Munich AUtomatic Segmentation (MAUS)

Phonemic Segmentation and Labeling using the MAUS Technique

F. Schiel, Chr. Draxler, J. Harrington

Bavarian Archive for Speech Signals
Institute of Phonetics and Speech Processing
Ludwig-Maximilians-Universität München, Germany

www.bas.uni-muenchen.de
info@bas.uni-muenchen.de

Overview

- Statistical Segmentation and Labeling
- Super Short Introduction to MAUS
- Pronunciation Model : Building the Automaton
- Pronunciation Model : From Automaton to Markov Model
- Evaluation of Segmentation and Labeling
- Software Package MAUS

Statistical Segmentation and Labeling

Let Ψ be all possible Segmentation & Labeling (S&L) for a given utterance.
Then the search for best S&L $\hat{K}$ is:

$$\hat{K} = \text{argmax}_{K \in \Psi} P(K|o) = \text{argmax}_{K \in \Psi} \frac{P(K)p(o|K)}{p(o)}$$

with o the acoustic observation of the signal.
Since $p(o)$ = const for all K this simplifies to:

$$\hat{K} = \text{argmax}_{K \in \Psi} P(K)p(o|K)$$

with: $P(K)$ = apriori probability for a label sequence,
$p(o|K)$ = the acoustical probability of o given K
(often modeled by a concatenation of HMMs)

Statistical Segmentation and Labeling

S&L approaches differ in creating Ψ and modeling $P(K)$

For example: *forced alignment*

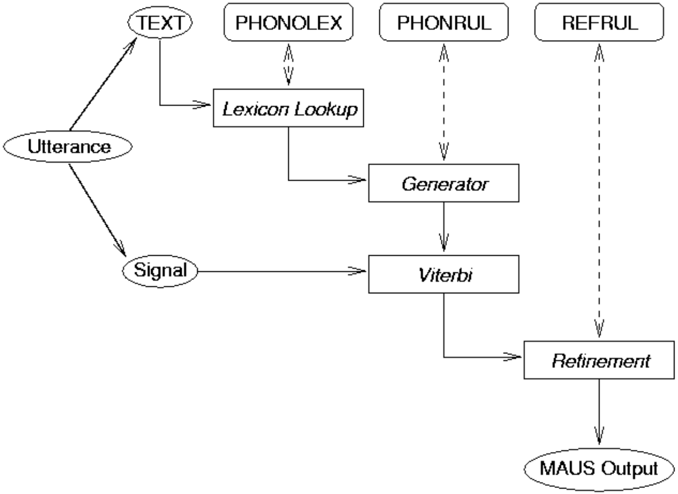
$$||\Psi|| = 1 \qquad \text{and} \qquad P(K) = 1$$

hence only $p(o|K)$ is maximized.

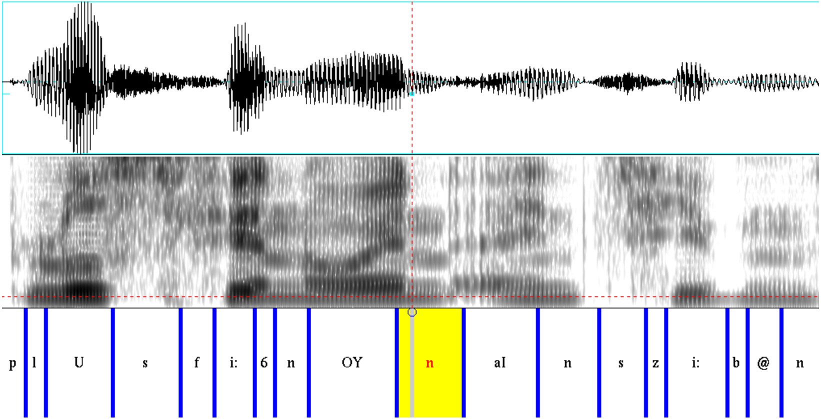
Other ways to model Ψ and $P(K)$:

- phonological rules resulting in M variants with $P(K) = \frac{1}{M}$
- phonotactic n-grams
- lexicon of pronunciation variants
- **Markov process** (MAUS)

Short Introduction to MAUS



Short Introduction to MAUS



Building the Automaton

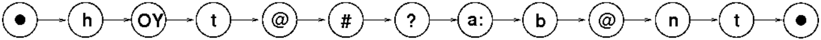
Start with the orthographic transcript:

heute Abend

By applying lexicon-lookup and/or a test-to-phoneme algorithm produce a (more or less standardized) citation form in SAM-PA:

hOYt@ ?a:b@nt

Add word boundary symbols #, form a linear automaton $\mathcal{G}_c$:



Building the Automaton

Extend automaton $\mathcal{G}_c$ by applying a set of substitution rules q_k where each $q_k = (a, b, l, r)$ with
a : pattern string
b : replacement string
l : left context string
r : right context string

For example the rules

(/@n/,/m/,/b/,/t) and (/b@n/,/m/,/a:/,/t/)

generate the reduced/assimilated pronunciation forms

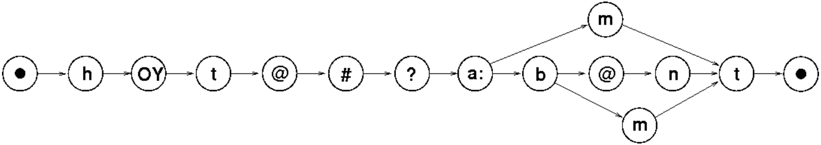
/?a:bmt/ and /?a:mt/

from the canonical pronunciation

/?a:b@nt/ (*evening*)

Building the Automaton

Applying the two rules to $\mathcal{G}_c$ results in the automaton:



From Automaton to Markov Process

Add transition probabilities to the arcs of $\mathcal{G}(N, A)$

- Case 1 : all paths through $\mathcal{G}(N, A)$ are of equal probability
  Not trivial since paths can have different lengths!
  Transition probability from node d_i to node d_j:

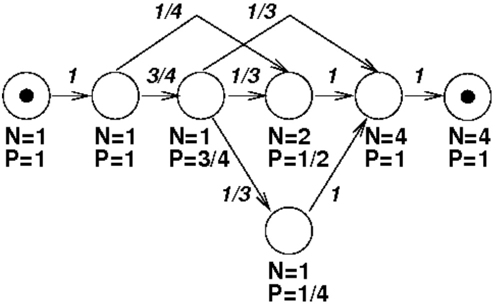
$$P(d_j|d_i) = \frac{P(d_j)N(d_i)}{P(d_i)N(d_j)}$$

$N(d_i)$: number of paths ending in node d_i
$P(d_i)$: probability that node d_i is part of a path

$N(d_i)$ and $P(d_i)$ can be calculated recursively through $\mathcal{G}(N, A)$ (see Kipp, 1998 for details).

From Automaton to Markov Process

Example:
Markov process with 4 possible paths of different length



Total probabilities:

$$1 \cdot \frac{3}{4} \cdot \frac{1}{3} \cdot 1 \cdot 1 = \frac{1}{4}$$
$$1 \cdot \frac{1}{4} \cdot 1 \cdot 1 = \frac{1}{4}$$
$$1 \cdot \frac{3}{4} \cdot \frac{1}{3} \cdot 1 = \frac{1}{4}$$
$$1 \cdot \frac{3}{4} \cdot \frac{1}{4} \cdot 1 \cdot 1 = \frac{1}{4}$$

From Automaton to Markov Process

- Case 2 : all paths through $\mathcal{G}(N, A)$ have a probability according to the individual rule probabilities along the path through $\mathcal{G}(N, A)$

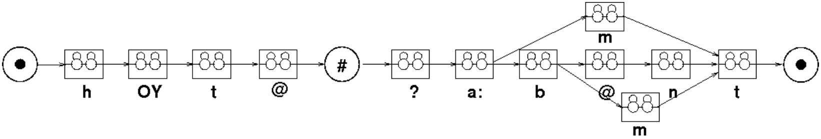
  Again not trivial, since contexts of different rule applications may overlap!
  This may cause total branching probabilities > 1

  Please refer to Kipp, 1998 for details to calculate correct transition probabilities.

From Markov Process to Hidden Markov Model

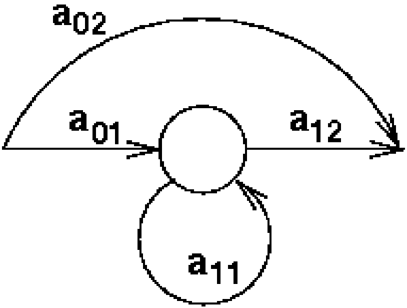
True HMM : add emission probabilities to nodes N of $\mathcal{G}_c$.

-> Replace the phonemic symbols in N by mono-phone HMM.
The search lattice for previous example:



From Markov Process to Hidden Markov Model

Word boundary nodes '#' are replaced by a optional silence model:



Possible silence intervals between words can be modeled.

Evaluation of Segmentation and Labeling

How to evaluate a S&L system?

Required: reference corpus with hand-crafted S&L ('gold standard').

Usually two steps:

1. Evaluate the accuracy of the label sequence (transcript)
2. Evaluate the accuracy of segment boundaries

Evaluation of Label Sequence

Often used for label sequence evaluation: Cohen's κ

κ = amount of overlap between two transcripts (system vs. gold standard); independent of the symbol set size (*Cohen 1960*).

We consider κ not appropriate for S&L evaluation, since

- no gold standard exists in phonemic S&L
- different symbol set sizes do not matter in S&L
- the task difficulty is not considered
  (e.g. read vs. spontaneous speech)

Evaluation of Label Sequence

Proposal: *Relative Symmetric Accuracy (RSA)* =
= the ratio from average symmetric system-to-labeler agreement $\widehat{SA}_{hs}$ to average inter-labeler agreement $\widehat{SA}_{hh}$.

$$RSA = \frac{\widehat{SA}_{hs}}{\widehat{SA}_{hh}} 100\%$$

Evaluation of Label Sequence

German MAUS:

- 3 human labelers
- spontaneous speech (Verbmobil)
- 9587 phonemic segments

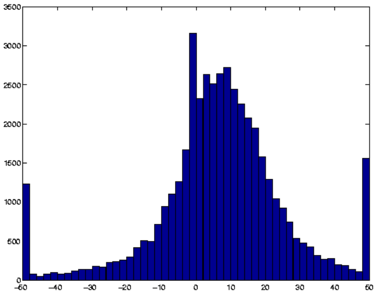
Average system - labeler agreement	$\widehat{SA}_{hs} = 81.85\%$
Average inter - labeler agreement	$\widehat{SA}_{hh} = 84.01\%$
Relative symmetric accurarcy	$RSA = 97.43\%$

Evaluation of Segmentation

- No standardized methodology
- Problem: insertions and deletions
- Solution: compare only matching segments
- Often: count boundary deviations greater than threshold (e.g. 20msec) as errors
- Better: deviation histogram measured against all human segmenters

Evaluation of Segmentation

German MAUS:



Note: center shift typical for HMM alignment

Software Package MAUS

MAUS software package:

ftp://ftp.bas.uni-muenchen.de/pub/BAS/SOFTW/MAUS

MAUS requires

- UNIX System V or *cygwin*
- Gnu C compiler
- HTK (*University of Cambridge*)

Current language support: German, English, Hungarian, Icelandic, Estonian, Portuguese, Spanish

A MAUS web services is currently in alpha.
If interested in a demo, please contact me after the talk.

References

- Kipp A (1998): Automatische Segmentierung und Etikettierung von Spontansprache. *Doctoral Thesis*, Technical University Munich.
- Wester M, Kessens J M, Strik H (1998): Improving the performance of a Dutch CSR by modeling pronunciation variation. *Workshop on Modeling Pronunciation Variation*, Rolduc, Netherlands, pp. 145-150.
- Kipp A, Wesenick M B, Schiel F (1996): Automatic Detection and Segmentation of Pronunciation Variants in German Speech Corpora. *Proceedings of the ICSLP*, Philadelphia, pp. 106-109.
- Schiel F (1999) Automatic Phonetic Transcription of Non-Prompted Speech. *Proceedings of the ICPhS*, San Francisco, August 1999. pp. 607-610.
- MAUS: *ftp://ftp.bas.uni-muenchen.de/pub/BAS/SOFTW/MAUS*
- Draxler Chr, Jänsch K (2008): WikiSpeech – A Content Management System for Speech Databases. *Proceedings of Interspeec* Brisbane, Australia, pp. 1646-1649.
- CLARIN: *http://www.clarin.eu/*
- Cohen J (1960): A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* 20 (1): 37-46.
- Fleiss J L (1971): Measuring nominal scale agreement among many raters. *Psychological Bulletin*, Vol. 76, No. 5 pp. 378-382.
- Burger S, Weilhammer K, Schiel F, Tillmann H G (2000): Verbmobil Data Collection and Annotation. In: *Verbmobil: Foundations* *Speech-to-Speech Translation* (Ed. Wahlster W), Springer, Berlin, Heidelberg.
- Schiel F, Heinrich Chr, Barfüßer S (2011): Alcohol Language Corpus. *Language Resources and Evaluation*, Springer, Berlin, Ne York, in print.

Software Package MAUS

How to adapt MAUS to a new language?

Several possible ways (in ascending performance and effort):

- Define a mapping from the phoneme set of the new language to the German set (or any other available language in MAUS). Constrain pronunciation to canonical form.
  Effort: nil
  Performance: for some languages surprisingly good.

Software Package MAUS

- Hand craft pronunciation rules (depending on language not more than 10-20) and run MAUS in the 'manual rule set' mode.
  Effort: small
  Performance: Very much dependent of the language, the type of speech, the speakers etc.
- Adapt HMM to a corpus of the new language using an iterative training schema (script `maus.iter`). Corpus does not need to be annotated.
  Effort: moderate (if corpus is available)
  Performance: For most languages very good, depending on the adaptation corpus (size, quality, match to target language etc.)

Software Package MAUS

- Retrieve statistically weighted pronunciation rules from a corpus. The corpus needs to be at least of 1 hour length and segmented/labeled manually.
  Effort: high.
  Performance: Unknown.