Learning Pronunciation Dictionary from Speech Data

Christian-M. Westendorf, Jens Jelitto

Dresden University of Technology
Institute for Technical Acoustics
Mommsenstr. 13, 01062 Dresden
GERMANY

email: west|jelitto@eakss1.et.tu-dresden.de

ABSTRACT

In this paper an algorithm and first results from our investigations in automatically learning pronunciation variations from speech data are presented. Pronunciation dictionaries establish an important feature in state-of-the-art speech recognition systems. In most systems only simple dictionaries containing the canonical pronunciation forms are implemented. However, for a good recognition performance more sophisticated dictionaries including pronunciation variations are essential. The generation of such dictionaries by hand is an extremely time consuming task, and the introduction of errors and inconsistencies is probable. We show an approach for automatically generating suitable pronunciation dictionaries from the speech data base itself, as they are desirable not only for speech recognition tasks but also for speech technology and phonologic research in general. The only knowledge sources besides the data base are the (unlabeled) signals and their transliterations on word level. First experiments yielding promising results have been performed with the software system DataLab [6], which integrates the recognition system of the TU Dresden.

1. INTRODUCTION

Since the phonetic dictionary represents the interface between speech analysis on the acoustic level and speech interpretation, its quality is extremly important for the overall performance of speech recognition systems. In many systems simple canonical pronunciation forms are used within the dictionaries. They represent the 'correct' pronunciations as they are to be found in lexica. However, in most cases the 'correct' pronunciation of a word doesn't have to do very much with the actual realization of this word.

Some of the main sources of speech variability are: regional differences, speaking style, age and social background of the speaker. Furthermore, even if one speaker realizes one word several times, the pronunciation will always be different. Therefore, an adequate representation of this variability has to be introduced.

A first approach for solving this problem is the generation of pronunciation variants by hand. In several more advanced systems a set of phonological rules is applied to introduce pronunciation variability [1] [4]. It could be shown, that systems using such extended dictionaries outperform systems using canonical pronunciation forms.

However, manual and rule based modification of the dictionary has several disadvantages. First, it is extremly time consuming both to include new pronunciation variants by hand or to generate suitable pronunciation rules for the investigated text corpus. Second, it needs specialized experts to handle such a task, but different experts will not get identical results. This may lead to inconsistencies within the dictionary representation. Third, the formal superposition of pronunciation rules, which are modeling different effects of the investigated language, may introduce errors in the sense of correct variability modeling.

Because of the disadvantages of manual and rule based dictionary modification an approach was chosen to generate the pronunciation variants automatically from the speech data base. Several researchers tried to tackle the problem this way [3] [2] [8]. The main advantage of such a method is the use of the data base as knowledge source. This will lead to consistent dictionaries containing only such variabilities, which actually occured in the data or which were introduced by the phoneme recognizer. Some of the aims of this approach are:

- to accelerate and simplify the dictionary generation process even for new domains,
- to increase the consistency and representativity of the dictionaries and the statistical reliability of the pronunciation variants,
- to find an algorithm for converting dictionaries, which are including particularities of the recognizer, into dictionaries, which model the 'pure' speech variability, and vice versa,
- on the basis of 'pure' pronunciation dictionaries phonetic-phonological investigations and rule verification are intended.

A special feature of the introduced approach is the matching of two graphs for the dictionary update. One graph represents the phoneme sequence generated from the text transcription including pronunciation variabilities introduced by the dictionary. The second graph contains alternative phoneme hypothesis from the phoneme recognizer output. The alignment of the graphs is based on a generalized VITERBI-search.

2. ALGORITHM

The iterative training algorithm is given by the following steps:

1. Training of a phoneme recognizer (i.e. of HMM type), which is able to produce a lattice of phonemic hypotheses from the speech signal.
2. For a given utterance the lattice of phonemic hypotheses is computed using the phoneme recognizer.
3. From the given text transcription a reference lattice is generated using the current state of the dictionary.
4. The best hypotheses sequence is computed by a common DP algorithm. This sequence has to fullfil conditions of connectivity and has to define the best match to the transcription of the spoken sentence as well.
5. After steps 2...4 have been executed for all given utterances the dictionary can be updated by the new computed phoneme sequences related to the dictionary entries. New detected nodes and transitions are added to the dictionary.
6. The steps 2...5 may be done iteratively using the current state of the dictionary for generating reference lattices.

2.1. Phoneme Hypothesis Lattice Generation

To generate the phoneme hypotheses, different methods are available. First, the categories have to be modeled by a trainable approach. A simple statistical model basing on frames or windows of the parameterized signal can be used to calculate a-posteriori-probabilities of the selected categories (see fig. 1). From these discrete hypotheses can be derived using a smoothing procedure and adaptive thresholds.

Better results may be obtained by a HMM-based model of the signal. Each category is modeled by 3...5 states. In this work, a HMM with 156 states for 50 categories has been applied [7]. The phonemic hypotheses are generated either by adaptive thresholding or by using a DP-based hypothesis

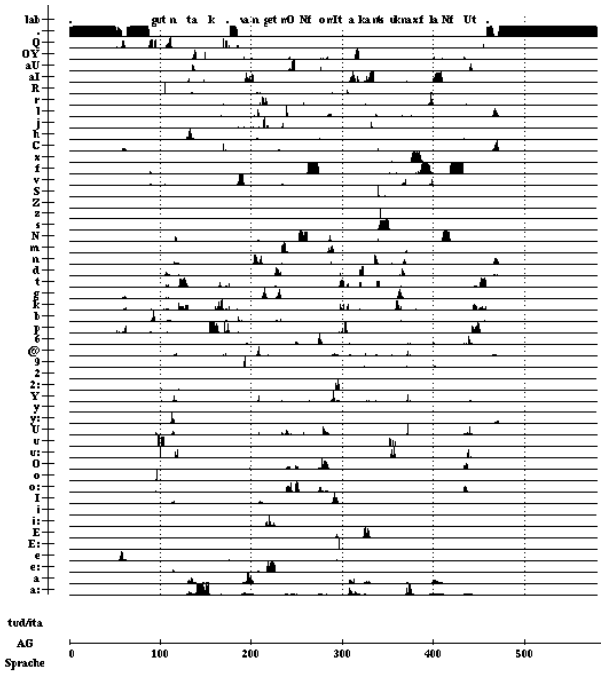

Figure 1: a-posteriori-probabilities for 50 phoneme classes

detection technique. Some restrictions concerning duration and score may be used to limit the number of the hypotheses.

In the second step, the phonemic hypotheses have to be connected to a lattice $\mathbf{\Gamma_1}$. The connection is restricted by time conditions like maximum gap width, maximum overlap and minimum forward time. Phonotactic restrictions are not used in this step [5].

2.2. Dictionary Representation

Because the dictionary has to be trainable, a flexible graph representation is essential. A dictionary compiler generates the dictionary graph representation from the common string representation of the standard pronunciation as input. The strings processed by the compiler may contain optional parts and alternatives as well, i.e. `mo:{[r][g][6]|N}` for the german word `morgen`. {.|.} denote alternatives, while [.] stand for optional substrings. By this means, known pronunciation rules may be included into the dictionary before starting the training process.

For a given sentence, a graph representation $\mathbf{\Gamma_2}$ is derived from the dictionary chaining the word graphs according to the word order. The nodes of $\mathbf{\Gamma_2}$ correspond to the phonemic categories depending on the sentence context.

2.3. Best Map Search

In step 4 of the algorithm, the best match between a path in the hypotheses lattice and the symbolic representation of the sentence has to be found. All possible matches are pathes

within the product of the two graphs $\mathbf{\Gamma_1}$ and $\mathbf{\Gamma_2}$ (see fig. 2).

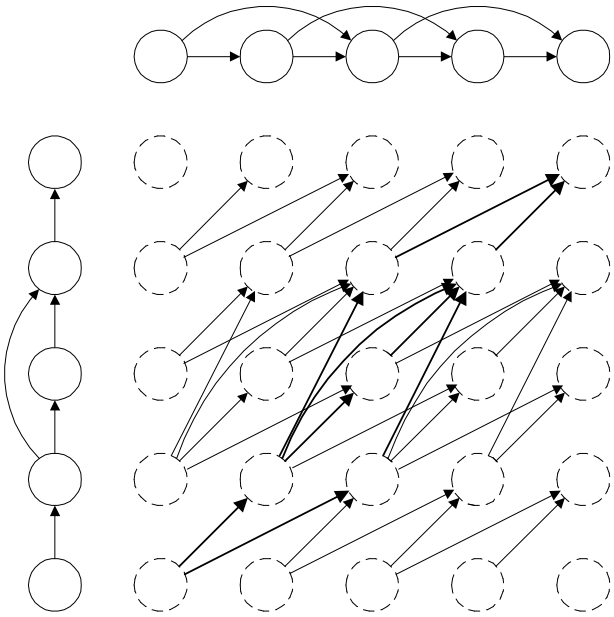

Figure 2: Example for the matching of two graphs with a generalized VITERBI -search

The set of nodes in the product graph is defined by the set product $\mathbf{S_1} \times \mathbf{S_2}$, where $\mathbf{S_1}$ denotes the nodes of $\mathbf{\Gamma_1}$ and $\mathbf{S_2}$ denotes the nodes of $\mathbf{\Gamma_2}$. Two nodes in the product graph are connected by an arc, if the corresponding nodes in the graphs $\mathbf{\Gamma_1}$ or $\mathbf{\Gamma_2}$ are connected. Insertions and deletions are modeled by loops added to $\mathbf{\Gamma_1}$ and $\mathbf{\Gamma_2}$ (removed in fig. 2 for simplicity).

The score of a node in the product graph depends on the two categories assigned to the node (hypothesis category and category from dictionary). In our version, the score expresses the compatibility of the categories. In principal, all confusions are allowed, but some confusions are preferable (i.e. confusions within the groups of vowels, plosives etc.).

The best map is searched by a DP algorithm applied to the product graph.

2.4. Dictionary Update

A best node sequence results from the DP search of the best map. The dictionary is updated by adding this node sequence. If new arcs and nodes are necessary, they are added to the dictionary. The appearence of nodes and arcs is counted in order to get arc and node probabilities.

As the experiments show, a check of the node sequences is necessary to avoid errors in dictionary update. The number of new nodes and arcs should be limited on a statisical basis. After one ore more iterations nodes and transitions should be deleted according to their number of occurence.

2.5. Implementation

The algorithm was implemented on DECstation 5000 under UNIX using the software system DataLab [6]. This systems performs all steps of the algorithms from signal analysis up to dictionary compilation and update. The algorithm is described by a special script language. Experiments are carried out using the PHONDAT database from the German project VERBMOBIL.

3. EXPERIMENTAL RESULTS

Several experiments have been carried out to prove the potentials of the introduced algorithm and to opimize the parameters. The following examples show some pronunciation variants introduced by the algorithm.

Example 1 shows the standard pronunciation of the German word 'fahren', as it can be found in the dictionary, and two variations resulting from the VITERBI-alignment.

f a: r @ n
f a: n
f o: a: n

In both pronunciation variants the phonemes 'r' and '@' are deleted, and in the second variation an additional phoneme 'o:' is inserted. Figure 3 shows the resulting structure of the dictionary entry for the word 'fahren' after the update.

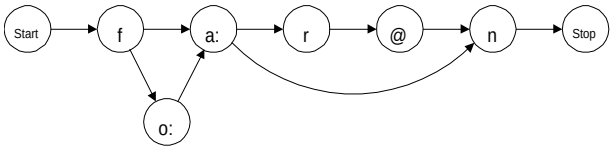

Figure 3: dictionary structure for the word 'fahren'

The second example shows a typical substitution for the realization of the word 'morgen'. The phoneme combination 'o r g @' is substituted by the phoneme 'OY', the realization of this word is shortened drastically. Furthermore, 'n' was substituted by the Nasal 'N'.

m o r g @ n
m OY N

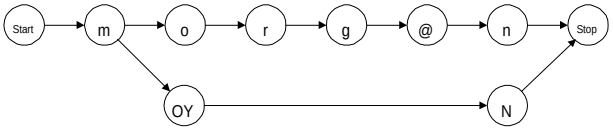

Figure 4: dictionary structure for the word 'morgen'

The last two examples show some more substitutions found by the algorithm.

h a n o: f 6
h a n o: f a

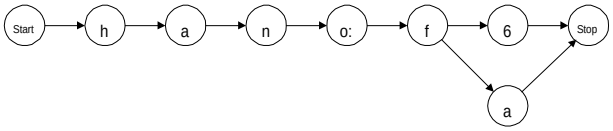

Figure 5: dictionary structure for the word 'Hannover'

h a m b U r k
h a N b U r k
h a m b u: r k

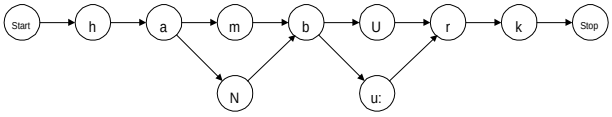

Figure 6: dictionary structure for the word 'Hamburg'

Variations can be caused by two different effects. First, the variation can be caused by the actual realization of the word. This is for instance highly probable for the shortened version of the word 'morgen'. Second, a variation can be introduced by the phoneme recognizer. If, for instance, the classifier often confuses the phonemes 'm' and 'N', the first variation of 'Hamburg' could be caused by the recognition system. As a result, both sources of variability are modeled in the dictionary. By means of special knowledge sources about the recognizer, such as the confusion matrix, it should be possible to separate the different influences.

4. CONCLUSIONS

The first experiences with the algorithm described in this paper allows different conclusions:

1. Some effort has to be done to get phonemic hypotheses with increased reliabilty. The frame-by-frame recognition rate for 50 categories has been estimated as about 50 %, but it strongly depends from the category. Most of the problems occures within the categories of the plosives or phonemes like `h,j,v,l`.
2. The matching process should be controlled by more sophisticated scores to avoid a complete mismatch between the hypothesis lattice $\mathbf{\Gamma_1}$ and the word lattice $\mathbf{\Gamma_2}$.
3. For a complete automatic learning procedure the detection of mismatch is needed. Some conditions and restrictions for dictionary update have to be introduced.

A general problem of the update of dictionaries on the word level is the lackof generalization ability. For this reason, the update should be performed at a lower lingustic level (i.e. at the level of syllables).

Nevertheless, the algorithm seems to be a practicable approach for studying the problems of automatic dictionary update. It is planned to apply this method also to larger amounts of spontaneous speech data.

5. ACKNOWLEDGEMENTS

This work was partially funded by the German Federal Ministry of Education, Science, Research and Technology (BMBF) in the framework of the Verbmobil Project under Grant 01 IV 102 K. The responsibility for the contents of this study lies with the authors.

6. REFERENCES

1. Xavier Aubert and Christian Dugast. Improved acoustic-phonetic modeling in PHILIPS' dictation system by handling liasions and multiple pronunciations. In *Proceedings of the EUROSPEECH'95*, volume 1, pages 767–770, Madrid, 1995.
2. Philipp Schmid, Ronald Cole, and Mark Fanty. Automatically generated word pronunciations from phoneme classifier output. In *Proceedings of the ICASSP 1993*, volume 2, pages II–223–II–226, Minneapolis, MN, 1993.
3. Tilo Sloboda. Dictionary learning: Performance through consistency. In *Proceedings of the ICASSP 1995*, volume 1, pages 453–456, Detroit, MI, 1995.
4. Maria-Barbara Wesenick and Florian Schiel. Applying speech verification to a large data base of german to obtain a statistical survey about rules of pronunciation. In *Proceedings of the ICSLP 1994*, volume 1, pages 279–282, Yokohama, 1994.
5. Christian-M. Westendorf. Erkennung fliessender Sprache auf der Basis diskreter Hypothesen – eine Alternative zu HMM? In *Elektronische Sprachsignalverarbeitung. Tagungsband der sechsten Konferenz*, Studientexte zur Sprachkommunikation, Bd. 12, S. 85–96, Wolfenbüttel, 1995.
6. Christian-M. Westendorf. DataLab - eine interaktive Toolbox für Signalanalyse und Mustererkennung. In *Tagungsband der DAGA'96*, Bonn, 1996. to appear.
7. Christian-M. Westendorf and Jens Jelitto. Vergleichende Untersuchungen zur Phonemerkennung. In *Tagungsband der DAGA'96*, Bonn, 1996. to appear.
8. Chuck Wooters and Andreas Stolcke. Multiple-pronunciation lexical modeling in a speaker independent speech understanding system. In *Proceedings of the ICSLP 1994*, volume 3, pages 1363–1366, Yokohama, 1994.