung sollten feinere Merkmale extrahiert werden.
- Eine sinnvolle Wortlänge für die Untersuchung ist zweisilbig (ausgewogene Verteilung der Konzepte) oder dreisilbig (durchschnittliche Silbenzahl).

Ein weiterer interessanter Ansatz ist die Erweiterung der Untersuchungsmethode. Die Modellierung des Grundfrequenzverlaufs und möglicherweise des Intensitätsverlaufs könnte mit einem Hidden Markov Model [Rabi89] vorgenommen werden, um eine bessere Repräsentation des Grundfrequenzverlaufs zu erzielen. Ein mit den Daten aus der Lernmenge trainiertes Modell gibt die Wahrscheinlichkeit aus, mit der der Grundfrequenzverlauf eines neuen Testfalls vom Modell emittiert wird. Dieser Wert kann als Eingabe für ein symbolisches Lernverfahren dienen, das die nicht kontinuierlich anfallende temporale Information und möglicherweise weitere Merkmale wie Range–Werte, Integral der Intensität etc. damit in Beziehung setzt. Wie für alle statistischen Verfahren stellt sich auch für diese hybride Untersuchungsmethode die Frage nach großen Datenmengen für das Training, wenngleich für eine solche Untersuchung aufgrund der geringen Dimensionalität des Observationsvektors eine gegenüber der Spracherkennung deutlich geringere Datenmenge ausreichen könnte. Da die vom Algorithmus produzierten Regeln bezüglich dem Längenattribut sehr einfach ausgefallen sind und die HMMs auch eine implizite Längenmodellierung

in den Transitionswahrscheinlichkeiten besitzen, ist vielleicht eine Modellierung aller Merkmale mit einem solchen Modell einfacher durchzuführen. Eine solche rein statistische Modellierung hat jedoch wieder den bereits erwähnten Nachteil nichtsymbolischer Ansätze, nicht einfach nachvollziehbar zu sein. Das Erkennen der Segmente anstelle von Betonung bei den Silben von Liliputaner wäre möglicherweise unentdeckt geblieben, wenn die Regeln nicht so leicht verständlich gewesen wären.

Bei den phonologischen Untersuchungen traten die Probleme mit zu geringen Daten beim Vergleich der Codierung in Typen mit der Codierung in Einzelmerkmale ohne das Längenmerkmal auf. Es wäre ein etwas besseres Abschneiden der letzten Codierung gegenüber der ersten Codierung zu erwarten gewesen, das sich aber nicht eingestellt hat. Die interessante Frage, die sich deshalb stellt, ist, ob der Algorithmus prinzipiell schlechtere Generalisierungen vornimmt als die, die sich in der phonologischen Literatur entwickelt haben, oder ob nur die wesentlich vergrößerte Anzahl nicht besonders guter Attribute mehr Trainingsdaten verlangt.

Auch das Suchen neuer guter Regeln ist eine herausfordernde und wohl sehr interessante Aufgabe. Allein die Untersuchung eines deutlich größeren Korpus kann hier zuverlässige Ergebnisse liefern. Am Institut für Maschinelle Sprachverarbeitung ist mit der CELEX-Datenbasis inzwischen eine umfangreiche, phonologisch und morphologisch aufbereitete Datenbasis vorhanden, die für eine solche erweiterte Untersuchung geeignet erscheint. Die Verarbeitung der Einträge der CELEX-Datenbasis sollte eigentlich kein Problem darstellen, da bei der Behandlung des Korpus für diese Arbeit darauf geachtet wurde, möglichst wenig manuell vorzunehmen, so daß jetzt geeignete Tools zur Verfügung stehen. Als Grundlage wird lediglich eine phonetische Transkription mit Silbentrennungen benötigt. Die Extraktion der Merkmale für den Lernalgorithmus kann dann maschinell durch ein verhältnismäßig leicht anzupaßbares PROLOG-Programm erfolgen. Beispielsweise ist die Eingabe der Wortliste mit den Transkriptionen durch eine Definite Clause Grammar beschrieben, so daß eine Anpassung an andere Eingabeformate problemlos möglich ist.

Idealerweise sollte die Transkription folgende Kriterien erfüllen:

1. Transkription in SAM-PA. Dies wird in den wenigsten Fällen ein Problem darstellen, da abzusehen ist, daß sich diese Notation in der ingenieurmäßig betriebenen Sprachverarbeitung durchsetzt. Auch die Tatsache, daß IPA nahezu 1 : 1 in SAM-PA abgebildet werden kann, ermöglicht die Verwendung von IPA-transkribierten Datenbeständen.
2. Silbentrennungen. Zur eindeutigen Zuordnung von Konsonanten zu einer Silbe benötigt die Umsetzung Information über die Silbengrenzen. Eine rein algorithmische Trennung ist hier nicht verwirklicht, meines Wissens gibt es dafür auch keine eindeutigen Richtlinien. Die Berücksichtigung des Maximum Onset Principle kann jedoch in nahezu allen Fällen die richtige Entscheidung erbringen. Für wirklich große, nicht getrennte Datenmengen sollte eine algorithmische Trennung erwogen werden, auch

wenn sie möglicherweise fehlerbehaftet ist.

3. Ambisyllabizität. Zur Kennzeichnung von ambisyllabischen Konsonanten erwartet die Umsetzung die in Kapitel 4.2.2 beschriebene Notation mit einem Unterstrich vor der Silbengrenze. Verfügt die Transkription nicht über diese Kennzeichnung, so wird eben nie die Merkmalsausprägung **ambiv.** für ein Coda-Merkmal erzeugt, ebenso nicht die Ausprägung **V(C)** für ein **Typ**-Merkmal. An der prinzipiellen Anwendbarkeit ändert sich dadurch nichts.
4. Ein Nukleus. Bei der Abarbeitung der Segmente durch die Grammatik wird sichergestellt, daß nur ein Nukleus pro Silbe existiert. Deshalb müssen unsilbische Vokale unbedingt als solche gekennzeichnet sein. Bei der Umsetzung werden diese unsilbischen Vokale wie Konsonanten behandelt. Zum Beispiel besteht in dem Wort „Ingenieur“ /In|Ze|'n[^]i2:~6/ der Onset der letzten Silbe aus /n[^]i/, der Nucleus aus /2:/ und die Coda aus /~6/. Diphtonge werden als eine Einheit betrachtet.

Das verwendete Lernverfahren hat sich prinzipiell als gut geeignet für die Bearbeitung der Fragestellungen erwiesen. Allerdings lag dies weniger an der spezifischen Ausprägung des Lernsystems als vielmehr an der Tatsache, daß es sich um ein symbolisches Verfahren handelte. Kurze Tests mit einfachen Versionen anderer symbolischer Lernverfahren (Cobweb [Fish87], AQ [Mich83], ID3 [Quin86]) lassen vermuten, daß mit diesen Verfahren vergleichbare Ergebnisse zu erzielen sind. Als Nachteil bei der Verwendung von DataLogic/R hat sich die zumindest in der vorliegenden Version unzureichende Dokumentation der in diesem System verwendeten Algorithmen erwiesen. Bei Untersuchungen wie den hier durchgeführten ist volle Kenntnis der Abläufe beim Lernen sehr wichtig, um die erzielten Ergebnisse richtig interpretieren zu können.

Anhang A

Durchschnittliche Silbenanzahl im Deutschen

Die 4:1 Verteilung zwischen unbetonten und betonten Silben für die Untersuchung des Wortakzents in Kapitel 3 scheint relativ extrem zu sein. Für das Deutsche wird für Lexika eine durchschnittliche Silbenzahl pro Wort von 2,55 berichtet [Ortm80, S. XL]. Berücksichtigt man die Textfrequenz, so erniedrigt sich der Wert auf 1,61 [Ortm80, S. XL]. Ich habe die durchschnittliche Silbenanzahl pro Wort im Deutschen grob abgeschätzt, um einen Überblick über die in einer potentiellen Anwendung auftretenden Silbenzahlen zu erhalten. Es stand ein Korpus aus 7.537.116 Wörtern Zeitungstext (Frankfurter Rundschau) zur Verfügung. Es handelt sich dabei um die Wörter aus den Dateien `data/eci1/ger03/*.eci` auf der „European Corpus Initiative Multilingual Corpus 1 (ECI/MCI)“ CD-ROM der Association for Computational Linguistics.

Da der Text nicht phonologisch aufbereitet ist, konnte ich die Anzahl nur grob nach unten abschätzen. Es fand keine aufwendige Analyse zur Erkennung der Silbengrenzen statt. Stattdessen wurde als Näherung für die Silbenanzahl die Anzahl der Vokalbuchstabenansammlungen, also zusammenstehende Buchstaben a, e, i, o, u, ä, ö, ü verwendet. Durch diese Vereinfachung ergibt sich eine untere Grenze für die Anzahl der Silben im Wort, da in der Schriftsprache davon ausgegangen werden kann, daß jede Silbe mindestens einen Vokalbuchstaben enthält, es aber durchaus sein kann, daß zwei, theoretisch auch noch mehr aufeinanderfolgende Silben nur eine Vokalbuchstabenansammlung besitzen. „Information“ würde demnach korrekt als viersilbig abgeschätzt [Mang90], „Frauen“ würde jedoch fälschlich als einsilbig eingestuft, da es nur eine Vokalbuchstabenansammlung besitzt. Die Näherung ist noch aus einem anderen Grunde ungenau: unter den Wörtern sind noch etliche Abkürzungen, beispielsweise „Tel.“, „od.“. Auch dieser Einfluß senkt die Anzahl weiter.

Meinen Abschätzungen zufolge besitzen Wörter im Deutschen durchschnittlich mehr als 1,98 Silben. Wenn man einsilbige Wörter, genauer: Wörter mit einer Vokalbuchstabenansammlung, wegläßt, für die eine Entscheidung betont/unbetont hinfällig ist, kommt man auf eine durchschnittliche Silbenzahl

von größer als 2,80. Verzichtet man auch noch auf die Hälfte der zweisilbigen Wörter, unterstellend, daß eine der Silben eine Schwasilbe ist, so erhält man eine durchschnittliche Silbenzahl von 3,08 Silben pro Wort. Möchte man realistische Verteilungen zwischen betonten und unbetonten Silben in den Trainingsdaten erhalten, müßte man demnach dreisilbige Wörter untersuchen oder die Daten durch zufälliges Streichen von unbetonten Silben solange reduzieren, bis sich etwa das Verhältnis 2 : 1 ergibt.

Die Abschätzungen sind plausibel, da Ortmann als Grundlage Wörter mit einer bestimmten Mindestauftretenshäufigkeit verwendet. Lange Wörter haben die Tendenz, seltener aufzutreten, was den höheren Wert meiner Untersuchung erklären würde. Zudem könnte man anführen, daß in einem Zeitungstext lange Wörter eher häufig auftreten. Die durchschnittliche Silbenzahl des Korpus, das in Kapitel 4 Verwendung fand, beträgt 2,69. Die gefundene Silbenanzahl im Zeitungstext ist auch mit dieser Zahl verträglich, da das Korpus aus Kapitel 4 durch die Beschränkung auf morphologisch einfache Wörter weniger sehr lange Wörter enthält.

Abbildung A.1 zeigt die Unix-Commandos zur Extraktion der Informationen aus dem Volltext, in Abbildung A.2 ist das Histogramm für die Vokalbuchstabenansammlungen zu sehen. Bei der Extraktion sorgt die erste Zeile für ein Auslesen des Textes. In der zweiten Zeile werden alle Nichtbuchstaben zusammengefaßt und durch ein Linefeed ersetzt, Ausgabe ist also eine Liste der Wörter. Die dritte Zeile filtert Einzelbuchstaben und Wörter heraus, die an zweiter Stelle einen Großbuchstaben aufweisen, zum Beispiel Abkürzungen wie BRD. Im nächsten Filter werden die Vokalbuchstabenansammlungen zu einem @ zusammengefaßt. Im vorletzten Schritt werden alle anderen Zeichen durch einen Unterstrich ersetzt und danach entfernt.

<i>Extraktion:</i>
<pre>bin/unix/textonly data/eci1/ger03/*.eci \ tr -sc "a-zA-ZöäüÖÄÜß" "\012" \ grep '[a-zA-ZöäüÖÄÜ][a-zöäüß]' \ tr -s 'aeiouAEIOUöäüÖÄÜ' '@' \ tr -cs '@\012' '_' \ sed -e s/_//g > /syllablecount</pre>
<i>Durchschnittliche Anzahl der Vokalbuchstabenansammlungen (alle):</i>
<pre>wc syllablecount nawk '{ print (\$3-\$1)/\$2}'</pre>
<i>Durchschnittliche Anzahl der Vokalbuchstabenansammlungen (Wörter mit mindestens zwei Vokalbuchstabenansammlungen):</i>
<pre>grep @@ syllablecount wc nawk '{ print (\$3-\$1)/\$2}'</pre>

Abbildung A.1: Extraktion der Werte aus dem Volltext.

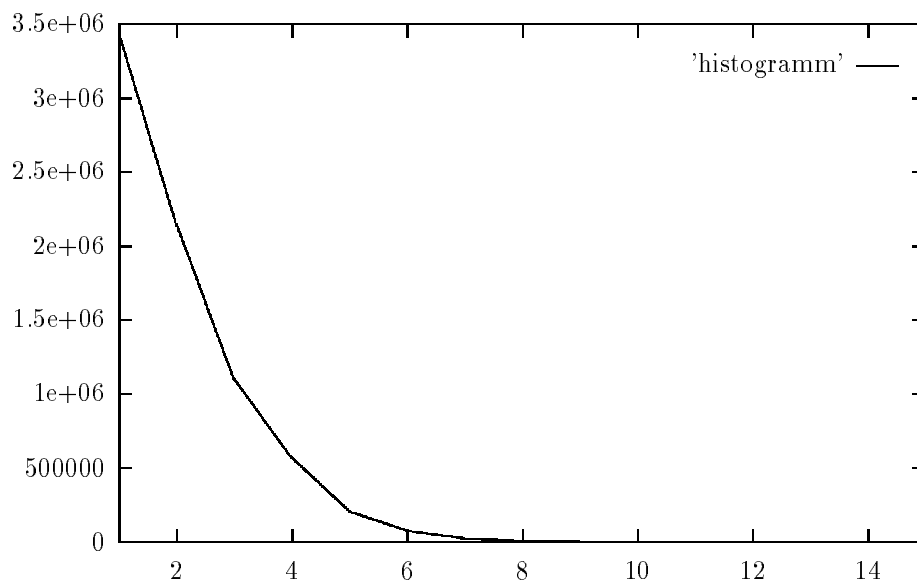


Abbildung A.2: absolute Häufigkeit über Anzahl der Vokalbuchstabenansammlungen

Anhang B

Literatur

- [Benw80] Benware, W. A. (1980) Zum Fremdwortakzent im Deutschen. *Zeitschrift für Dialektologie und Linguistik*, **47**, 289–312.
- [Bußm83] Bußmann, H. (1983) *Lexikon der Sprachwissenschaft*, Alfred Kröner Verlag, Stuttgart.
- [ChomHall68] Chomsky, N., Halle, M. (1986) *The sound pattern of English*, Harper and Row, New York.
- [ClemKeys83] Clements, G. N., Keyser, S. J. (1983) *CV Phonology. A Generative Theory of the Syllable*. MIT Press, Cambridge, Massachusetts etc.
- [Dael&al93] Daelemans, W., Gillis, S., Durieux, G., van den Bosch, A. (1993) Learnability and markedness in data-driven acquisition of stress. *Computational Phonology. Edinburgh Working Papers in Cognitive Science*, **8**, 157–178. (avail. by anonymous FTP from scott.cogsci.ed.ac.uk)
- [Dogil95] Dogil, G. (1995) Phonetic Correlates of Accent. In: van der Hulst, H. (ed.) *Word Prosodic Systems of European Languages*, DeGruyter, Berlin. (*in Ersch.*)
- [Eise91] Eisenberg, P. (1991) Syllabische Struktur und Wortakzent: Prinzipien der Prosodik deutscher Wörter. *Zeitschrift für Sprachwissenschaft*, **10**, 37–64.
- [Epst&al87] Epstein, G. et al. (1987) Approximation reasoning and Scott's information systems. *Proceedings, 2nd International Symposium on Methodologies for Intelligent Systems*, North Holland, Amsterdam.
- [Féry86] Féry, C. (1986) Metrische Phonologie und Wortakzent im Deutschen. *Studium Linguistik*, **20**, 16–43.

- [Fish87] Fisher, D. H. (1987) Knowledge Acquisition via Incremental Conceptual Clustering. *Machine Learning* **2**, 139–172.
- [Four&al89] A. Fourcin, G. Harland, W. Barry and V. Hazan (eds.) (1989) *Speech Input Output Assessment: Multilingual Methods and Standards*, Ellis Horwood Limited, Chichester, West Sussex, England.
- [Fudg84] Fudge, E. (1984) *English word-stress*, Allen & Unwin, London etc.
- [Gieg85] Giegerich, H. J. (1985) *Metrical Phonology and phonological structure. German and English*. Cambridge University Press, Cambridge etc.
- [Grew&al91] Grewendorf, G., Hamm, F., Sternefeld, W. (1991) *Sprachliches Wissen: eine Einführung in moderne Theorien der grammatischen Beschreibung*, 5. Auflage, Suhrkamp, Frankfurt am Main.
- [Grzy93] Grzymala-Busse, J. W. (1993) Selected Algorithms of Machine Learning from Examples. *Fundamenta Informaticae*, **18**, 193–207.
- [GrzySiko88] Grzymala-Busse, J. W., Sikora, D. J. (1988) LERS1—A System for Learning from Examples Based on Rough Sets. *Tech. Rep. TR-88-5*, University of Kansas.
- [Hall92] Hall, T. A. (1992) *Syllable structure and syllable-related processes in German*, Niemeyer, Tübingen.
- [Jess94] Jessen, M. (1994) *A survey of German word stress*, Manuskript, Universität Stuttgart.
- [JessMara95] Jessen, M., Marasek, K. et al. (1995) Acoustic correlates of word stress and the tense / lax opposition in the vowel system of German. *Proceedings International Conference of Phonetic Sciences 1995, Stockholm. (Eingereicht)*
- [Kras88] Krasowski, H. (1988) *Rough Set Based Method for Evaluation of Pilot Performance*, Ph.D. thesis, Technical University of Rzeszow.
- [Lieb65] Lieberman, P. (1965) On the acoustic basis of the perception of intonation by linguists. *Word*, **21**, 40–54.
- [Mang90] Mangold, M. (1990) *Duden Aussprachewörterbuch: Wörterbuch der deutschen Standardaussprache*, 3. Auflage, Bibliographisches Institut & F. A. Brockhaus AG, Mannheim.
- [Mich83] Michalski, R. S. (1983) A Theory and Methodology of Inductive Learning. *Artificial Intelligence*, **20**, 111–161.

- [Mroz89] Mrozek, A. (1989) Rough Set Dependency Analysis Among Attributes in Computer Implementation of Expert Inference Models. *Internat. J. Man-Mach. Stud.*, **30**, 457–473.
- [Nöth91] Nöth, E. (1991) *Prosodische Information in der automatischen Spracherkennung*, Niemeyer, Tübingen.
- [Ortm80] Ortmann, W. D. (1980) *Sprechsilben im Deutschen*, Manuskriptdruck, Goethe-Institut e. V. München.
- [Pawl82] Pawlak, Z. (1982) Rough sets. *International Journal of Information and Computer Sciences*, **11**, No. 5, 341–356.
- [Pawl91] Pawlak, Z. (1991) *Rough sets: Theoretical Aspects of Reasoning About Data*, Kluwer Academic, Dordrecht, The Netherlands.
- [Pawl&al86] Pawlak, Z., Słowski, K., Słowski, R. (1986) Rough Classification of patients after highly selective vagotomy for duodenal ulcer, *Internat. J. Man-Mach. Stud.*, **24**, 413–433.
- [Quin86] Quinlan, J. R. (1986) Induction of decision trees. *Machine Learning*, **1**, 81–106.
- [Rabi89] Rabiner, L. R. (1989) A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proc. of the IEEE*, Vol. **77**, No. 2, 257–285.
- [Rame92] Ramers, K. H. (1992) Ambisyllabische Konsonanten im Deutschen. In: Eisenberg, P., Ramers, K. H. & Vater, H. *Silbenphonologie des Deutschen*, Narr, Tübingen, 246–283.
- [Venn90] Vennemann, T. (1992) Syllable structure and syllable cut prosodies in Modern Standard German. In: Bertinetto, P. M., Kentstowicz, M. & Loporcaro, M. (eds.) *Certamen Phonologicum II. Papers from the 1990 Cortona Phonology Meeting*, Rosenberg & Sellier, Turin 211–243.
- [Venn91] Vennemann, T. (1991) Skizze der deutschen Wortprosodie. *Zeitschrift für Sprachwissenschaft*, **10**, 86–111.
- [Venn92] Vennemann, T. (1992) Syllable structure and simplex accent in Modern Standard German. In: Ziolkowski, M., Noske, M. & Deaton, K. (eds.) *Papers from the 26th Regional Meeting of the Chicago Linguistic Society. Volume 2. The parasession on the syllable in phonetics & phonology*, 399–412.
- [Wong&al86] Wong, S. K. M., Ziarko, W., Li Ye, R. (1986). Comparison of rough-set and statistical methods in inductive learning. *International Journal of Man-Machine Studies*, **25**, 53–72.

- [Ziar89a] Ziarko, W. (1989). A Technique for Discovering and Analysis of Cause-Effect Relationships in Empirical Data. *International Joint Conference on Artificial Intelligence, Proceedings, Workshop on Knowledge Discovery in Databases, Detroit*.
- [Ziar89b] Ziarko, W. (1989). Data Analysis and Case-Based Expert Systems Development Tool ROUGH. *Proceedings, Workshop on Case-Based Reasoning, Pensacola Beach, Florida*, Morgan Kaufmann, Los Altos, CA.
- [Ziar93] Ziarko, W. (1993). Variable Precision Rough Set Model. *Journal of Computer and System Sciences*, **46**, 39-59.
- [Ziar&al93] Ziarko, W., Edwards, R. G. D. (1993) An application of the rough sets approach to discovery of strong predictive rules in stock market data. Available via anonymous FTP from ftp.cs.uregina.ca (/pub/ebrsc).