TR-IT-0282

CHATR: Modeling Prosody with MARS and Text-to-Speech for German

Marcel Riedi

October 1998

ABSTRACT

This technical report describes new models of segmental duration and fundamental frequency, and different improvements and additions needed for German text-to-speech with CHATR. The duration and frequency models were realized with "multivariate adaptive regression splines", a nonparametric regression method very well suited for problems having mixed ordinal and categorical input factors. Models for German and English have been developed. The text-to-speech additions and improvements include modules for generating ToBI for German text input and the handling of words not contained in the lexicon. "Decision trees" were used for these tasks.

Contents

Chapter 1

Introduction

After finishing my PhD thesis on the subject of "controlling segmental duration in speech synthesis systems" at the ETH Zürich, Switzerland, I spent four months (July – October 1998) at department 2 of ATR ITL.

My main concerns during this time were the prediction of duration and fundamental frequency for German and English, and the German-specific parts of the CHATR system. A description of the CHATR system can be found in [ATR97]. The work done on the German-specific parts of the system before my stay at ATR are described in [Bri97] and [Str97].

The part of this work concerned with the prediction of duration and fundamental frequency is strongly related to my previous work done at the ETH. Besides continuing my previous work I was also interested in extending CHATR to a "complete" text-to-speech system for German. This required works on many parts of the system. Many problems could be viewed as symbol mapping problems. They were handled with decision trees. This "simple" approach was selected because of its flexibility (and also because of the limited time available). In this report the work carried out in this context will be described, which includes:

- German speech databases and lexicon
- Prediction of segmental duration and fundamental frequency contours for German and English
- Generation of boundary tones, phrase accents, and pitch accents (the tone tier of ToBI) for text input
- Strategy for handling words not contained in the lexicon
- Text normalization (partially)

1.1 Outset

For testing purposes, the code implemented in this work was included in CHATR version 0.94 (July 1, 1998). The work done on the German-specific parts of the system until then are described in [Bri97] and [Str97]. Only the work described in [Bri97] was included in CHATR 0.94, the other parts were missing. In particular, the ToBI generation code (see [Str97]) was missing. After including the missing parts, several German texts were synthesized.

The realization of duration and fundamental frequency for German clearly was of insufficient quality. This was not surprising since no duration model for German had been included in the system. Instead, the fall-back strategy (using average phoneme durations observed in the speech database) was used. Fundamental frequency was predicted with a modified version of the Anderson, Pierrehumbert, and Liebermann technique. For English, on the other hand, duration and fundamental frequency models were realized with linear regression models.

The manually setup rule system used to generate the tones and breaks indices (ToBI) worked reasonably well, but as discussed above, were not correctly realized by the prosody realization models.

The pronunciation (and part of speech) of the words contained in the input text were looked up in a wordform lexicon. If a word could not be found in the lexicon, synthesis of the whole text would fail. Furthermore, the system did not do any preprocessing on German input text.

1.2 Improving the System

To improve the speech signal quality of the system, a new CHATR speech database for German was created. This database contained a larger amount of text than the previously existing ones.

In a next step, multivariate adaptive regression splines models of segmental duration and fundamental frequency were developed for German and English. The input information required by these models includes ToBI. Based on the same data that has been used to train the German prosody models, a decision tree was constructed which generates ToBI for German text input. By using the same data problems related to the interfacing of ToBI generation and prediction of duration and fundamental frequency could be avoided.

To allow the handling of any text input a few text preprocessing functions (text normalization) for German have been added to the system. Only some of the required steps of text normalization have been implemented.

In particular, abbreviations and acronyms currently still are not correctly handled.

1.3 Statistical Modeling

The statistical methods applied in this work are multivariate adaptive regression splines" (MARS) and "decision trees". An overview of statistical modeling and MARS in the context of duration modeling is given in [Rie97] and [Rie98]. See [Fri91] for a detailed description of MARS. Decision trees are described, e. g., in [IND92] and [Qui93].

Chapter 2

Data

There was not enough time to create new databases, specifically adapted to the problems at hand. Two databases created at other laboratories, Siemens Germany and University of Stuttgart, were therefore used in this work. They will be described in the following sections.

2.1 Siemens Germany

The Siemens data was spoken by a professional male speaker. It consists of read news. A total of 954 sentences, which corresponds to approximately 3 hours of speech, are in the database. The speech was automatically segmented (phones) with an HMM-based segmentation tool. For each sentence the following information is available:

- Recorded speech in PhonDat format
- Output produced by the segmentation tool (HTK format)
- Orthographic text

Because of the large amount of data and the suitable speaker characteristics, this data was used to make a new CHATR speech database.

2.2 University Stuttgart

The data recorded at the University of Stuttgart was also spoken by a professional male speaker. It consists of read news broadcasted on "Deutschlandfunk". A total of 359 sentences (approximately 48 minutes of speech, grouped in 72 texts) are in the database. The speech was automatically segmented with an HMM-based tool. Tones and breaks indices (ToBI) were manually labeled. A detailed description of the data and further references can be found in [Moe98]. For each of the 72 texts the following information is available:

- Recorded speech in ESPS format
- Segment (phone) boundaries in ESPS/xlabel format
- Syllable boundaries and lexical stress in ESPS/xlabel format
- Word boundaries in ESPS/xlabel format
- Tones tier of ToBI in ESPS/xlabel format
- Breaks tier of ToBI in ESPS/xlabel format
- Part of speech tags in ESPS/xlabel format
- Orthographic text

This was the only German database containing ToBI labeling available. It was mainly used to train the prosody models and the decision trees used to generate ToBI for text input.

2.3 CHATR Speech Databases

Previously two CHATR speech databases existed for German, both made with data from PhonDat 1 and 2 ("kko": make speaker, about 45 min and "rtd": female speaker, about 42 min).

One way to improve the quality of the synthesized speech is to increase the size of the speech database. Especially the Siemens data described above contains a larger amount of speech. Both, the Siemens and the University of Stuttgart data, were used to make CHATR speech databases. The Siemens data was spoken by a speaker named "Aichinger". This is way this speech database was named "aich". The University Stuttgart data was recorded from news spoken on "Deutschlandfunk" (name of the speech database is "dlf").

The instructions for making such a database given in [ATR97] were followed. Because of the size of the Siemens data the weight training functions had to be slightly modified. Instead of using all data available, only the first thousand of each phoneme class were used to calculate the weights. Using all data did not work because the memory requirements were too high to run it on any of the available machines.

The automatic segmentation and labeling carried out for both sets of data used very similar sets of phonemes, which did not fit any of the German phoneme sets already available in CHATR ("sampaG" and "german"). Therefore a new phoneme set, "sampaGd", was defined (see Appendix C). This new phoneme set was used for the "aich" and the "dlf" databases. This also required a new lexicon, "sampaGd.dic", which was derived from the "sampaG" lexicon. The differences between "sampaG" and "sampaGd" are listed in the following:

- "sampaGd" contains no special symbol ('+' in "sampaG") for vowels in lexically stressed syllables.

- There are no glottal closures in "sampaGd" (because they are not labeled in the Siemens and Stuttgart data). They are included in the following vowel.

- All nasalized vowels were replaced by two-letter combinations in "sampaGd" (no nasalized vowels in "aich" and "dlf").

- All occurrences of the phoneme '6' were replaced by the two phonemes '@ r' in "sampaGd".

- All occurrences of '6' in diphthongs were replaced by a separate phoneme 'r' in "sampaGd".

Chapter 3

Modeling of Prosodic Parameters

3.1 Introduction

The aim of modeling duration and fundamental frequency in this work is to produce models that have a high prediction accuracy, i. e. they should predict durations and frequency values similar to the ones observed in natural speech. The information available as input to the models is some phonological representation of the text to be uttered. It contains the sequence of phonemes, syllable and word boundaries, information about the grouping of words into phrases and the distribution of accents (cf. Chapter 4), and other information that can derived from the input text.

Speech pause durations are not predicted with the duration model described in this chapter. They are assigned some constant duration depending on the type of pause (cf. Chapter 4).

The model of (segmental) duration described in the following predicts the duration of phones. The fundamental frequency model predicts one, two, or three frequency values per syllable, depending on the consonants in the onset and coda. The input to the models (so-called input factors or features[1]) are all independent of the specific phoneme set being used. To achieve this, the information related to the place and type of articulation, frontness, height, etc. is used. This allows the implemented code to be used for other languages (after possibly implementing some new factors required by a language). Models for German and English have been trained in this work.

[1] The terms factor and feature will be used interchangeably.

3.2 Data

The tools available in CHATR were used to prepare the train and test data. After implementing the selected feature functions and doing some modifications to the raw Stuttgart data (mapping GToBI to ToBI, modifying lexical stress information, etc.), PhonoForm utterances were created for the "dlf" data as described in [ATR97]. With these PhonoForm utterances (72 texts with 359 sentences) the train and test data could easily be generated.

For the English data the PhonoForm utterances were already available[2].

3.3 Duration

The model of segmental duration implemented in this work is very similar to the one described in [Rie98]. The only differences are the data used to train and test the model and the selected set of input factors. The data has been described above and the selected set of input factors and the results will be described in the following.

3.3.1 Input Factors

The following factors have been selected (shown together with their abbreviation used in the remainder of this report, and in the implemented code):

a Vowels: combination of length and frontness, consonants: type of articulation

b Vowels: height, consonants: place of articulation

app ap an ann As 'a', for the two preceding and following segments

bpp bp bn bnn As 'b', for the two preceding and following segments

rv Vowels: roundness, consonants: voiced or unvoiced

sp Position of segment in syllable (counting segments)

sn Number of segments in syllable (in number of segments)

ac Accentuation of syllable

ls Lexical stress of syllable

[2] These PhonoForms were used after correcting an error in one of the utterances which had a syllable containing only a consonant.

wp Position of syllable in word

fo Position in foot (and type of foot)

po Position of foot in sentence

pp Position of syllable in sentence

In the following tables for each factor all of its possible values and the coding used in the MARS model[3] are shown. Most of the factors have also been used in the work described in [Rie98].

Factor a

vowels

length	short/schwa			long/geminate			diphthong		
frontness	1	2	3	1	2	3	1	2	3
MARS code	1	2	3	4	5	6	7	8	9

consonants

type	stop	fric.	affric.	nasal	liquid	closure	
MARS code	10	11	12	13	14	15	

For the two previous and the two following segments additionally a value for speech pauses (MARS code 16) occurring in the context is added. A description of vowel frontness and length, and consonant articulation type can be found in the phoneme set definition description in [ATR97].

Factor b

height (vowels)	1	2	3						
place (cons.)				labial	alv.	palat.	labio-d.	dental	velar
MARS code	1	2	3	4	5	6	7	8	9

For the two previous and the two following segments additionally a value for speech pauses (MARS code 10) occurring in the context is added. See [Rie98] for more information on this factor. See the phoneme set definition description in [ATR97] for more information about vowel height and consonant articulation place.

[3] The MARS software requires the values of categorical factors to be represented as integers (only represented, order does not matter!).

Factor rv

rounded (vowel)	no	yes		
voiced (consonant)			no	yes
MARS code	1	2	3	4

Factor sp

position	1	2	3	4	5	6 or later
MARS code	1	2	3	4	5	6

Factor sn

number	1	2	3	4	5	6 or more
MARS code	1	2	3	4	5	6

Factor ac

H/L pitch accent	yes	no
MARS code	1	2

Factor ls

lexical stress	yes	no
MARS code	1	2

Factor wp

position	initial	medial	final	single
MARS code	1	2	3	4

Factor fo

	MARS code
stressed singleton	1
stressed, one following	2
stressed, two following	3
stressed, three or more following	4
unstressed singleton	5
unstressed pair	6
unstressed three or more	7
unstressed (anacrusis)	8

See [Cam93] for more information.

Factor po

	MARS code
IP initial foot	1
IP final foot	2
ip initial foot	3
ip final foot	4
medial foot	5
sentence with one foot	6

'IP' denotes an "intonation phrase" and 'ip' an "intermediate phrase".

Factor pp

	MARS code
IP initial syllable	1
IP final syllable	2
ip initial syllable	3
ip final syllable	4
medial syllable	5

3.3.2 Output Coding

Because the mean squared error is being minimized in the MARS training algorithm the duration to the power of 0.25 has been modeled. Another advantage is that for shorter durations the same prediction error will have a smaller impact on the decoded duration than for longer durations. A more detailed discussion comparing the different types of output coding can be found in [Rie98].

3.3.3 MARS Construction and Results

The MARS training algorithm has been used with its default values (MARS v3.6). In particular, the "speed" parameter has been set to 4. The maximum number of basis (M_{max}) functions added to the model was varied and the maximum number of factors interacting with each other (K_{max}) was set to 3

About 75% of the total of 35156 samples available in the prepared data (one sample consists of a vector of factor values with the resulting duration observed in the "dlf" data) were used to train ($M_{max} = 600, K_{max} = 3$) the duration model. The resulting correlation coefficient calculated on the test set was 0.75.

method	MARS	lr	average
r	0.80	0.76	0.43

Table 3.1: Comparison of the correlation coefficients r of the MARS duration model for English, the linear regression model, and the average model. All data available in "f2b" (English) was used to calculate r.

The prediction accuracy of the MARS model in terms of the correlation coefficient was compared to the simple model which uses the average phoneme duration observed in the "dlf" data as prediction. Using all data in "dlf" this simple average model achieved a correlation coefficient of 0.37, whereas the MARS model achieved a correlation of 0.76.

Using 75% of the English "f2b" data (total of 38360 samples) a MARS model was constructed ($M_{max} = 600, K_{max} = 3$) with a test set correlation coefficient of 0.77. A comparison of the correlation coefficient r of this model and of the previously available linear regression model and the simple average model is shown in Table 3.1. The MARS model performed somewhat better than the linear regression model and these two models had a clearly higher prediction accuracy than the average model.

3.4 Fundamental Frequency

Fundamental frequency (F_0) has also been modeled with MARS. For each syllable at least one, and at most three frequency values are predicted. For the nucleus the model always predicts a value, for the onset and the coda only if the first, respectively last, consonant is voiced.

The data used to train the model has been described above. The selected set of input factors and the results will be described in the following.

3.4.1 Input Factors

In [Tra95] a neural network was used to model fundamental frequency. This network used feedbacks of the outputs of some of its hidden neurons. This allowed the declination behavior to be modeled. Because of stability problems, the output of a MARS model cannot be directly used as feedback. To avoid the feedback requirement and still be able to model declination, factors related to the position within intermediate and intonation phrases have been included in the model. A large set of factors, related to different levels of speech have been selected (based on the factors described in [Tra95] and

[Moe98]):

vl Length of nucleus

vh Height of nucleus

cl Type of last consonant of onset

cr Type of first consonant of coda

vlp, vhp, clp, crp, vln, vhn, cln, crn As above for the previous and the following syllable

lip Length of intermediate phrase in number of syllables (ordinal)

lasip Last syllable of intermediate phrase

poip Normalized position in intermediate phrase (position / length, ordinal)

ls Lexical stress

at Type of pitch accent

ds Downstep

bt Type of "phrase accent"/"boundary tone"

lspp, atpp, dspp, btpp, lsp, . . . , dsnn, btnn As above for the two preceding and following syllables

lIP Length of intonation phrase in number of syllables (ordinal)

lasIP Last syllable of intonation phrase

poIP Normalized position in intonation phrase (position / length, ordinal)

lat Pitch accent type of previous accentuated syllable

latd Distance in number of syllables to previous accentuated syllable (ordinal)

rat Pitch accent type of following accentuated syllable

ratd Distance in number of syllables to following accentuated syllable (ordinal)

lbt "Phrase accent"/"boundary tone" type of previous accentuated syllable

lbtd Distance in number of syllables to previous "phrase accent"/"boundary tone" (ordinal)

rbt "Phrase accent"/"boundary tone" type of following accentuated syllable

rbtd Distance in number of syllables to following "phrase accent"/"boundary tone" (ordinal)

sp Position in syllable

For each factor all of its possible values and the coding used in the MARS F_0 model[4] are shown in the following.

Factors vl, vh, cl, cr

For the factors "vl", "vh", "cl", and "cr" the information available in the phoneme set definition is used (see [ATR97] and Appendix C).

vl		
nucleus is long vowel or diphthong	true	false
MARS code	1	2

vh		
high or 'mid and front' nucleus	true	false
MARS code	1	2

cl		
'unvoiced or plosive'	true	false
MARS code	1	2

cr		
'unvoiced or plosive'	true	false
MARS code	1	2

For the preceding and following syllable ("vlp", "vhp", etc.) an additional coded value ('3') is used for contexts before the first and after the last syllable of a phrase.

[4] As mentioned before, the MARS software requires the values of categorical factors to be represented as integers (order does not matter).

Factors lip, lasip, poip

"lip" is an ordinal factor. Its values are the number of syllables in the intermediate phrase $([1, 2, 3, \ldots])$.

Factor "lasip" has two values, it is true (coded as '1') for the last syllable of an intermediate phrase and false ('2') otherwise.

Factor "poip" is the normalized position of the syllable in the intermediate phrase. It is calculated by dividing the position of the syllable (counting syllables) in the intermediate phrase by the length of the intermediate phrase.

Factors ls, at, ds, bt

ls		
lexical stress	true	false
MARS code	1	2

at			
pitch accent type	H	L	none
MARS code	1	2	3

ds		
downstep	true	false
MARS code	1	2

bt								
boundary type	H–	L–	H–H%	L–H%	H–L%	L–L%	%H	none
MARS code	1	2	3	4	5	6	7	8

For the two preceding and following syllable ("lspp", "atpp", etc.) additional coded values ('3', '4', '3', and '9') are used for contexts before the first and after the last syllable of a phrase.

Factors lIP, lasIP, poIP

"lIP" is an ordinal factor and its values are the number of syllables in the intonation phrase $([1, 2, 3, \ldots])$.

Factor "lasIP" is true (coded as '1') for the last syllable of an intonation phrase and false ('2') otherwise.

Factor "poIP" is the normalized position of the syllable in the intonation phrase (division of the position of the syllable in the intonation phrase by its length).

Factors lat, latd, rat, ratd, lbt, lbtd, rbt, rbtd

lat			
previous pitch accent type	H	L	none
MARS code	1	2	3

“latd” is the distance to the previous accentuated syllable in number of syllables (ordinal factor).

rat			
following pitch accent type	H	L	none
MARS code	1	2	3

“ratd” is the distance to the following accentuated syllable in number of syllables (ordinal factor).

lbt				
previous boundary type	H–	L–	%H	IP boundary
MARS code	1	2	3	4

“lbtd” is the distance to the previous “phrase accent”/”boundary tone” in number of syllables (ordinal factor).

rbt						
following boundary type	H–	L–	H–H%	L–H%	H–L%	L–L%
MARS code	1	2	3	4	5	6

“rbtd” is the distance to the following “phrase accent”/”boundary tone” in number of syllables (ordinal factor).

Factor sp

sp			
position in syllable	onset	nucleus	coda
MARS code	1	2	3

3.4.2 Output Coding

In the F_0 models described in [Tra95] and [BH96], the fundamental frequency values have been directly modeled. In this work, the z-score of fundamental frequency has been modeled. In [BH96] 3 frequency values were predicted for each syllable, one for the start, one for the center of the nucleus, and one for the end of the syllable. Three different models were used for these 3

r	MARS	lr
all data	0.68	0.60

Table 3.2: Comparison of the correlation coefficients r of the MARS model and the linear regression model. All data available in "f2b" (English) was used to calculate r.

values, independent of the type of phone at the start and end of the syllable (no phonetic properties were used in the input factors of these models).

A different method was used in this work. F_0 is only predicted for the start and end of the syllable if the first consonant of the onset, respectively the last consonant of the coda, is voiced and not a plosive[5]. Phonetic properties are included in the set of input factors ("vl", "vh", "cl", etc.) and a single model is used to predict all frequency values. The input factor "sp" indicates whether the predicted value is related to the onset, nucleus, or coda of the syllable. This way, instead of treating the modeling of the different F_0 values independently from each other, all the data available is used to train a single model, thereby taking advantage of the dependencies.

3.4.3 MARS Construction and Results

The MARS training algorithm has been used with its default values (MARS v3.6). The "speed" parameter was set to 4. Different maximum numbers of basis functions have been tried.

About 75% of the total of 24725 samples available in the prepared data (one sample consists of a vector of factor values with the resulting F_0 observed in the "dlf" data) were used to train ($M_{max} = 350, K_{max} = 3$) the fundamental frequency model . The resulting correlation coefficient calculated on the test set was 0.58.

Using 75% of the English "f2b" data (total of 26932 samples) a MARS model was constructed ($M_{max} = 350, K_{max} = 3$) with a test set correlation coefficient of 0.61. A comparison of the correlation coefficient r of this model and of the previously available linear regression model is shown in Table 3.2. The MARS model performed somewhat better than the linear regression model.

[5] A setup with three different models has also been tried with MARS, but the prediction accuracy was inferior to the single-model approach.

3.5 Conclusions

In the studies described in [Rie98] manually segmented data was used to train and test MARS-based duration models. In this work automatically segmented data was used.

MARS was also used to model fundamental frequency in a successful way. Instead of using feedbacks to account for the dependency of a frequency value on the previously predicted values, the input factors have been extended.

Whereas the dependencies of the MARS algorithm on the different learning algorithm parameters were investigated for the problem of modeling duration (cf. [Rie98]), such studies were not yet done for F_0 modeling. Varying parameters might lead to improvements.

A direct modeling of F_0 has been applied here. Coding the output in a way that the prediction error has a normal-like distribution might further improve the prediction accuracy.

Only a relatively small amount of data has been used in this work. In particular, the data did not cover questions and spontaneous speech. It will be interesting to see how well this type of modeling will work with such larger amounts of data containing more prosodic variations.

Chapter 4

Generation of ToBI for Text Input

4.1 Introduction

The naturalness of synthetic speech depends on the correct grouping of words into phrases and the distribution of accents. This information is contained in the tone tier of ToBI (cf. [BH94]). The prosody models described in Chapter 3 use ToBI to calculate some of the input factors.

In many TTS systems currently only syntactical information is used to determine phrasing and accentuation (semantics would also be required). The aim is then to generate phrases and accents for a text uttered in a neutral information transmission context. E. g., emphatic and contrastive stresses cannot be treated correctly this way.

The approach used in this work directly tries to generate ToBI given only the part of speech sequence and the punctuation of the text. Sentences are not syntactically analyzed, instead, decision trees are trained to map the part of speech sequence and punctuation to ToBI.

One decision tree, which will be henceforth called the *tone tree*, is used to predict phrase accents and boundary tones (see Figure 4.1). And another tree, here called the *accent tree*, predicts pitch accents (see Figure 4.2). Both trees use the same set of input factors, consisting of the part of speech of the preceding and following words, the part of speech of the current word and the punctuation following the current word.

These trees generate accents and boundaries for all words in the text. Phrase accents and tone boundaries are always positioned at the end of a word. But pitch accents are related to syllables. The algorithm used to assign the pitch accents to syllables (using lexical stress) is described in [Str97].

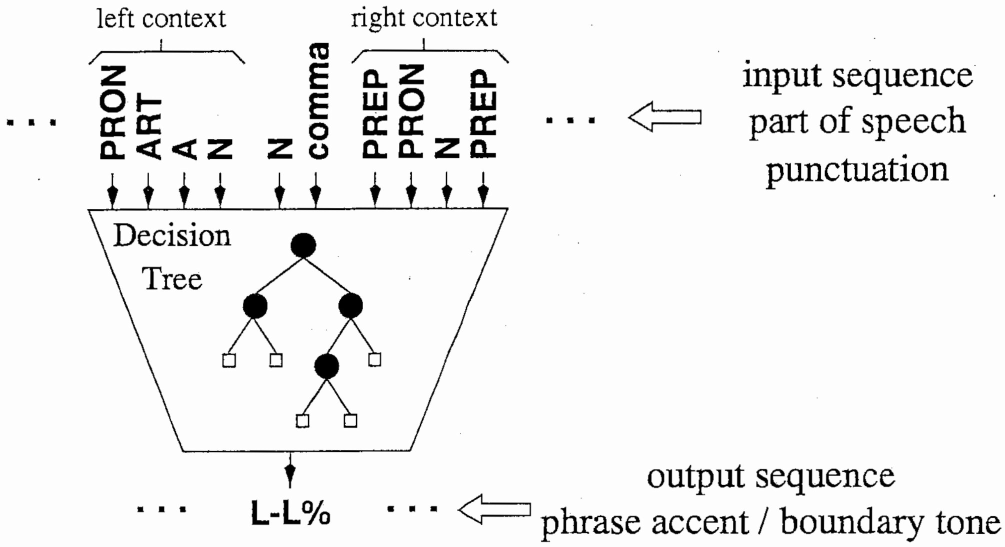

Figure 4.1: Generating "phrase accents" and "boundary tones" with the tone tree (context size 4).

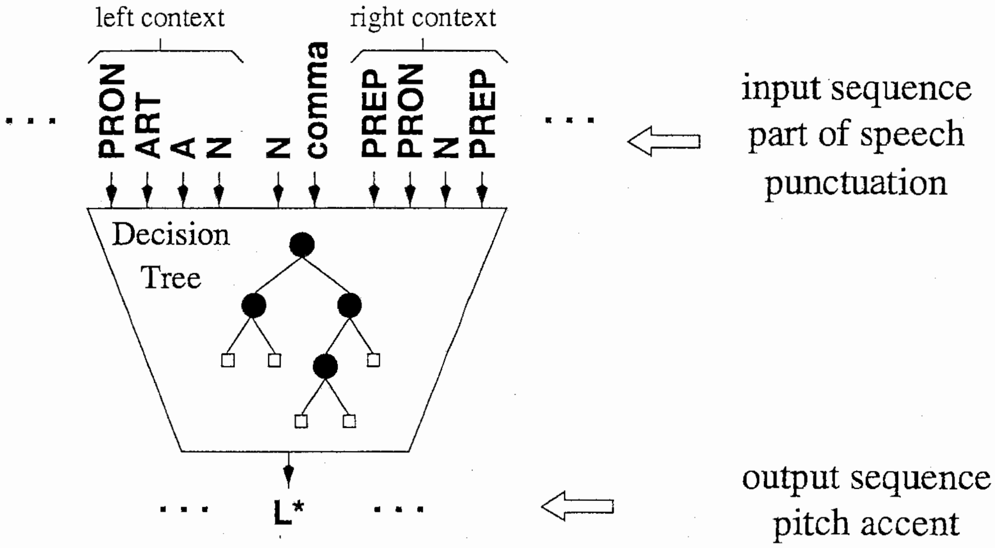

Figure 4.2: Generating "pitch accents" with the accent tree (context size 4).

4.2 Data

The data from the University of Stuttgart was used to train the tone and accent trees. The phrase and accent information therein was labeled with the Stuttgart labeling system (see [May95]). These labels were mapped to GToBI, which is an extension of ToBI (see [Rey96]). With these mapped labels the required train data could easily be generated.

The possible part of speech categories of the inputs are those looked up either in the "sampaGd" lexicon or those predicted by the part of speech tree described in Chapter 5:

A	adjectives	**NUM**	numerals
ADV	adverbs	**PREP**	prepositions
ART	articles	**PRON**	pronouns
C	conjunctions	**V**	verbs
N	nouns		

An additional value, "nopos", is used for the context factors at the start and end of sentences. The punctuation input factor either has the value "comma", "dot", or "nopunct". These are the only punctuation marks occurring in the Siemens data. When using the tone and accent trees for ToBI generation, question marks, colons, etc., are mapped to appropriate values.

The possible outputs of the tone tree currently included in CHATR are

H–
L–

H–H%
H–L%
L–H%
L–L%

%H
notone

Trees have been constructed with data containing the downstepped variants of phrase accents and boundary tones ('!H–', '!H–H%', '!H–L%'). But because of their low number of occurrences in the data, these downstepped variants were not selected during tree construction. Some trees were trained with data containing the downstep information and for other trees this information was removed from the data.

The possible outputs of the accent tree currently included in CHATR are:

H*   L*   noaccent

Again, downstepped variants ('!H*', 'L*+!H', 'H +!H*') and also 'L*+H' were not selected during tree construction.

With the Stuttgart data a total of 5727 input/output pairs as needed to construct the tone and accent trees could be produced.

context	prediction accuracy	size
0	83.9	4
1	84.4	10
2	84.3	22
3	85.4	62
4	85.4	34
5	85.2	32
7	85.0	60

Table 4.1: Prediction accuracy for the test data and the size (in number of nodes and leafs) of the pruned tone trees for different context sizes. The data containing the downstepped variants has been used to construct these trees.

4.3 Tree Construction

75% of the data was used to construct the tone and the accent tree. This resulted in 4295 input/output pairs available for tree construction and 1432 pairs for testing.

The software package IND was used to construct the tone and accent decision trees. The following options (see [IND92] for further details) were used:

```
mktree -e -s cart -v -v phrase
mktree -e -s cart -v -v accent
```

4.4 Results

4.4.1 Phrase Accents and Tone Boundaries

Trees with different context sizes (number of words to the left and right of the current word used in the input) have been constructed. The resulting prediction accuracies (calculated with the test set) and the resulting tree sizes are shown in Table 4.1. The data including the downstep information has been used to construct these trees.

Within the 1432 input/output pairs of the test data, 1101 had the output 'notone'. By simply predicting 'notone' for all input data a prediction accuracy of 76.9% could be achieved.

Including the distances of the current word to the previous intermediate and intonation phrase boundaries did not improve the prediction accuracy of the resulting tree.

context	prediction accuracy	size
0	64.2	4
1	68.8	44
2	70.0	18
3	70.3	22
4	70.7	16
5	70.7	18
7	70.1	42

Table 4.2: Prediction accuracy for the test data and the size (in number of nodes and leafs) of the pruned accent trees for different context sizes. The data containing the downstepped variants has been used to construct these trees.

A prediction accuracy of 84.7% resulted when trained with the data not containing the downstep information and a context size of 4. In the train data there are some cases with sentence ends not labeled by a boundary tone. The result was that the tone tree not always generated a boundary tone for sentence ends. This lead to unnatural sounding predictions of the prosody modules, and, combined with the pause prediction method used in CHATR, also to speech pauses with unnatural durations. The tone tree was therefore manually modified to make sure that at a sentence end a boundary tone would always be generated. This modified tree was included in CHATR.

4.4.2 Pitch Accents

Trees with different context sizes (number of words to the left and right of the current word used in the input) have been constructed. The resulting prediction accuracies and the resulting tree sizes are shown in Table 4.2. The data including the downstep information has been used to construct these trees.

847 of a total of 1432 input/output pairs of the test data, had the output 'noaccent'. By simply predicting 'noaccent' for all input data a prediction accuracy of 59.1% could be achieved.

As for the tone tree, including the distances of the current word to the previous intermediate and intonation phrase boundaries did not improve the prediction accuracy of the resulting tree.

When using the data not containing the downstep information, the prediction accuracy was 72.4% for a context size of 4. This tree was included in CHATR.

4.5 Conclusions

One problem observed for the tone tree are longer sequences of words not containing any punctuation. In such cases 'notone' can be predicted for a long sequence of words. This will result in poorly predicted fundamental frequency values (since such cases did not occur often enough in the F_0 train data).

Either the data available for training needs to be increased or some strategy taking the distances to the previous intermediate and intonation phrase boundaries into account must be used.

Problems with the accent tree are less obvious. Further experimentation is required.

Chapter 5

Unknown Words

5.1 Introduction

Often in TTS systems the following word features are looked up in some kind of lexicon:

- pronunciation
- syllable boundaries
- distribution of lexical stress
- part of speech (respectively word category)

In CHATR (at least for English and German) currently a lexicon containing orthographic wordforms[1] together with this information is used.

In a TTS system intended for use with unrestricted text input there must be some kind of strategy to handle *unknown words*, i. e., wordforms not contained in the lexicon. Such words will always exist, since the set of words used in a language changes. The methods used to handle this problem are strongly language dependent.

For English, the manually setup letter to sound rules developed at the US Naval Research Laboratory, Washington DC, are used in CHATR to determine the pronunciation of unknown words. Syllable boundaries are set in a way that each syllable has one vowel and the syllables are cut at minimum sonority (see "`lex_syllabify`" in "`lexicon.c`"). The strategy adopted for the assignment of lexical stress to unknown words is to always mark the first

[1] Another strategy is to use a morpheme lexicon, which allows the same amount of words to be covered with a much smaller lexicon.

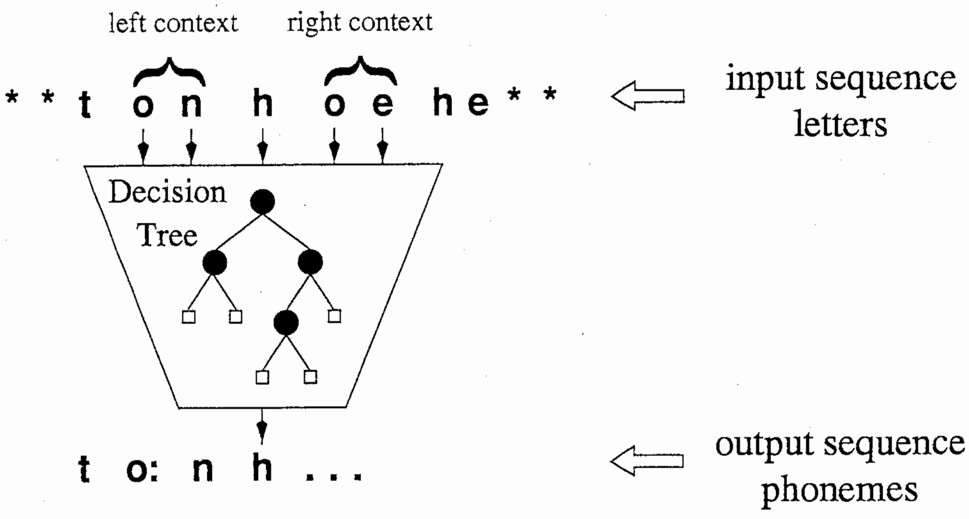

Figure 5.1: Letter to sound tree (context size 2).

syllable as lexically stressed[2]. Additionally, if the word has more than three syllables, the syllable before the last one in a word is marked as being lexically stressed. For English seven word classes are differentiated (determiner, preposition, pronoun, part of the verb "to be", conjunction, some other function word, and content word). All unknown words are assigned the word class "content word".

For German no such strategy to handle unknown words existed in CHATR. Synthesis would fail if a wordform could not be found in the lexicon. The following describes the realized strategy to handle unknown German words.

The pronunciation of an unknown word is determined with a decision tree. With this tree for each letter zero, one, or two phones are predicted (in the following this tree will be denoted as the *letter to sound tree*). As shown in Figure 5.1[3], the input information used by the decision tree is the current letter and preceding and following letters. No information regarding morphology, syntax, or semantics is used. This information is sometimes necessary to determine the correct pronunciation of a word (e.g. homographs). In some of these cases the letter to sound tree will fail, i.e., it will predict inaccurate pronunciations. Another area where the tree might fail is the inclusion of words from non-German languages in German sentences. Often the pronunciations of these words follow completely different rules. Because of this, it does not make much sense to use the same decision tree to predict

[2] In CHATR two levels of lexical stress are used: stressed and unstressed.

[3] In this report phonetic transcriptions are shown in SAMPA notation.

	highest N		2. highest N	
#s	pattern	N	pattern	N
2	su	20677	us	2511
3	suu	39215	usu	10346
4	suuu	41376	usuu	12338
5	suuuu	21576	uusuu	6727
6	suuuuu	5901	usuuuu	2243
7	suuuuuu	1241	usuuuuu	655
8	suuuuuuu	264	uuuusuuu	165

Table 5.1: Number of occurrences 'N' of lexical stress patterns in the "sampaGd" lexicon for words with '#s' syllables. The two most often occurring patterns are shown for each '#s'. The pattern 'suu', e. g., stands for a word with 3 syllables where the first syllable is stressed and the other syllables are unstressed.

the pronunciation of this kind of words. There are similar problems regarding the pronunciation of names. These problematic cases should be handled with lexicon entries. The letter to sound tree is only meant as fall-back solution for the words missing in the lexicon and will work reasonably well with words having a regular German pronunciation. The letter to sound tree will be described in more detail in Section 5.2.

To determine the syllable boundaries and the distribution of lexical stress of unknown words the same methods as for English are used. These heuristic solutions do not take morphology into account. They work in many cases, but might fail in those cases where syllable boundaries, respectively lexical stresses depend on morphology.

The results of an analysis of the distribution of lexical patterns in the "sampaGd" CHATR lexicon (which is derived from the Celex database) is shown in Table 5.1. These results indicate that a better heuristic for the lexical stress distribution of German words would be to have the first syllable stressed and the others unstressed.

To determine the word category of an unknown word another decision tree has been constructed. This tree, which in the following will be denoted as the *part of speech tree*, uses as input the word categories of the preceding and following words (see Figure 5.2).

So-called *closed word categories* contain a fixed, relatively small number of words (articles, pronouns, prepositions, and conjunctions). These categories should be fully contained in the lexicon. This simplifies to a certain degree the problem of determining the part of speech of unknown words since only

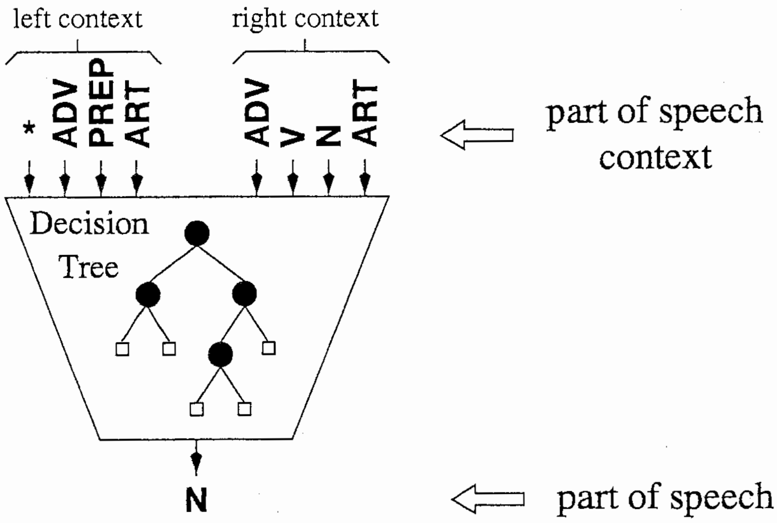

Figure 5.2: Part of speech tree (context size 4).

open word categories need to be considered. The categories considered are verbs, nouns, adjectives, and adverbs[4]. The part of speech tree is described in more detail in Section 5.3.

5.2 Letter to Sound Tree

5.2.1 Data

The data needed to construct a letter to sound tree are many pairs of "input letters" with an "output phoneme". These input/output pairs can be generated using the "sampaGd" lexicon, which contains words and their phonemic transcriptions. Unfortunately the letters do not directly map to phonemes. Depending on the context a letter can be mapped to zero, one, or several phonemes. Even if the number of letters in a word is equal to its number of phonemes, there is not necessarily a one-to-one relation between the letters and the phonemes. One letter could be related to two phonemes and in the same word two letters could be transcribed by a single phoneme, as for example in the word "Zimmer" with the phonemic transcription "`t s I m @ r`". The

[4] Another word category used in the "sampaGd" lexicon are numerals (NUM). Numbers not contained in the lexicon are treated in a similar way as unknown words. The differences are the method for determining the pronunciation and the word category (they are always classified as numerals).

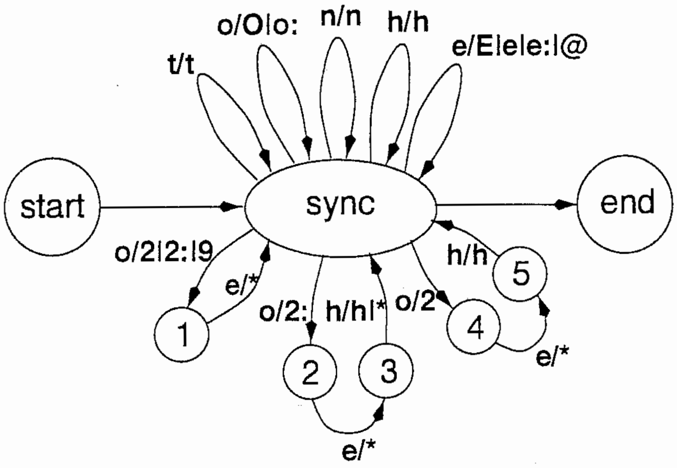

Figure 5.3: Part of the finite state transducer used to align the letters of a word to the phonemes of its phonemic transcription. For example, 'o/O|o:' means that by making this transition, the letter 'o' and either the phoneme 'O' or 'o:' will be consumed.

realization of the 'Z' are the two[5] phonemes "`t s`" and the letter sequence 'mm' is realized by a single phoneme `'m'`.

The main problem when preparing the lexicon entries for constructing a letter to sound tree is a consistent alignment between the letters and the phones. If this alignment is not carefully done, there will be a lot of inconsistencies and contradictions in the train data, which will result in a lower quality letter to sound tree.

The method used to align the letters and the phones was to manually set up a simple finite state transducer (FST). This FST simultaneously consumes letters and phonemes when going through the states of a finite state machine. Part of such an FST is shown in Figure 5.3. Only from the "sync" state the end state can be reached. The ambiguities arising when stepping through the FST are resolved by selecting the FST path with the highest numbers of non-sync states. If the FST reaches its end state, an alignment could be made.

In a first attempt a very simple FST was used. By analyzing the unalignable cases, the FST could be manually improved (mostly by adding more states to the FST). The improved FST was then used for alignment and again the unalignable cases were analyzed and the FST improved. This

[5] The combination of the two phonemes `'t'` and `'s'` could also be viewed as a single affricate. Here they will be treated as two separate phonemes.

was repeated until the set of unalignable words only consisted of non-German words and erroneously transcribed entries in the lexicon. This set consisted of 8176 entries out of a total of 319080 entries in the "sampaGd" lexicon.

For 310904 of the lexicon entries the orthographic word and its transcription could be aligned. For each letter in each word an input/output pair as needed to train the letter to sound tree can be determined. This will result in a very large train set, more than could be handled in a reasonable way.

An easy solution to reduce the amount of data would be to just randomly pick as many input/output pairs as needed. This way the distribution of the frequencies of the letters in the train data will be similar to the distribution in the lexicon. This distribution is very unbalanced since some letters occur much more often than others (e.g., 'n' occurs more often than 'q'). The letters that only occur a few times in the lexicon will occur even less often in the train data. The tree construction algorithm will work better with a more balanced train set. Additionally, especially those letter sequences involving non-sync states in the FST used for the alignment should be in the train data (that's where the interesting things happen).

The algorithm described in the following has therefore been used to extract the train data from the full set of data available[6]: In random order each of the aligned words was considered to be included in the train set. A word was included whenever it contained a letter sequence that has not already been included in the train data a certain number (this number has been set to 400) of times. Possible letter sequences are the sequences that begin and end in the sync-state of the FST that has been used for alignment. When separating a word into letter sequences, in ambiguous cases, longer sequences are preferred to shorter ones. A total of 18266 words have been selected.

5.2.2 Tree Construction

The order of the words included in the train data was then again randomized. 75% of this data (13700 words) was used to construct the letter to sound tree and the rest has been reserved to test the prediction accuracy of the tree. This resulted in 158427 input/output pairs available for tree construction and 52888 pairs for testing.

The software packages IND and C4.5 were used to construct letter to sound decision trees. The trees were constructed with IND using the following options (see [IND92] for further details):

[6] This algorithm selects words. Another approach (probably resulting in more balanced train data) would be to first generate all input/output pairs (needs lots of disk space) and then use a selection algorithm similar to the one described here.

SW and option	prediction accuracy	size
IND cart	95.9	382
IND c4	96.1	914
C4.5	97.9	17495
C4.5 -s	98.5	3276

Table 5.2: Prediction accuracy for the test data and the size (in number of nodes and leafs) of the pruned trees for different software packages and options used. A context of size 4 has been used.

```
mktree -e -s cart -v -v glts
```

and

```
mktree -e -s c4 -v -v glts
```

The following options were used with the C4.5 package (see [Qui93] for further details):

```
c4.5 -u -f glts
```

and

```
c4.5 -u -f glts -s
```

5.2.3 Results

Trees with different sizes of context (number of letters to the left and right of the current letter used as tree input) have been constructed. For the different software packages and sets of options used the resulting prediction accuracy (calculated with the test set) are shown in Table 5.2.

The decision tree software currently included in CHATR can only handle trees with binary splits. The C4.5 package produces trees with splits having more than two branches. To use these trees (which had higher prediction accuracies) they would either have to be transformed to binary trees or the CHATR decision tree software would need to be modified, which, because of time constraints, was not done in the context of this work.

In Table 5.3 the prediction accuracy for different context sizes are shown. These trees have been constructed with IND using the strategies "cart" and "c4". The tree constructed with the "cart" strategy and a context size of 2 resulted in a prediction accuracy of 96.4%. This tree is currently included in CHATR.

strategy	context	prediction accuracy	size
cart	1	93.2	482
cart	2	96.4	600
cart	3	96.2	432
cart	4	95.9	382
cart	6	95.9	360
c4	1	93.2	586
c4	2	96.6	1270
c4	3	96.4	872
c4	4	96.1	914
c4	6	96.0	902

Table 5.3: Prediction accuracy for the test data and the size (in number of nodes and leafs) of the pruned trees for different context sizes (only for IND).

5.3 Part of Speech Tree

5.3.1 Data

The part of speech (POS) tree predicts the POS of a word given the POS of a certain number (which is the context size) of preceding and following words. To train a decision tree for this purpose, syntactically correct part of speech sequences are needed. The categories used in the CHATR "sampaGd" lexicon are:

A	adjectives	**N**	nouns
ADV	adverbs	**NUM**	numerals
ART	articles	**PREP**	prepositions
C	conjunctions	**PRON**	pronouns
I	interjections	**V**	verbs

All these categories are possible input values to the part of speech tree. Additionally a dummy input value ("nopos") is needed for the context at the beginning and end of a sentence.

The sentences of the "aich" and "dlf" databases have been used to generate the train data. The "dlf" database includes complete POS sequences for all its sentences, but uses a different set of POS categories than the "sampaGd" lexicon. To generate the POS sequences used to train the POS tree, the POS categories of the "dlf" data were mapped to the POS categories of the "sampaGd" lexicon.

For the "aich" sentences (only text available) the POS sequences were generated by simply looking up the POS of each word in the "sampaGd" lexicon. For some (orthographic) words there are several entries with different POS and for other words there is no entry. This reflects the situation that will be encountered when using the part of speech tree in a TTS system. An additional input value (in this work denoted by "X"[7]) has therefore been added, which is used for the ambiguous and the unknown words occurring in the context.

With the POS sequences of the "aich" and "dlf" data, the input/output pairs used to train the part of speech tree could easily be generated. The possible POS categories predicted by the tree are verbs, adverbs, nouns, and adjectives. A total of 15665 input/output pairs (verbs: 25%, nouns: 38%, adjectives: 16%, and adverbs: 21%) were generated.

5.3.2 Tree Construction

75% of the input/output pairs (11749 pairs) was used to construct the part of speech tree and the rest was reserved for testing purposes.

The software package IND was used to construct part of speech decision trees. The strategies "cart" and "c4" have been used. The trees were constructed using the following options (see [IND92] for further details):

```
mktree -e -s cart -v -v posmark
```

and

```
mktree -e -s c4 -v -v posmark
```

5.3.3 Results

Part of speech trees using different context sizes were trained with the strategies "cart" and "c4". The resulting prediction accuracies and tree sizes (in number of nodes and leafs) are shown in Table 5.4. The highest accuracy (65.1%) could be observed when using the "cart" tree construction strategy for a context size of 4 (always predicting a noun would result in a 38% prediction accuracy).

Furthermore, the prediction accuracy could be slightly improved by including an input factor for punctuation. This factor indicates whether a comma or period directly follows the word for which the POS is being predicted. With this additional factor, using the "cart" strategy and a context size of 4, the prediction accuracy could be increased to 66.6% for the test data. This is the tree currently included in CHATR.

[7] Interjections, which did not occur in the train data, were mapped to the value "X".

strategy	context	prediction accuracy	size
cart	1	60.8	46
cart	2	64.2	232
cart	3	64.8	238
cart	4	65.1	100
cart	6	64.9	132
c4	1	61.0	84
c4	2	64.6	978
c4	3	64.0	3750
c4	4	63.7	4514
c4	6	62.4	5532

Table 5.4: Prediction accuracy for the test data and the size (in number of nodes and leafs) of the pruned trees for different context sizes.

5.4 Conclusions

With the letter to sound tree a pronunciation can be determined for any sequence of letters input to the system. The quality strongly depends on the type of word. The tree has been trained with German words and will work well for words following "standard" German pronunciation rules. The letter to sound tree will quite often fail for foreign (non-German) words and proper names.

To determine the syllable boundaries and lexical stress the same algorithms as for English are currently used. Syllabification could be improved by using a German-specific set of rules (see, e. g., [Tra95]). As shown above, a better heuristic for assigning lexical stress to unknown words would be to put a stress only on the first syllable. This heuristic strategy could be further improved by checking for words starting with a prefix known to be lexically unstressed.

In German the first letter of a noun is written with an uppercase letter. This has not been taken advantage of in the current part of speech tree. Some simple rule preceding the POS prediction could be added to the system which detects nouns. Other words and nouns at the beginning of a sentence would still need to be predicted by the part of speech tree. Another possibility would be to directly include the case of the first letter as an input to the tree.

The part of speech tree could also be used to resolve ambiguities in the lexicon. A Word can have several entries in the lexicon with different part of speech. Currently always the first entry is picked. A better strategy would be to predict the POS with the part of speech tree and then, based on the outcome of the prediction, select one of the lexicon entries.

Appendix A

CHATR Modifications

In the following the new and modified files are listed. These files work with CHATR 0.94 (July 1998).

Prosody

New files

dur_mars.c MARS duration prediction

dur_mars.h Header file for dur_mars.c

dur_lr.h Header file for dur_lr.c

tobi_f0_mars.c MARS fundamental frequency prediction

tobi_f0_mars.h Header file for tobi_f0_mars.c

mars.c MARS model functions

mars.h Header file for mars.c

dlf_dur_mars.model MARS duration model for German

dlf_f0_mars.model MARS fundamental frequency model for German

f2b_dur_mars.model MARS duration model for English

f2b_f0_mars.model MARS fundamental frequency model for English

Modified files

duration.c Added 'Duration_Method' option 'MARS'

duration.h Cleaned up

ToBI.c Bug fixes, added MARS F_0 method (is selected if 'tobi_f0_mars_method' is 'true')

pitch_range.c Cleaned up

ToBI Generation and Part of Speech Tree

The part of speech prediction is called inside the ToBI generation code.

New files

text_gtobi.c ToBI and part of speech prediction code

text_gtobi.h Header file for text_gtobi.c

gtone.ch Tone tree

gaccent.ch Accent tree

gposmark.ch Part of speech tree

Modified files

hlp.c Disable hlp_realise_accents if "text_prosody_strategy" is "DiscTree"

hlp_input.c Generate ToBI for text input if "text_prosody_strategy" is "DiscTree"

feats.c Interfaces tree feature functions to decision tree code

Letter to Sound Tree

New files

glts.c Letter to sound and text normalization code

glts.ch Letter to sound tree

Modified files

lexicon.c Added handling of unknown German words

lexicon.h Function prototype added

feats.c Interfaces letter to sound tree feature functions to decision tree code

Others

New files

sampaGd_def.ch Phoneme set definition

sampaGd_lexicon.ch Setup of sampaGd lexicon

Modified files

New files

chatr.c Debug output to terminal added

chatr_main.c Use tts instead of jtts for German

commands.c Added MARS to Parameter help

pf_input.c Added code to test lower level prosody

syllable.c Initialization of new files added in Syl struct

syllable.h Added several fields to the Syl struct

word.c Modify addition of boundaries for German text input

itlspeakers.ch Added entries for dlf and aich speech databases

reduce.ch Added entry for sampaGd in "schwas"

Appendix B

CHATR Setup

In the following the setup required to use the new modules is described. This assumes that you have a version of CHATR with the new and modified files described in Appendix A.

Prosody

- (Parameter Duration_Method MARS)
- (set tobi_f0_mars_method 'true)
- For aich database:

```
(set dur_mars_params
  '((model_file "/<somewhere>/dlf_dur_mars.model")
    (target_dur_mean 2.7569)      ; mean of dur^0.25
    (target_dur_sd   0.3831)))    ; sd of dur^0.25

(set tobi_f0_mars_params
  '((model_file "/<somewhere>/dlf_f0_mars.model")
    (target_f0_mean 101.0)      ; mean of f0 (nucleus)
    (target_f0_sd   22.0)))     ; sd of f0 (nucleus)
```

- For dlf database:

```
(set dur_mars_params
  '((model_file "/<somewhere>/dlf_dur_mars.model")
    (target_dur_mean 2.8107)      ; mean of dur^0.25
    (target_dur_sd   0.3609)))    ; sd of dur^0.25
```

```
(set tobi_f0_mars_params
  '((model_file "/<somewhere>/dlf_f0_mars.model")
    (target_f0_mean 97.4374)      ; mean of f0 (nucleus)
    (target_f0_sd   17.6273)))    ; sd of f0 (nucleus)
```

ToBI Generation and Part of Speech Tree

- Load tone tree “gtone_tree” with (load_library ”gtone.ch”)
- Load accent tree “gaccent_tree” with (load_library ”gaccent.ch”)
- Load part of speech tree “gposmark_tree” with (load_library ”gposmark.ch”)
- Do (set text_prosody_strategy ’DiscTree) to activate ToBI generation for German text input

Letter to Sound Tree

- Load letter to sound tree “glts_tree” with (load_library ”glts.ch”)
- Set (Lexicon Fail GLTS) to activate letter to sound tree for German

Others

- Load speech database with (speaker_dlf) or (speaker_aich)

Appendix C

Phoneme Set “sampaGd”

```
(Phoneme Def sampaGd
  ;na     vc  lng h   fr  rnd typ plc vox
(
( #       -   0   -   -   -   0   0   -)

( aI      +   d   3   1   -   0   0   +)
( aU      +   d   3   3   -   0   0   +)
( OY      +   d   3   2   +   0   0   +)

( i:      +   l   1   1   -   0   0   +)
( I       +   s   2   1   -   0   0   +)

( y:      +   l   1   1   +   0   0   +)
( Y       +   s   2   1   +   0   0   +)

( u:      +   l   1   3   +   0   0   +)
( U       +   s   2   3   +   0   0   +)

( e:      +   l   2   1   -   0   0   +)
( E       +   s   3   1   -   0   0   +)
( E:      +   l   3   1   -   0   0   +)

( 2:      +   l   2   1   +   0   0   +)
( 9       +   s   3   1   +   0   0   +)
( o:      +   l   2   3   +   0   0   +)
( O       +   s   3   3   +   0   0   +)

( a       +   s   3   2   -   0   0   +)
( a:     +   1   3   2   -   0   0   +)

( @      +   s   2   2   -   0   0   +)

( p      -   0   -   -   +   s   l   -)
( b      -   0   -   -   +   s   l   +)
( t      -   0   -   -   +   s   a   -)
( d      -   0   -   -   +   s   a   +)
( k      -   0   -   -   +   s   v   -)
( g      -   0   -   -   +   s   v   +)

( f      -   0   -   -   +   f   b   -)
( v      -   0   -   -   +   f   b   +)
( s      -   0   -   -   +   f   a   -)
( z      -   0   -   -   +   f   a   +)
( S      -   0   -   -   +   f   p   -)
( Z      -   0   -   -   +   f   p   +)
( x      -   0   -   -   +   f   v   -)
( C      -   0   -   -   +   f   p   -)
( h      -   0   -   -   +   f   v   -)
( j      -   0   -   -   +   f   p   +)

( m      -   0   -   -   +   n   l   +)
( n      -   0   -   -   +   n   a   +)
( N      -   0   -   -   +   n   v   +)

( l      -   0   -   -   +   l   a   +)
( r      -   0   -   -   +   l   a   +)
))
```

Bibliography

[ATR97] ATR ITL Department 2. *The CHATR User Guide.* ATR Interpreting Telecommunications Research Laboratories, 1997.

[BH94] M. E. Beckman and J. Hirschberg. *The ToBI Annotation Conventions.* 1994.

[BH96] A. W. Black and A. Hunt. *Generating F_0 contours from ToBI labels using linear regression.* Proceedings ICSLP, 1994.

[Bri97] C. Brinckmann. *German in Eight Weeks – A Crash Course for CHATR.* ATR Technical Report TR-IT-0236, 1997.

[Cam93] N. Campbell. *Multi-level timing in speech.* ATR Technical Report TR-IT-0035, 1993.

[Fri91] J. H. Friedman. *Multivariate Adaptive Regression Splines.* The Annals of Statistics, 1991.

[IND92] NASA Ames Research Center. *IND Software, Version 2.1.* http://ic-www.arc.nasa.gov/ic/projects/bayes-group/ind, 1992.

[May95] J. Mayer. *Transcription of German Intonation: The Stuttgart System.* Technical report, Universität Stuttgart, 1995.

[Moe98] G. Möhler. *Theoriebasierte Modellierung der deutschen Intonation für die Sprachsynthese.* PhD thesis, Universität Stuttgart, 1998.

[Qui93] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, 1993.

[Rey96] M. Reyelt, et al. *Prosodische Etikettierung des Deutschen mit ToBI.* Verbmobil Report 154, 1996.

[Rie97] M. Riedi. *Modeling Segmental Duration with Multivariate Adaptive Regression Splines.* Proceedings Eurospeech'97, 1997.

[Rie98] M. Riedi. *Controlling Segmental Duration in Speech Synthesis Systems.* PhD thesis, ETH-Zürich, Nr. 12487, 1998.

[Str97] K. Striegnitz. *Teaching CHATR German Intonation – Lesson One.* ATR Technical Report TR-IT-0237, 1997.

[Tra95] C. Traber. *SVOX: The Implementation of a Text-to-Speech System for German.* PhD thesis, ETH-Zürich, Nr. 11064, 1995.