Vorlesung

Sprachdialogsysteme

Timo Baumann
baumann@informatik.uni-hamburg.de

https://nats-www.informatik.uni-hamburg.de/SDS19

Universität Hamburg, Department of Informatics
Language Technology Group

Heute

Reprise Spracherkennung

Sprachsynthese in a nutshell

- spezifische Schwierigkeiten der "Text-to-Speech"-Synthese

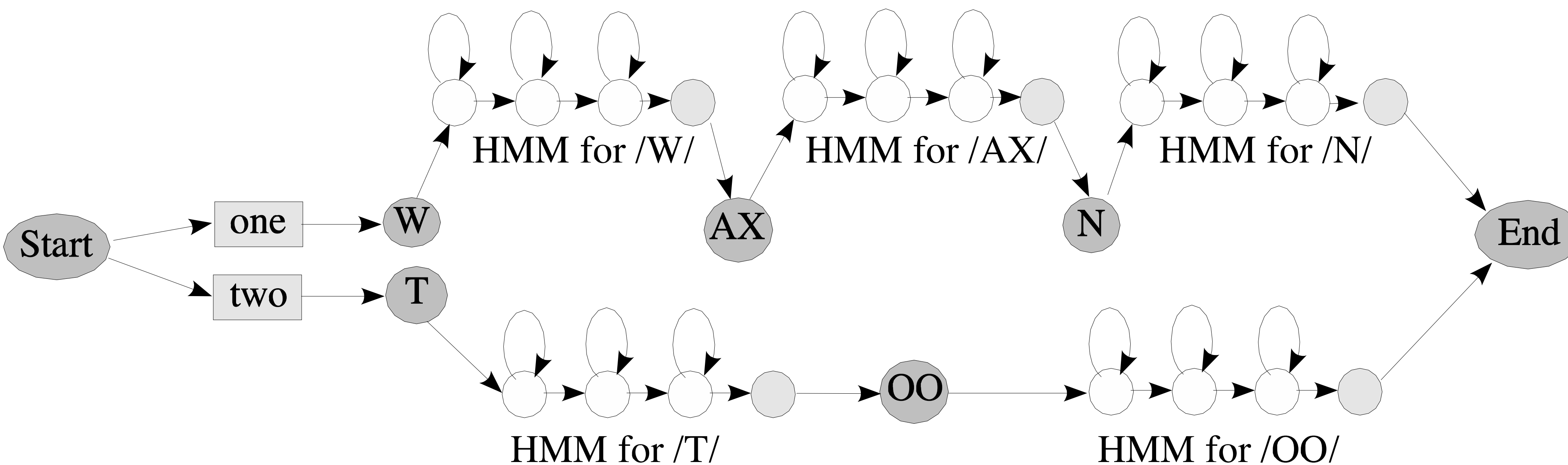
Reprise: Spracherkennung

Token-Pass-Algorithmus

Hidden-Markov Models

- $\hat{W} = \arg\max W : P(O|Ph) \times P(Ph|W) \times P(W)$
- einheitliches Modell für Spracherkennungsvorgang
- **Markov**-Annahme: die Zukunft hängt nur von einer kurzen Vergangenheit ab
  - bzw.: Vergangenheit kann in einen Zustand gepresst werden
  - Observation kann ohne Betrachtung der vollen Historie "verstanden" werden
- wir konstruieren einen Zustandsgraphen in dem jeder Zustand die gesammte (relevante) Historie zusammenfasst

The Search Graph

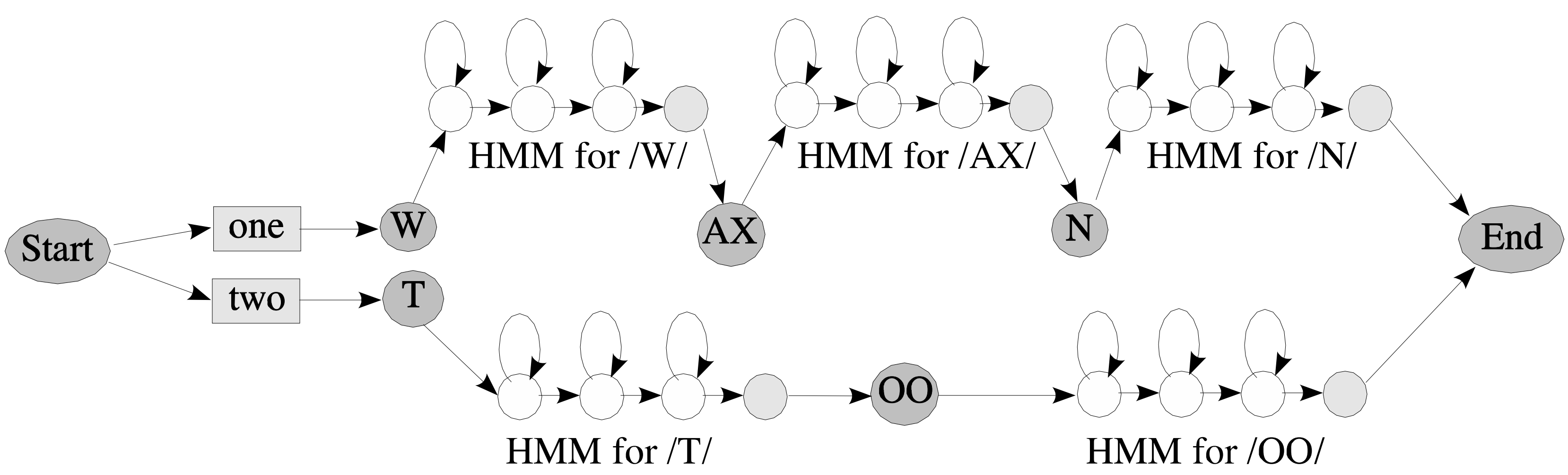


built from language model (here: S→"one"|"two"), lexicon (one→/W AX N/, two→/T OO/), and phone models

aus: Walker et al., Sphinx-4: A Flexible Open Source Framework for SR, 2004.

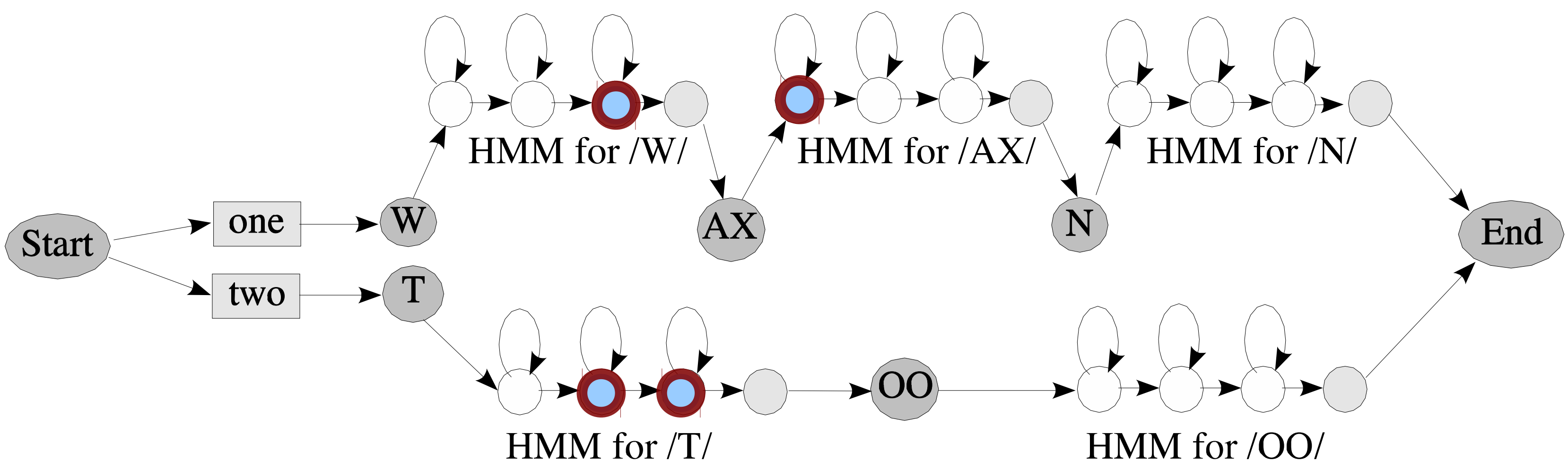
Decoding: Searching for Cheap Paths

- we're looking for the path in the graph that
  - distributes the observations to (emitting) phone states
  - while keeping costs at a minimum
    (identical to the highest probability)



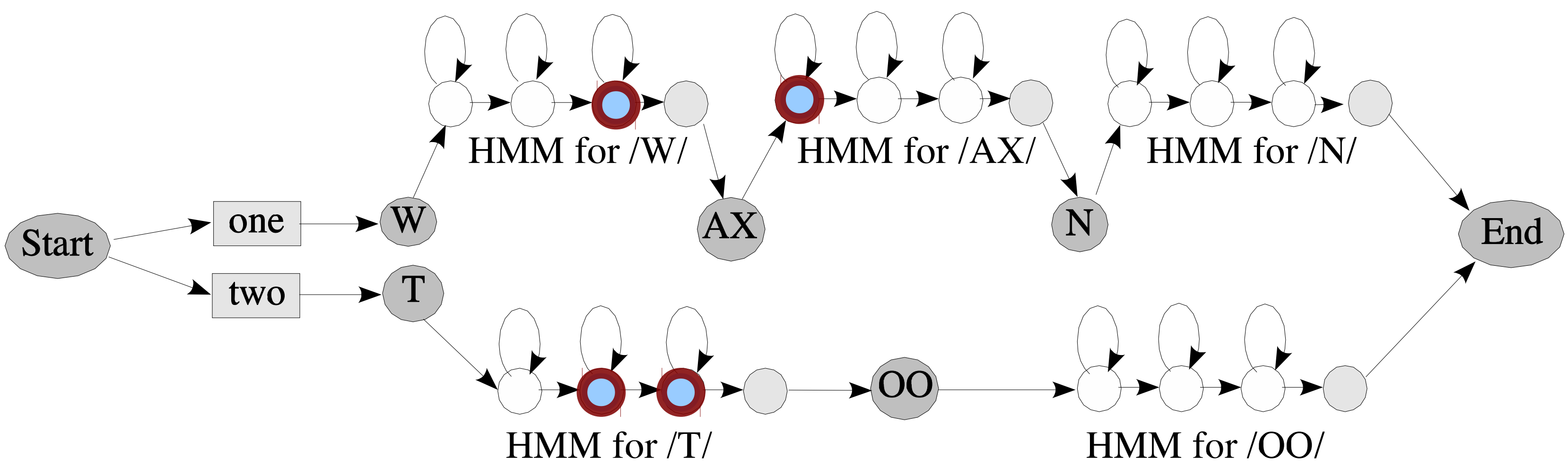
Token-Pass Algorithm: Basic Idea

- time-synchronous search of the observations
  - at every point in time, keep a number of hypotheses, that are represented each by a token
  - generate new tokens from old tokens in every step
  - the winner: best token that reaches the final state in the end



Token-Pass Algorithm: Basic Idea

- every *token*
  - stores the current state in the graph
  - the sum of costs incurred so far
    - possibly differentiated for LM and AM costs
  - details to preceding token (necessary to recover path)

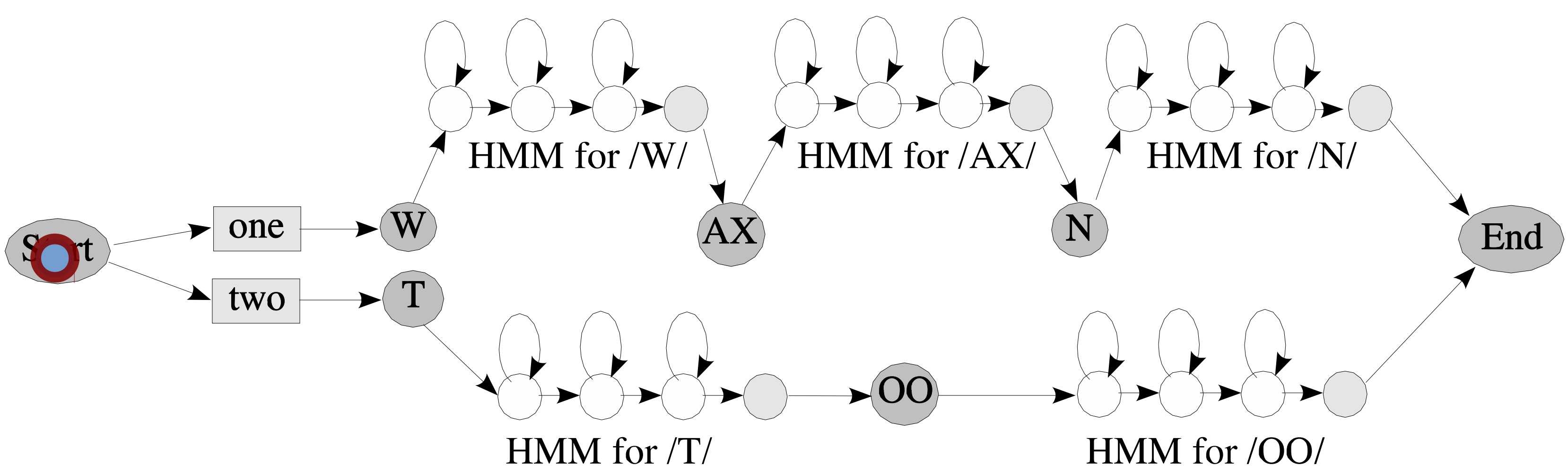


Token-Pass Algorithm en détail

- start with an empty token in the initial state
- for all tokens
  - take the next observation
  - generate all successor tokens from the current state
  - add costs (transition, observation)
  - of all token that are in one state keep only the best token
    - principle of *dynamic programming*: the best path leading here is the only relevant path in the globally best path

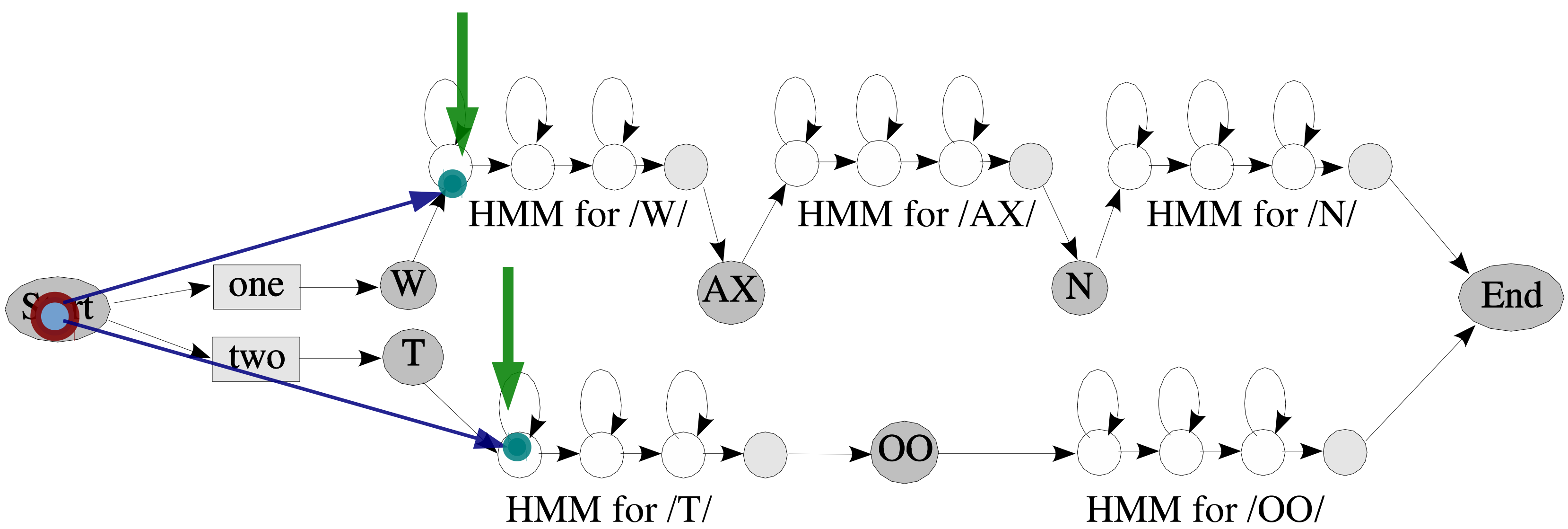
Token-Pass Algorithm

- Initialization: put a token into initial state
- find next tokens (forward to next emitting state)
  - add transition costs for edges
  - add emission/acceptance cost of observation



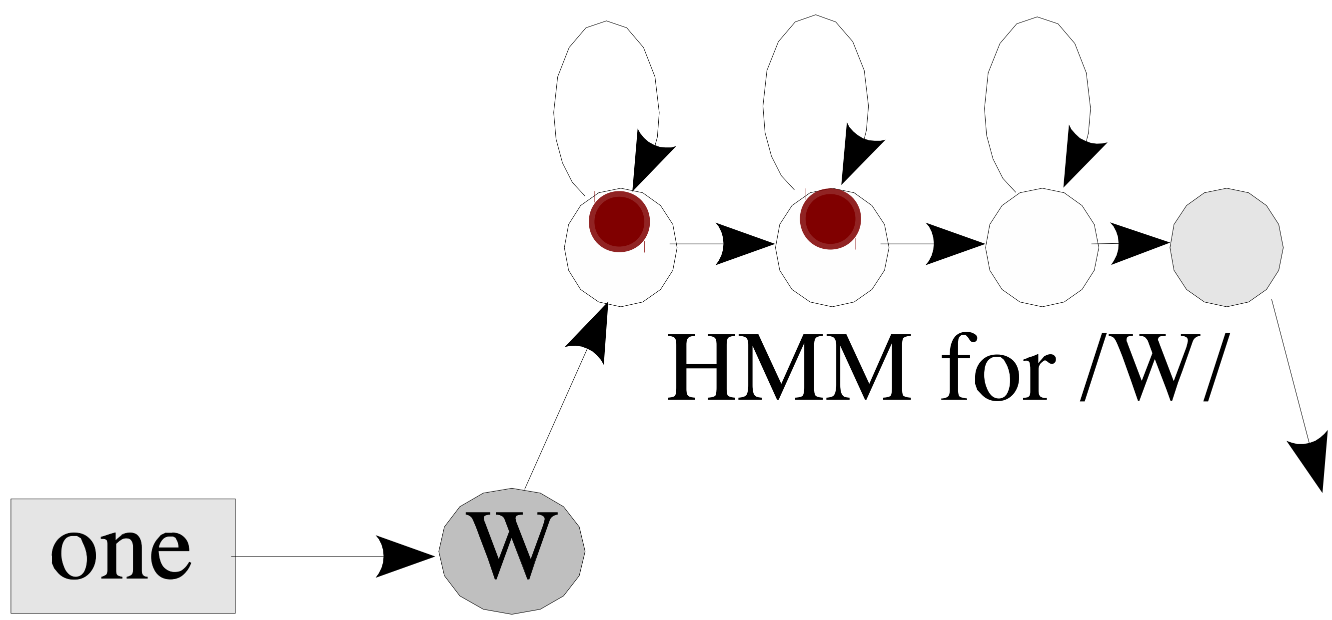
Token-Pass Algorithm

- Initialization: put a token into initial state
- find next tokens (forward to next emitting state)
  - add transition costs for edges
  - add emission/acceptance cost of observation



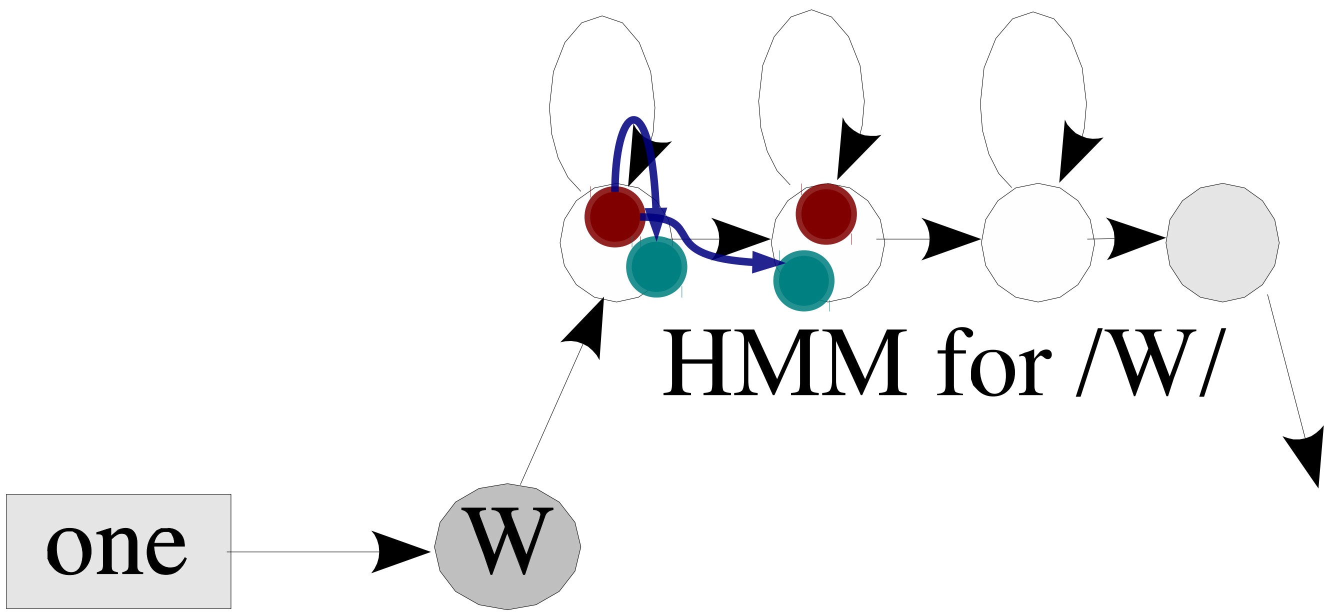
Token-Pass Algorithm: Multiple Tokens in the Same State

- different alignments of observations to one state path
- only the best path needs to be kept
  - all others can't be on the best final path



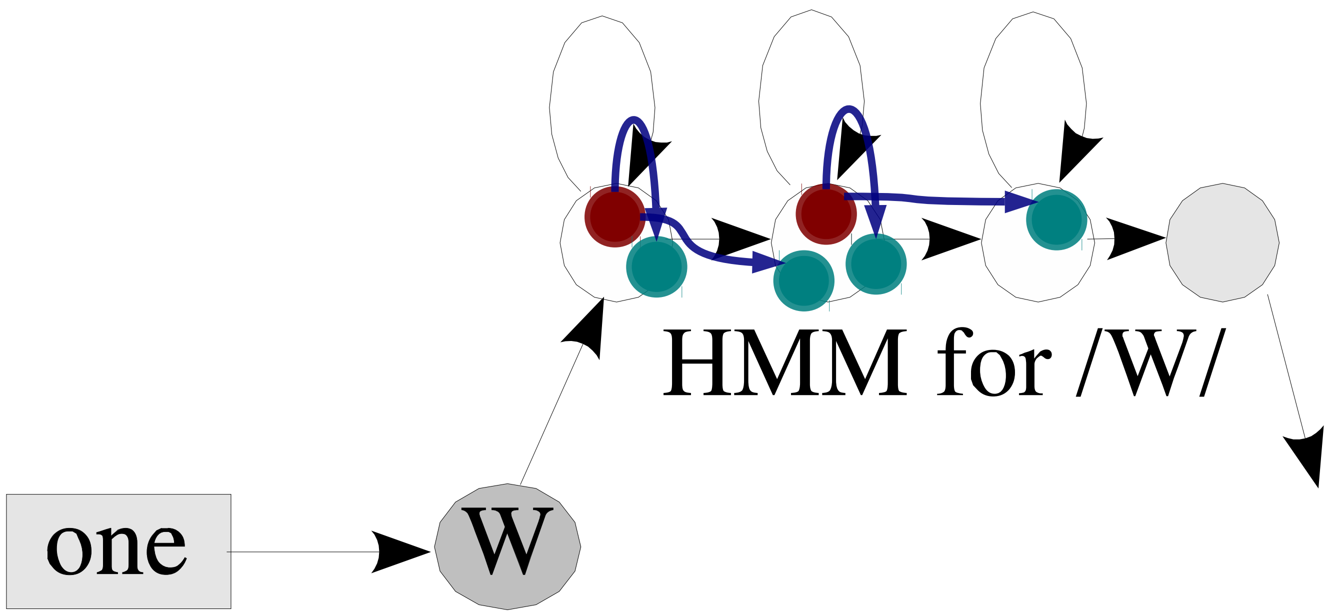
Token-Pass Algorithm: Multiple Tokens in the Same State

- different alignments of observations to one state path
- only the best path needs to be kept
  - all others can't be on the best final path



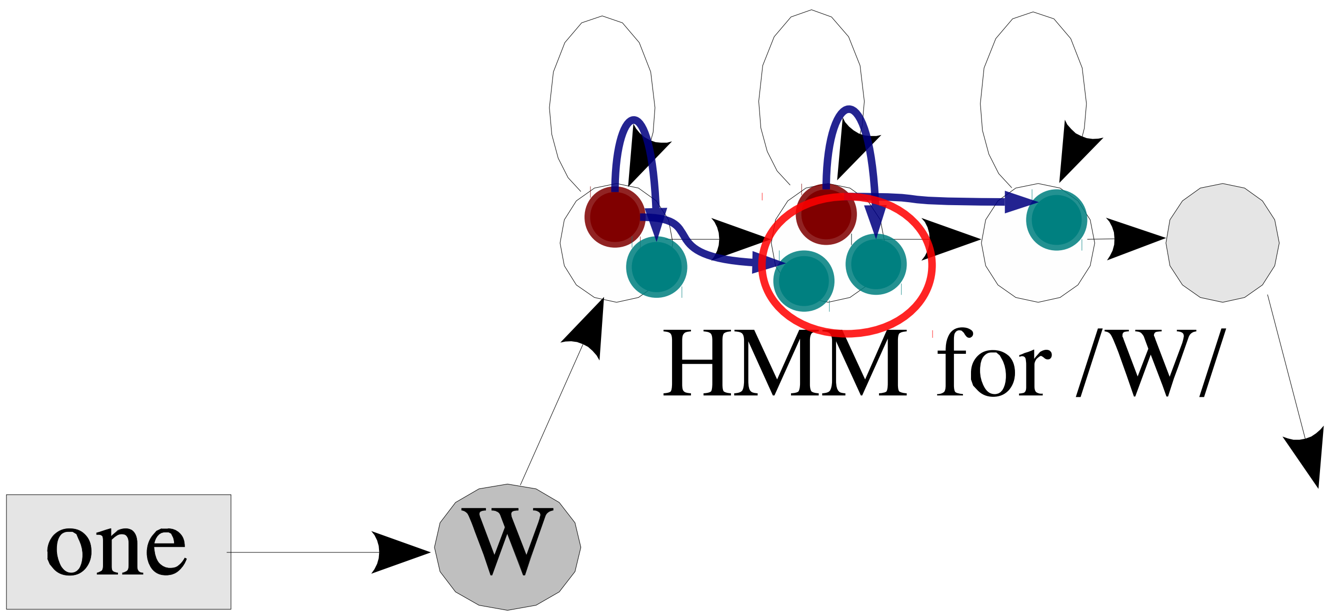
Token-Pass Algorithm: Multiple Tokens in the Same State

- different alignments of observations to one state path
- only the best path needs to be kept
  - all others can't be on the best final path



Token-Pass Algorithm: Multiple Tokens in the Same State

- different alignments of observations to one state path
- only the best path needs to be kept
  - all others can't be on the best final path



Training and decoding optimizes for $P(\mathbf{W}|\mathbf{O})$.
What does this mean?
What could/should be done differently?

$$\hat{\mathbf{W}} = \arg\max \mathbf{W} : P(\mathbf{W}|\mathbf{O})$$

vs.

$$\hat{\mathbf{W}} = \arg\max \mathbf{W} : P(\mathbf{O}|\mathbf{W})P(\mathbf{W})$$

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Confidence estimation

- we don't solve the original question arg max W: P(W|O)
  - hence, we can't use the probability to say how confident we are
  - we do this because P(O) is untractable to compute and we need to use Bayes' rule
- come up with a heuristic to generate a *confidence measure/rejection threshold* (per sentence or better per word)
  - based on search parameters, acoustic parameters, language model probabilities, dialogue state, multi-modal information, confusion matrices, ...
  - highly useful for downstream processing: „Sorry, I am unsure: did you say Dallas Airport or Dulles Airport in DC area?“ more useful than „Sorry, I am unsure, can you repeat please?“ which is more useful than „Ok, I'll look for flights to Dallas.“

Confidence estimation

- not all utterances are equally important
- we do not typically care for how many utterances we get right, but for the proportion of words that we get right
- but not even all words are equally important
- we have large corpora for speech+text, but little interactional data → hard to optimize for specific types of interaction

jetzt aber zum heutigen Thema:

Sprachsynthese

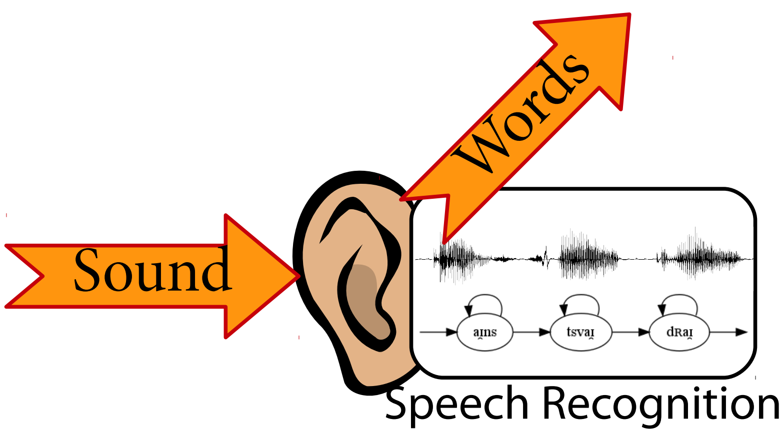
Beispiele

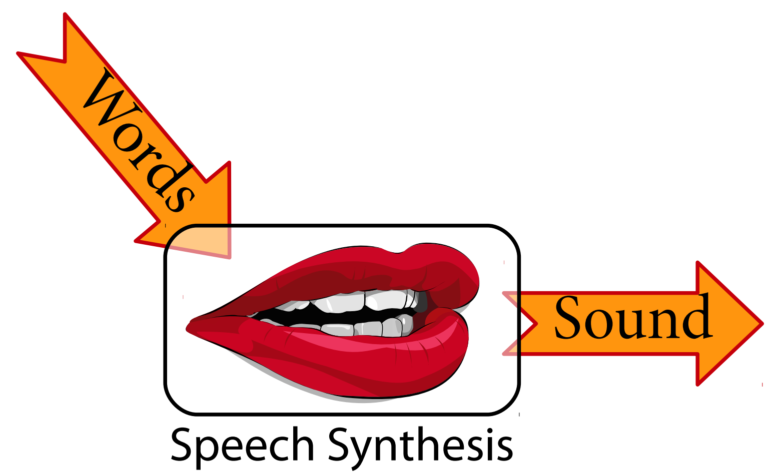
- der erste (digitale) singende Computer (IBM, 1961)
  → hand-optimiertes Vocoding
- aktuelle Implementierung derselben Technik: espeak
  → regel-basiertes Vocoding
- basierend auf Sprachaufnahmen: DreSS-FR, Mbrola
  → Diphon-Synthese
- moderne Variante: MaryTTS
  → generelle konkatenative Synthese (nicht bloß Diphone)
- smartere Version
  → HMM-basierte Synthese (Master-level course ;-)

Input und Output von Sprachdialogsystemen

- Erkennung
  - Reduktion des Signals auf Wörter

  → *Abstrahieren* der Details

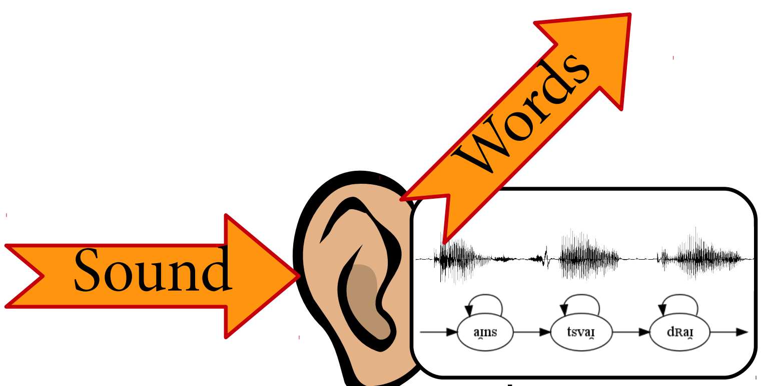




Input und Output von Sprachdialogsystemen

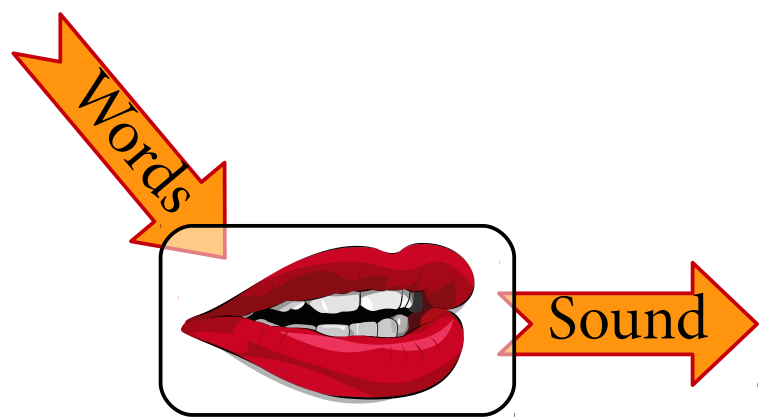
- Erkennung
  - Reduktion des Signals auf Wörter

➔ *Abstrahieren* der Details



Speech Recognition

- Synthese
  - Wörter allein beschreiben das Signal nur ungenügend
  - Natürlichkeit *entsteht* aus den Details



Speech Synthesis

Was *fehlt* der Schriftsprache?

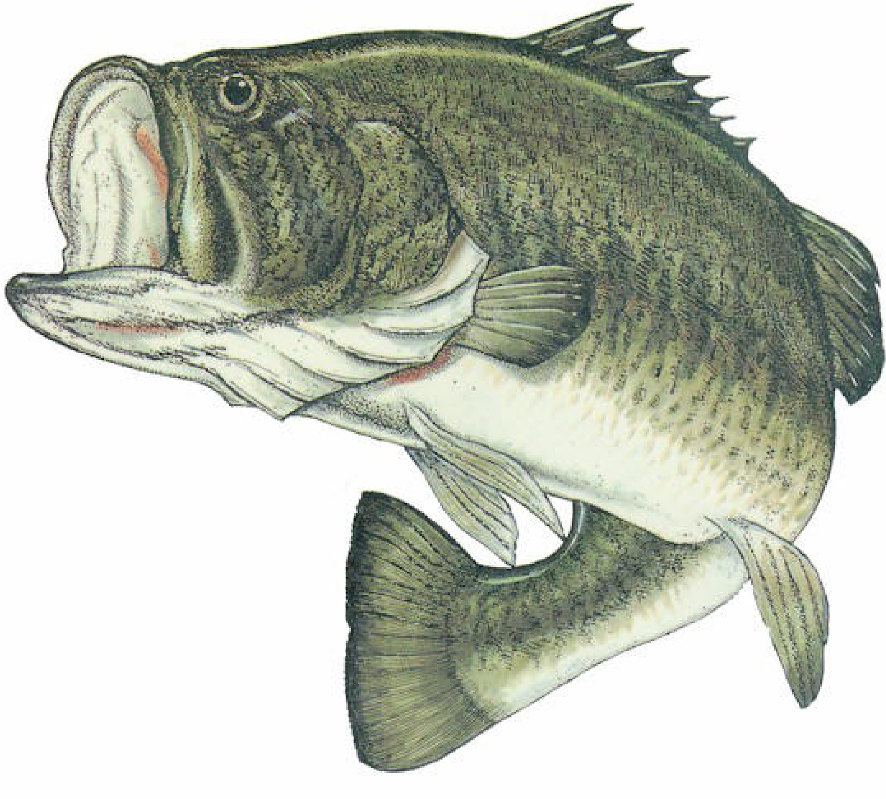
Written vs. Spoken Language
Timo's list

- Abkürzungen, Daten, Zahlen, Währungen, ...
- Homographe: Bass
- Text hat weder Rhythmus noch Melodie!
  - Prosodie ist hochrelevant um Bedeutung auszudrücken
  - Interpunktion löst das Problem nur teilweise.

Homographe

[baɪs] [bæs]

Bass



Informationsstruktur

Information Structure

The linguistic means of structuring information, in order to optimize information transfer within discourse

- Topic / Focus
- Given / New information

- not directly conveyed in textual representation
  - but to a certain degree by prosody
- to reconstruct the structure, listeners also use
  - context of the utterance in the whole conversation
  - world knowledge

Focus and Accentuation

Focus and Accentuation

- “I didn't say we should kill him.”
  - someone else said we should kill him
  - I am denying that I said we should kill him
  - I wrote it down or implied it, but I didn't say it
  - I said someone else should do the job
  - I said that we absolutely must kill him
  - getting him a little nervous would have been enough
  - we got the wrong guy

Focus and Accentuation

- “I didn't say we should kill him.”
  - someone else said we should kill him
  - I am denying that I said we should kill him
  - I wrote it down or implied it, but I didn't say it
  - I said someone else should do the job
  - I said that we absolutely must kill him
  - getting him a little nervous would have been enough
  - we got the wrong guy

Focus and Accentuation

- "I didn't say we should kill him."
  - someone else said we should kill him
  - I am denying that I said we should kill him
  - I wrote it down or implied it, but I didn't say it
  - I said someone else should do the job
  - I said that we absolutely must kill him
  - getting him a little nervous would have been enough
  - we got the wrong guy

Focus and Accentuation

- “I didn't say we should kill him.”
  - someone else said we should kill him
  - I am denying that I said we should kill him
  - I wrote it down or implied it, but I didn't say it
  - I said someone else should do the job
  - I said that we absolutely must kill him
  - getting him a little nervous would have been enough
  - we got the wrong guy

Focus and Accentuation

- “I didn't say we should kill him.”
  - someone else said we should kill him
  - I am denying that I said we should kill him
  - I wrote it down or implied it, but I didn't say it
  - I said someone else should do the job
  - I said that we absolutely must kill him
  - getting him a little nervous would have been enough
  - we got the wrong guy

Focus and Accentuation

- “I didn't say we should kill him.”
  - someone else said we should kill him
  - I am denying that I said we should kill him
  - I wrote it down or implied it, but I didn't say it
  - I said someone else should do the job
  - I said that we absolutely must kill him
  - getting him a little nervous would have been enough
  - we got the wrong guy

Focus and Accentuation

- “I didn't say we should kill him.”
  - someone else said we should kill him
  - I am denying that I said we should kill him
  - I wrote it down or implied it, but I didn't say it
  - I said someone else should do the job
  - I said that we absolutely must kill him
  - getting him a little nervous would have been enough
  - we got the wrong guy

Information Structure

- information structure is an active area of research:
  - unknown how exactly to represent IS (cross-linguistically, cross-genre, in dialogue, …)
  - unknown how (exactly) IS influences speech

- problem of premature implementation:

can we really expect a computer to successfully perform speech synthesis even before the basic research has been done?

Prosody

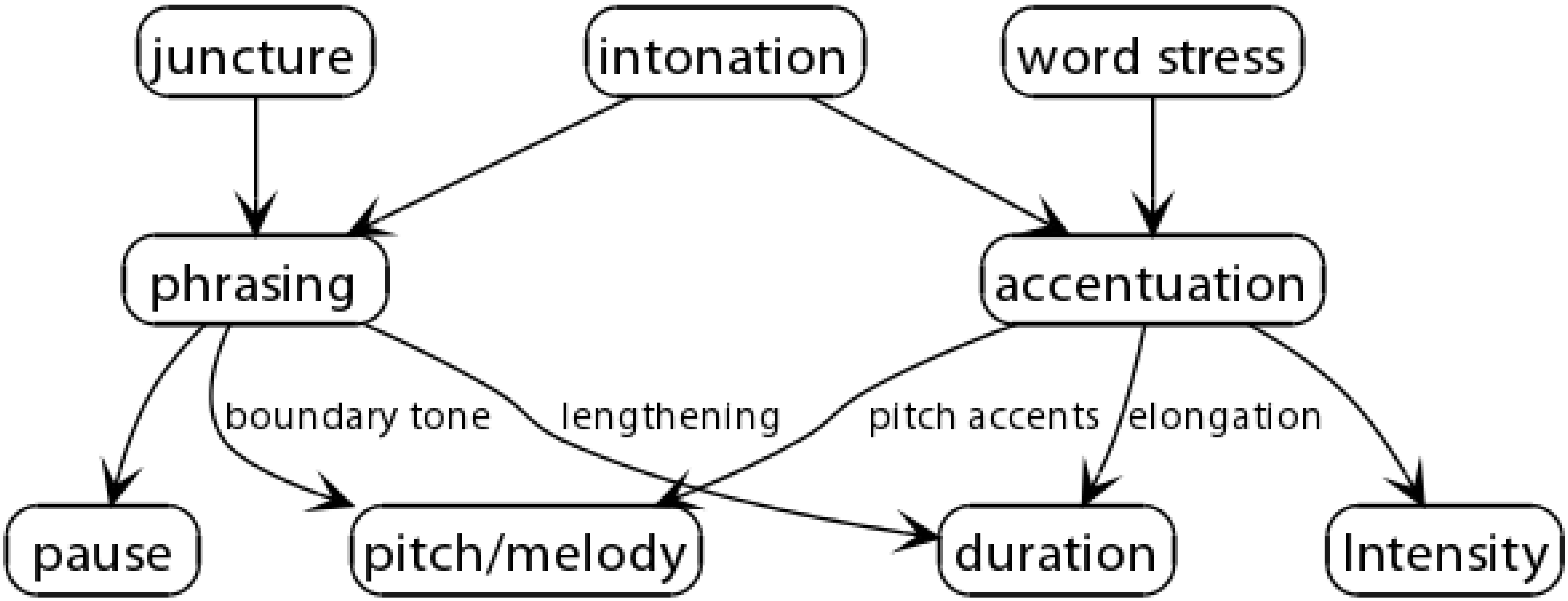
supra-segmental properties of speech

- phenomena:
  - pitch (i.e., melody / fundamental frequency)
  - loudness / intensity
  - duration, pauses

- phonetically: accentuation and phrasing

- phonologically: (word)stress, intonation, juncture

Prosody:
Phonology – Phonetics – Phenomena



What a computer *can* do

- problems that are well understood:
  - find solutions based on a model
  - use lists of exceptions if model is faulty
- problems that are somewhat understood:
  - use heuristics to get details right
  - try to avoid taking a stand
- problems that aren't yet understood:
  - require additional instructions in the input
  - guess

What a computer *can* do: focus

- human listeners are predictive (and forgiving):
  - it's worse to be very wrong occasionally
    than to say everything a little bit wrongly
  - human listeners will select the correct interpretation
    (using *their* world knowledge) from available options
- solution:
  - put a small accentuation on all possible focus points
- however
  - system does not *take a stand*, it sounds indifferent, bored

Process diagram of Speech Synthesis

input

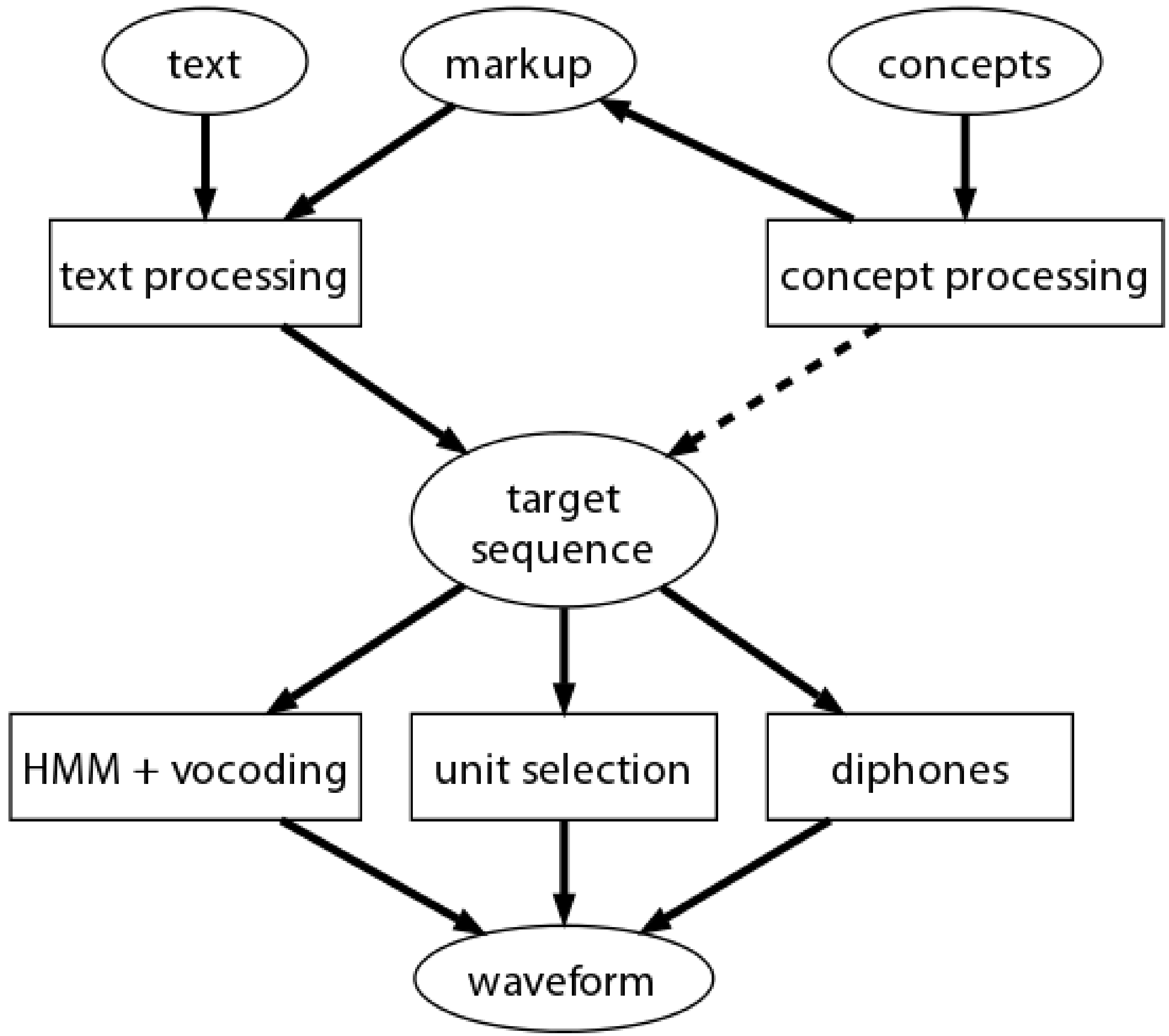
✓ symbolic processing

problem: text is missing detail

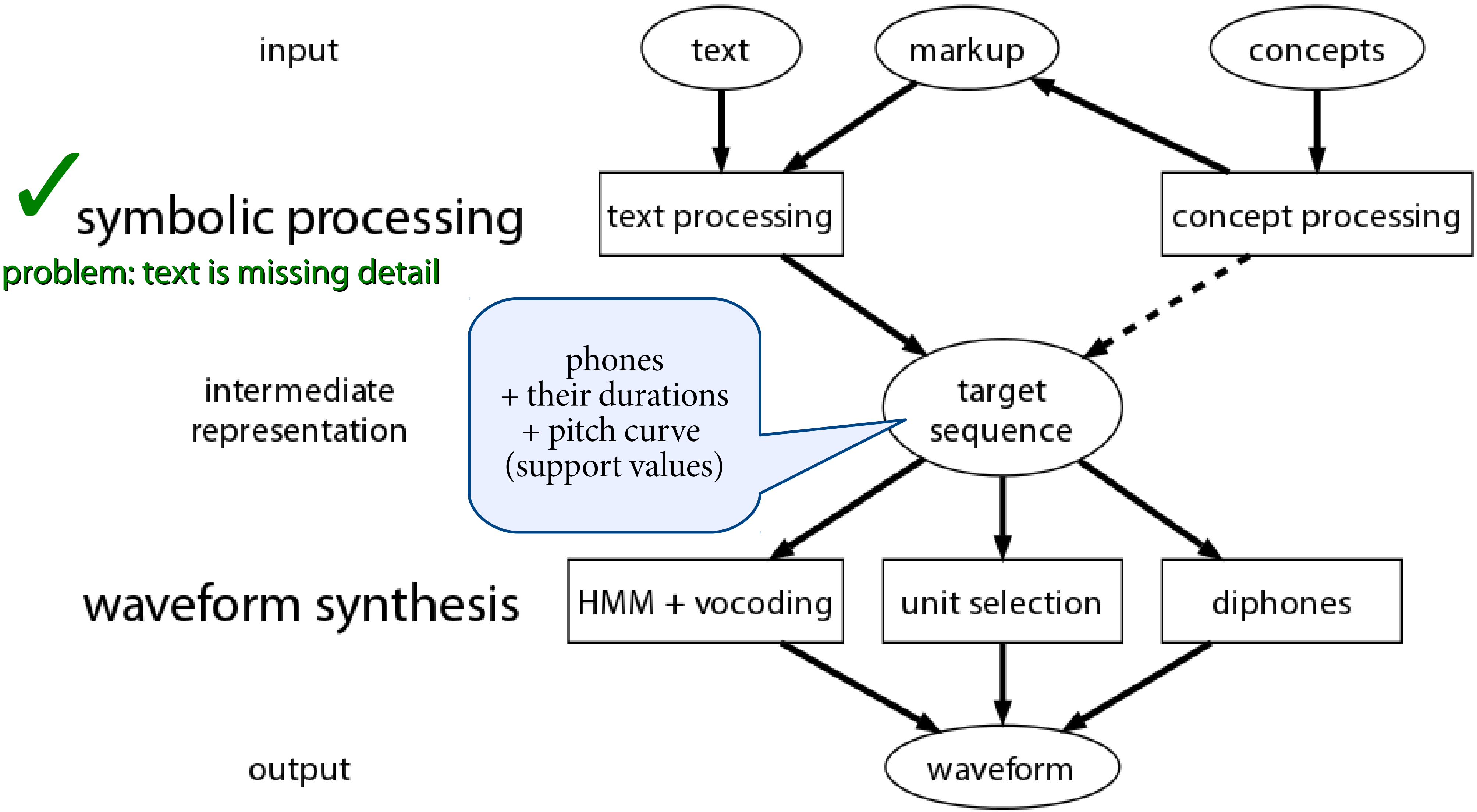
intermediate
representation

waveform synthesis

output



Process diagram of Speech Synthesis



waveform synthesis

Waveform Synthesis

from the target sequence (phones+duration+pitch)

1. formant-based:

   rules to determine target formants and other parts of the signal
   rules to determine transitions

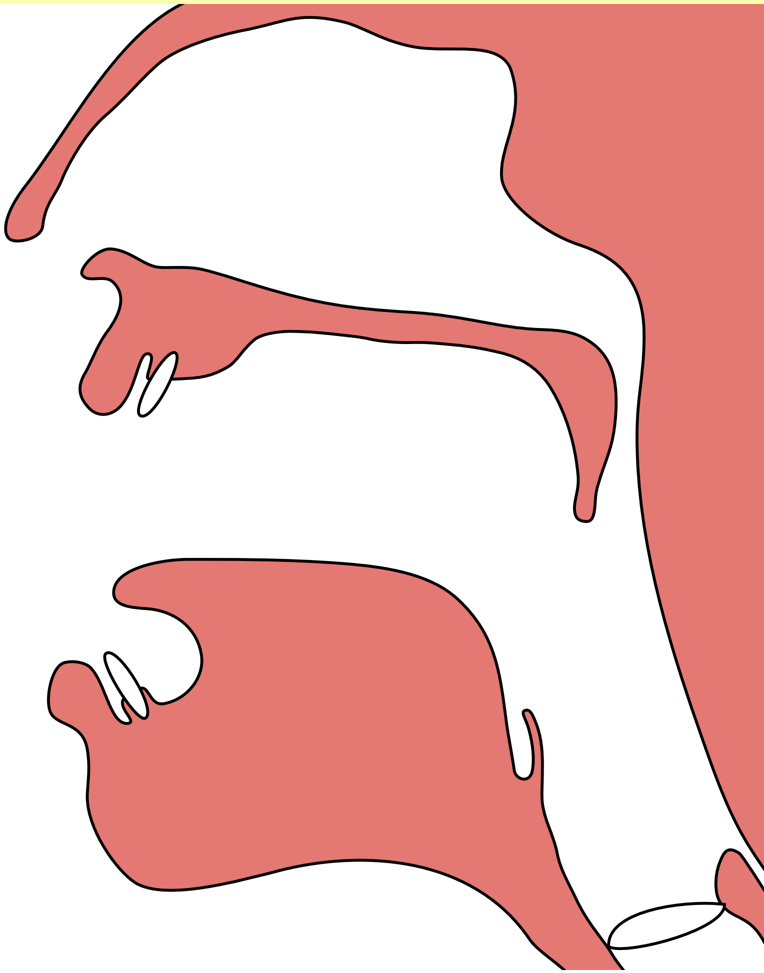
2. pattern-based:

   database of many short speech segments
   segments are concatenated one after the other

3. model-based approach in 2 weeks

Speech Production: Source-Filter Model

- glottal folds produce primary signal
- vocal tract acts as a filter



Diphone Synthesis

- Concatenation of short speech snippets
- units from center of a phone to center of the next: _h+ha:+a:l+lo:+o:_+_v+vi:+i:g+ge:+e:t+ts+s_
  - concatenation within "stable" phase of the phone
  - coarticulation is (largely) covered
- 40 phones → ~1600 diphones!
  - recorded from one speaker → one voice
  - additional signal processing for duration+pitch change

General Concatenative Synthesis

- alternatives for the mapping target → speech snippets
  - more speech material in database
  - selection of material that better fits the target sequence
- selection becomes a search of best concatenation
  - costs of fit of concatenation between snippets
  - costs of fit of snippets to target sequence
- computationally expensive (search)
  - very high memory demands (500MB+ per voice)
- results can be very natural sounding

what do you *like* better:
formant-based or pattern-based synthesis? why?

Vielen Dank.

baumann@informatik.uni-hamburg.de

Notizen

- wieder viel zu viel Material, aber was soll's :-)
- Beispielsysteme angehört, yay.
- Details zu Informationsstruktur ausgelassen, aber Beispiel (I didn't kill him) durchgenudelt. Quintessenz: wir haben ein premature-implementation-Problem.

Further Reading

- Speech Synthesis in General:
  - P. Taylor (2009): *Text-to-Speech Synthesis*. Cambridge Univ Press. ISBN: 978-0521899277. InfBib: A TAY 43070 (accessible introduction to the topic)
  - Rabiner & Juang (1993): *Fundamentals of Speech Recognition*. Prentice Hall. Stabi: A 1994/994. (in-depth mathematical approach)
  - Dong Yu, Li Deng (2015): *Automatic Speech Recognition: A Deep Learning Approach*. Springer. InfBib: A AUT 51465 (NN-based methods)
- The MaryTTS Speech Synthesis System:
  - Schröder & Trouvain (2003): "The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching", *Int. J. of Speech Technology* **6**(3).

Desired Learning Outcomes

- Ziel der Sprachsynthese ist es, die natürliche Varianz von Sprache zu erzeugen
  - dies ist das Gegenteil vom Ziel der Spracherkennung, die versucht Varianz aufzulösen!
- Probleme/Ambiguitäten linguistischer Vorverarbeitung:
  - Aussprachevarianten
  - Prosodie und Informationsstruktur sowie Emotionalität
  - Synthesetechniken: Formant- und Diphonsynthese