

# AUTOMATIC DETECTION AND SEGMENTATION OF PRONUNCIATION VARIANTS IN GERMAN SPEECH CORPORA

*Andreas Kipp, Maria-Barbara Wesenick, Florian Schiel*

Institut für Phonetik und Sprachliche Kommunikation  
Universität München, Germany (IPSK)  
kip|wesenick|schiel@phonetik.uni-muenchen.de

## ABSTRACT

In this paper we present a hybrid statistical and rule-based segmentation system which takes into account phonetic variation of German. Input to the system is the orthographic representation and the speech signal of an utterance to be segmented. The output is the transcription (SAM-PA) with the highest overall likelihood and the corresponding segmentation of the speech signal. The system consists of three main parts: In a first stage the orthographic representation is converted into a linear string of phonetic units by lexicon lookup. Phonetic rules are applied yielding a graph that contains the canonic form and presumed variations. In a second HMM-based stage the speech signal of the concerning utterance is time-aligned by a Viterbi search which is constrained by the graph of the first stage. The outcome of this stage is a string of phonetic labels and the corresponding segment boundaries. A rule-based refinement of the segment boundaries using phonetic knowledge takes place in a third stage.

## 1. INTRODUCTION

For many applications in speech processing as in ASR and speech synthesis (e.g. PSOLA) reliable segmentation and labeling of large speech databases is required. Also as ASR increasingly uses discriminative techniques and tackles the challenge of analyzing spontaneous speech the demand for statistically based pronunciation models in different languages is growing.

Because of the large amount of data in today's speech corpora time-consuming manual segmentation is virtually impossible. Furthermore, it is subjective and prone to inconsistency, because no two human experts are likely to produce exactly the same segmentation for the same utterance. Not even the same trained person will come to exactly the same transcription if asked to repeat the segmentation of the same utterance [1].

On the other hand automatic methods like segmental-k-means are feasible, but mostly a forced alignment of the speech signal according to just one given linear string of labels is done. Hence, pronunciation variations occurring

in natural speech are mapped onto the segmental models of this phonetic unit sequence. These models are certainly able to model some of the pronunciation processes but not all: elisions and insertions can hardly be covered in this way. Furthermore the discriminative power of the models is weakened.

In previous work [2] this problem was addressed by optionally taking the phonetic unit sequence to be aligned from manual transcriptions instead of using a pronunciation dictionary for this purpose. This led to satisfactory results but, however, again involved manual transcriptions.

In this paper we present a system which accomplishes the detection of the pronunciation variant and its time-alignment in one step. The possible variants are obtained by applying pronunciation rules to the canonic form of an utterance. The term canonic form refers to the standard pronunciation of an utterance based on a pronunciation dictionary that has just one entry for each orthographic word. The canonic form is a simple transform (lexicon lookup and concatenation) of the orthographic representation and can be represented by a string of phonetic symbols. The main system divides into three parts which are described in the following sections:

- Generation of a graph which contains all presumed pronunciation variants (section 2.).
- HMM-based time alignment of this graph to the speech signal (section 3.).
- Refinement of the segment boundaries (section 4.).

The sections 5. and 6. show the results and give a short discussion.

## 2. GENERATION OF VARIANTS

A graph structure was chosen for representing the variants, because a simple list of possible variations, as used in previous work [5], turned out to be very time consuming and lead to redundant steps during time alignment.

The nodes of the graph correspond to phonetic symbols taken from the extended SAM Phonetic Alphabet of German [6]

and the edges to possible transitions which may have a probability associated with them. By choosing a path from the initial node of the graph to the terminal node a number of symbols are visited subsequently. These symbols make up a string of phonemes i.e. a possible pronunciation variant (or the canonic form) of an utterance. The following subsections describe what the rules look like and how they are applied to the canonic form to obtain the graph.

## 2.1. Set of Pronunciation Rules

The generation of the graph is based on a set of pronunciation rules. The rules were selected by analyzing manual transcriptions and extrapolating the results, with the aim that pronunciation processes well known from literature (e.g [3]) are also covered. Currently, the rule set consists of approx. 1500 rules. For details refer to [7].

A rule  $r_i, i = 0 \dots N - 1$  from the corpus consists of a symbol string on the left-hand side  $\mathbf{a}_i = \langle a_i(0), \dots a_i(K_i - 1) \rangle$  that has to match a substring of the canonic form and a symbol string on the right-hand side  $\mathbf{b}_i = \langle a_i(0), \dots a_i(L_i - 1) \rangle$  which represents the variation described by that rule.  $a_i(k)$  and  $b_i(l), k = 0 \dots K_i, l = 0 \dots L_i$  are phonetic symbols from the extended SAM-PA of German.

## 2.2. Application of the Rules

As a first step the canonic form of an utterance is represented as a graph with just one path from the initial to the terminal node. Along this path a start symbol followed by the phonetic symbols of the canonic form and finally an ending symbol are emitted. The resulting graph is called the canonic form graph  $G^{(0)}$ . Every node in this graph has just one successor (except for the terminal node).

In order to get the minimum number of nodes and edges that have to be added to  $G^{(0)}$  for each rule two additional quantities  $n_i$  and  $m_i$  are calculated for each rule, where  $n_i$  is the number of symbols that are identical at the beginning of  $\mathbf{a}_i$  and  $\mathbf{b}_i$  with  $a_i(k) = b_i(k), k = 0 \dots n_i - 1$ . Similarly  $m_i$  is the number of identical symbols at the end of  $\mathbf{a}_i$  and  $\mathbf{b}_i$  with  $a_i(K_i - k) = b_i(L_i - k), k = 1 \dots m_i$ . For these identical symbols no nodes have to be inserted.

Next, all rules are applied subsequently to  $G^{(0)}$  according to the algorithm described in Table 1. Note that rules are applied only to the canonic form graph  $G^{(0)}$ . In this way all presumed variations are covered in the graph without redundant nodes and edges. All hypotheses contained in the graph are judged to have an equal a priori probability. The edges get scored with transition probabilities to fulfill this presumption.

Figure 1 shows the graph of a single word. The initial and terminal nodes are marked with the symbols “<” and “>” respectively. Graphs of larger utterances may contain a huge number of hypotheses (up to  $2^{32}$  for a utterance of

```

for  $i = 0 \dots N - 1$ 
  if the graph  $G^{(0)}$  contains a node sequence  $\mathbf{n}_a$  which
  emits  $\mathbf{a}_i$  then
    if  $L_i - n_i - m_i > 0$  then
      add a node sequence  $\mathbf{n}_b$  of length  $L_i - n_i - m_i$  emit-
      ting the symbols  $b_i(l), l = n_i \dots L_i - m_i - 1$ ;
      mark first node of  $\mathbf{n}_b$  as start node  $N_{start}$  and last
      node of  $\mathbf{n}_b$  as end node  $N_{end}$  of alternative path
    else mark the node of  $\mathbf{n}_a$  emitting  $a_i(n_i - 1)$  as
     $N_{start}$  and the node emitting  $a_i(L_i - m_i)$  as  $N_{end}$ 
    (if either  $n_i = 0$  or  $m_i = 0$   $N_{start}$  or  $N_{end}$  are un-
    defined and not required in later processing)
    endif
    if  $n_i > 0$  then
      add a transition from the node of  $\mathbf{n}_a$  emitting
       $a_i(n_i - 1)$  to  $N_{start}$ 
    else keep in memory that transitions from all pre-
    decessors of the first node of  $\mathbf{n}_a$  to  $N_{start}$  have to
    be inserted (pending transitions)
    endif
    if  $m_i > 0$  then
      add a transition from  $N_{end}$  to the node of  $\mathbf{n}_a$  node
      emitting  $a_i(L_i - m_i)$ 
    else keep in memory that transitions from  $N_{end}$  to
    all successors of the last node of  $\mathbf{n}_a$  have to be in-
    serted (pending transitions)
    endif
  endif
end for
repeat
  add pending transitions from inserted nodes to succes-
  sors of nodes in  $G^{(0)}$  (This may increase the number of
  predecessors of other nodes in  $G^{(0)}$  and introduce new
  pending transitions);
  add pending transitions from predecessor nodes in  $G^{(0)}$ 
  to inserted nodes (This may increase the number of
  predecessors of other nodes in  $G^{(0)}$  and introduce new
  pending transitions);
until no more transitions have to be inserted

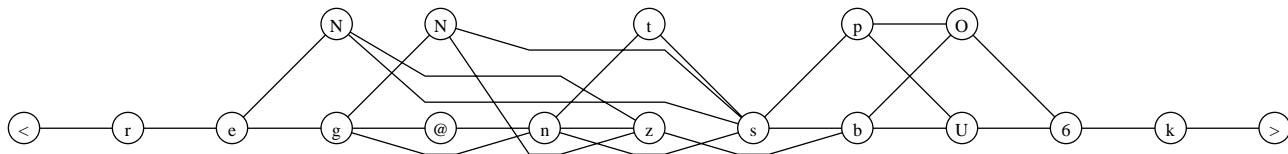
```

**Table 1:** Algorithm for the application of pronunciation rules

10s length).

## 3. HMM-BASED ALIGNMENT

In order to do the time alignment a data driven Viterbi beam search in a HMM state space constrained by the hypotheses contained in the graph is performed. We use context-free semicontinuous HMMs [8] modeling 42 the phoneme classes of SAM-PA. The statistical models have the following characteristics:



**Figure 1:** Graph containing all presumed variations of the word “Regensburg” /reg@nsbU6k/

- Features: 12 cepstral coefficients + energy + zero-crossing rate + first and 2nd derivative every 10ms.
- 5 codebooks, diagonal covariance matrices.
- 3 to 6 states per HMM.
- Initialization with data segmented by hand (2400 utterances from 12 speakers).

The state space is made up of all stages of HMMs which correspond to the symbols of nodes in the graph. If  $M$  is the number of nodes in the graph,  $S_m$  with  $m = 0 \dots M - 1$  the number of stages of the HMM corresponding to the node  $N_m$  and  $T$  is the number of time-steps (i.e. the number of feature-vectors to be processed), the state space is a  $(\sum_{m=0}^M S_m) \times T$  matrix.

At the first time step all successors of the initial node and a silence model are started up. That means that all grid points in the first time slot of the state space corresponding to initial states of these models are activated.

During the search active grid points are propagated according to the possible transitions within the HMM. Each time a state of a HMM is reached that allows a transition to another HMM, new models are launched according to the successor nodes in the graph. This is done by propagating the grid point of this state to grid points in the next time slot representing the initial states of these new models.

At each grid point in the next time slot the transitions between HMMs compete with those within HMMs and the best predecessor for each point is selected taking into account the acoustic score and the transition probabilities within HMMs and between the nodes of the graph.

Optionally, unlikely hypotheses i.e. grid points with low score may be pruned away. This speeds up the alignment essentially but however bears the risk of losing the hypothesis with the highest overall likelihood.

The procedure described above constrains the search to the variants included in the graph. The actual labeling and segmental information is obtained by backtracking of the Viterbi path.

## 4. REFINEMENT

Since the preprocessing computes the feature vector over a Hamming window of 20ms length which is shifted in 10ms

steps the boundaries obtained by the backtracking lay on a 10ms grid and have a (theoretical) inaccuracy of up to 10ms. Furthermore, some acoustic events cannot be properly modeled with a low time resolution like this.

The aim of the refinement stage is to correct the boundaries determined by the previous stage with methods that work on a much higher time resolution than the Viterbi preprocessing.

Currently a time domain method is used to shift the boundaries of vowels to the positive zero-crossing which precedes its peak amplitude. Other boundaries are simply shifted to the next zero-crossing.<sup>1</sup>

## 5. RESULTS

One possibility to estimate the quality of the automatic segmentations is to compare them to segmentations produced by hand. The difference in terms of the transcription symbols assigned to the speech signal and the segment boundaries has to be considered.

To compare two segmentations, first a dp-match is performed which finds the best match between their transcription symbols. We define  $M = \frac{2n_c}{(n_1+n_2)}$  as the match between the two segmentations where  $n_c$  is the number of corresponding symbols,  $n_1$  and  $n_2$  is the total number of symbols in each segmentation. For the evaluation of the segment boundaries a distribution of relative frequencies of the deviation is calculated. Only boundaries of subsequent segments, which have been assigned to the same symbols in both segmentation are considered.

A fundamental problem lies in the fact, that a unique correct transcription of an utterance does not exist. Therefore, a reference segmentation can only be defined arbitrarily. Instead of selecting a single transcription as a reference, we compared as many transcriptions of the same data as available to each other and to the automatic transcriptions.

Table 2 shows the average match  $M_0$  between 3 different manual segmentations of one speaker (200 utterances) from the PHONDAT II [6] corpus and an automatic segmentation of the same data. As it can be seen the human segmenters differ less from each other (match between 93.1% and 94.4%) than from the automatic segmentations (match

<sup>1</sup>These guidelines are obligatory at the IPSK for manual transcriptions. They are also applied to automatic transcriptions for comparability

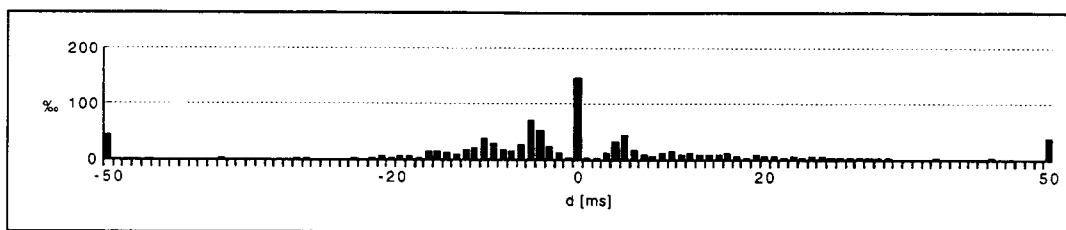


Figure 2: Distribution of relative frequencies of boundary deviation  $d$

	chr	kat	man	AUT
chr	100.0	93.5	93.1	88.1
kat	93.5	100.0	94.4	87.5
man	93.1	94.4	100.0	88.2

Table 2: Comparison between 3 manual segmentations (chr, man, kat) and an automatic (AUT) of one speaker (200 utterances) of the PHONDAT II corpus. The numbers give the average match  $M$  in percent. See text for details

between 87.5% and 88.2%), but the difference is less than 7%.

Figure 2 shows the distribution of relative frequencies of the boundary deviation  $d$  between an automatic and a manual segmentation. About 15% of all evaluated segment boundaries match exactly ( $|d| < 0.5\text{ms}$  deviation). There are some equidistant maxima with decaying relative frequency. Their distance is approximately the pitch period, because the refinement stage shifts the boundaries to the zero-crossings preceding the peak amplitude. The two peaks at the edges of the range are the sum of extreme outliers ( $|d| \geq 50\text{ms}$  deviation). On an average 59% of all boundaries differ less than 10ms (basis: 1 speaker PHONDAT II, 200 sentences, 3 manual segmentations vs. one automatic segmentation).

A thorough analysis of the results obtained with this systems going into phonetic details can be found in [4].

## 6. DISCUSSION AND FUTURE WORK

The results show that high quality segmentations of speech signals which can compete with manual ones may be obtained automatically if phonetic knowledge is incorporated in the segmentation process. In our approach a set of pronunciation rules is the basis of this knowledge. It is generic and not fine tuned to any corpus. The aim is to cover as many variants as possible, even if they are not very likely and to let the acoustics, i.e. the statistic models decide, which is the most likely to have been occurred. Therefore the rule set is quite large. However, this requires a powerful HMM stage because with a growing number of hypotheses contained in the graph, the task of aligning tends more and more to speech recognition.

A reliable statistical survey of pronunciation variants on the other hand, which could be used to control the Viterbi search by pruning away unlikely variants, is hard to obtain, because the available amount of speech data segmented by hand is not sufficient for this purpose. A feasible way would be to start with a large rule set and carefully train it to the task by biasing variants that occur frequently during segmentation. This leads of course to a task specific rule set.

As we are currently extending the system to spontaneous speech, which naturally contains more pronunciation variants than read speech, a large rule set is certainly necessary. Another way to increase the performance of the system is to improve the HMM stage. Therefore we are integrating a powerful ASR-system for spontaneous speech in our system. Preliminary test show encouraging results.

## 7. REFERENCES

1. B. Eisen, H. G. Tillman, and C. Draxler, *Consistency of judgments in manual labeling of phonetic segments: The distinction between clear and unclear cases*, Proc of the ICSLP (Banff), 1992, pp. 871-874.
2. Brugnara and Falavigna, *Automatic segmentation and labeling of speech based on hidden markov models*, Speech Communication 12 (1993), no. 4, 357-370.
3. K. Kohler, *Einführung in die Phonetik des Deutschen*, E. Schmitd, Berlin, 1977.
4. M.-B. Wesenick and A. Kipp, *Estimation the quality of phonetic transcriptions an segmentations of speech signals*, Proc. of the ICSLP (Philadelphia), 1996.
5. M.-B. Wesenick and F. Schiel, *Applying speech verification to a large data base of german to obtain a statistical survey about rules of pronunciation*, Proc. of the ICSLP (Yokohama), vol. 1, 1994, pp. 279-282.
6. B. Pompino-Marschall, *Phondat. Verbundvorhaben zum Aufbau einer Sprachsignalatenbank für gesprochenes Deutsch*, Forschungsberichte des Instituts für Phonetik und Sprachliche Kommunikation der Universität München (FIPKM), 1992, pp. 99-128.
7. M.-B. Wesenick, *Automatic generation of german pronunciation variants*, Proc. of the ICSLP (Philadelphia), 1996.
8. X.D. Huang and M.A. Jack, *Hidden markov modeling of speech, based on semicontinuous model*, Electronic Letters 24 (1988), no. 1, 6-7.