Automatic Speech Recognition (CS753)

Lecture 18: Search & Decoding (Part I)

Instructor: Preethi Jyothi

Mar 23, 2017

Recall ASR Decoding

$$W^* = \arg\max_{W} \Pr(O_A|W)\Pr(W)$$

$$W^* = \arg\max_{w_1^N, N} \left\{ \left[\prod_{n=1}^{N} \Pr(w_n|w_{n-m+1}^{n-1}) \right] \left[\sum_{q_1^T, w_1^N} \prod_{t=1}^{T} \Pr(O_t|q_t, w_1^N) \Pr(q_t|q_{t-1}, w_1^N) \right] \right\}$$

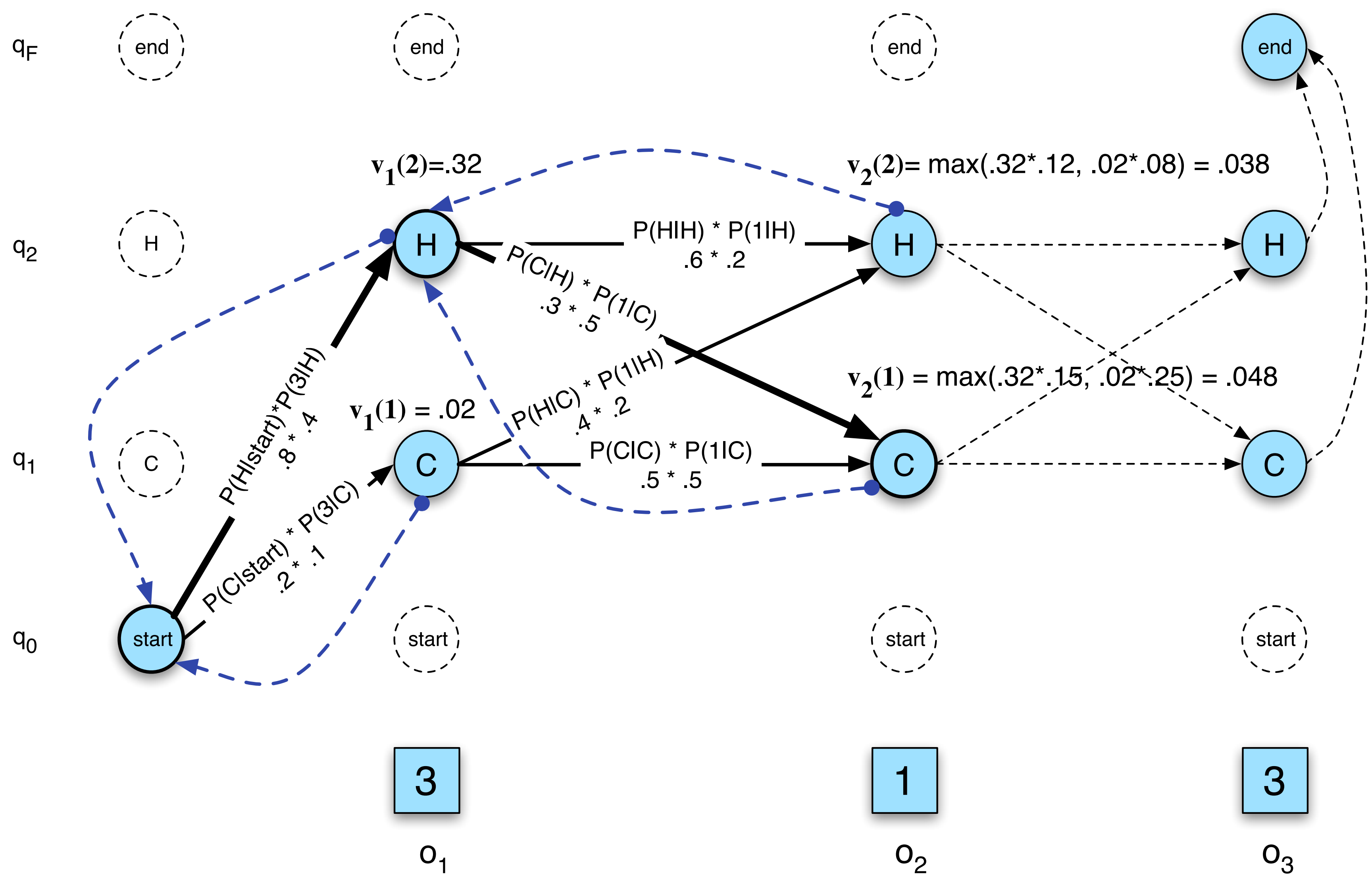
Viterbi

$$\approx \arg\max_{w_1^N, N} \left\{ \left[\prod_{n=1}^{N} \Pr(w_n|w_{n-m+1}^{n-1}) \right] \left[\max_{q_1^T, w_1^N} \prod_{t=1}^{T} \Pr(O_t|q_t, w_1^N) \Pr(q_t|q_{t-1}, w_1^N) \right] \right\}$$

- Viterbi approximation divides the above optimisation problem into sub-problems that allows the efficient application of dynamic programming
- An exact search using Viterbi is infeasible for large vocabulary tasks!

Recall Viterbi search

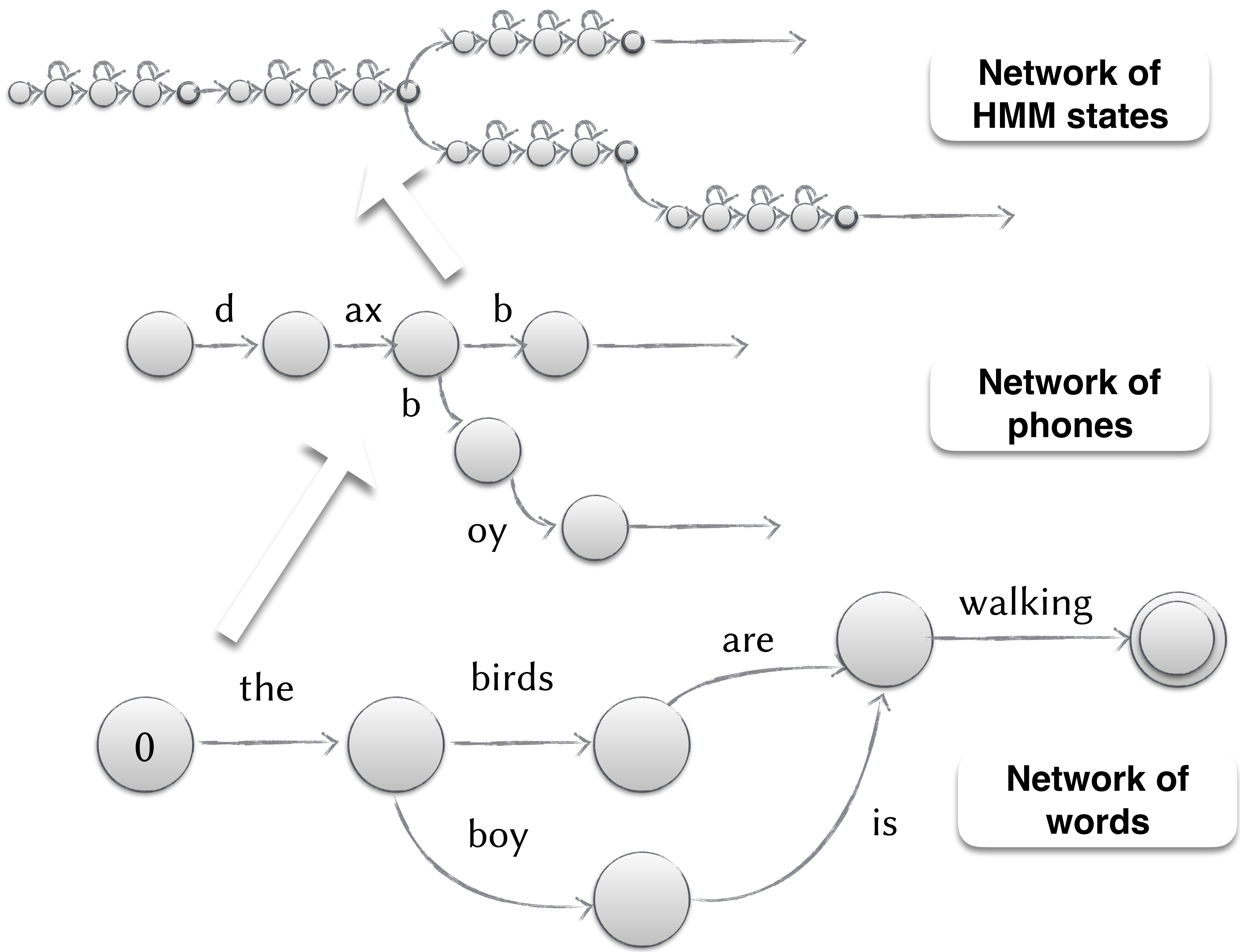
- Viterbi search finds the most probable path through a trellis of time on the X-axis and states on the Y-axis



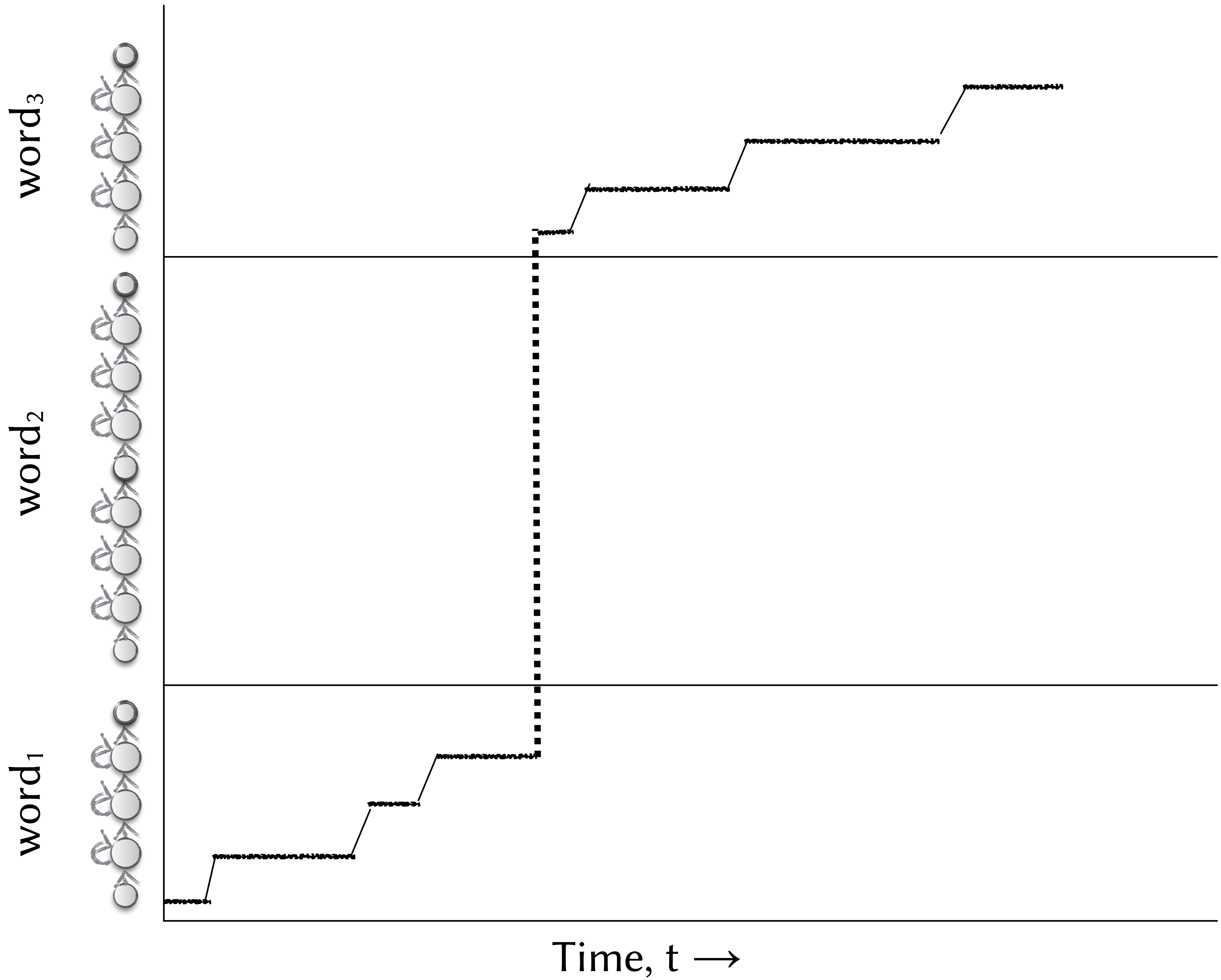
- Viterbi algorithm: Only needs to maintain information about the most probable path at each state

Image from [JM]: Jurafsky & Martin, 3rd edition, Chapter 9

ASR Search Network



Time-state trellis



Viterbi search over the large trellis

- Exact search is infeasible for large vocabulary tasks
  - Unknown word boundaries
  - Ngram language models greatly increase the search space
- Solutions
  - Compactly represent the search space using WFST-based optimisations
  - Beam search: Prune away parts of the search space that aren't promising

Viterbi search over the large trellis

- Exact search is infeasible for large vocabulary tasks
  - Unknown word boundaries
  - Ngram language models greatly increase the search space
- Solutions
  - Compactly represent the search space using WFST-based optimisations
  - Beam search: Prune away parts of the search space that aren't promising

Two main WFST Optimizations

- Use determinization to reduce/eliminate redundancy

Recall not all weighted transducers are determinizable

To ensure determinizability of $L \circ G$, introduce disambiguation symbols in L to deal with homophones in the lexicon

read : r eh d #0
red : r eh d #1

Propagate the disambiguation symbols as self-loops back to C and H. Resulting machines are $\tilde{H}$, $\tilde{C}$, $\tilde{L}$

Two main WFST Optimizations

- Use determinization to reduce/eliminate redundancy
- Use minimization to reduce space requirements

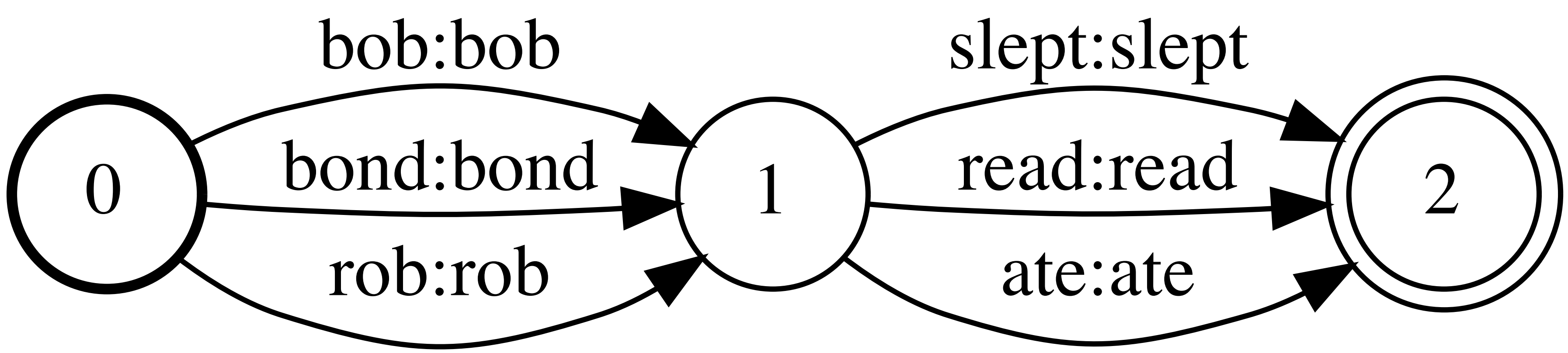
Minimization ensures that the final composed machine has minimum number of states

Final optimization cascade:

$$N = \pi_\varepsilon(\min(\det(\tilde{H} \circ \det(\tilde{C} \circ \det(\tilde{L} \circ G)))))$$

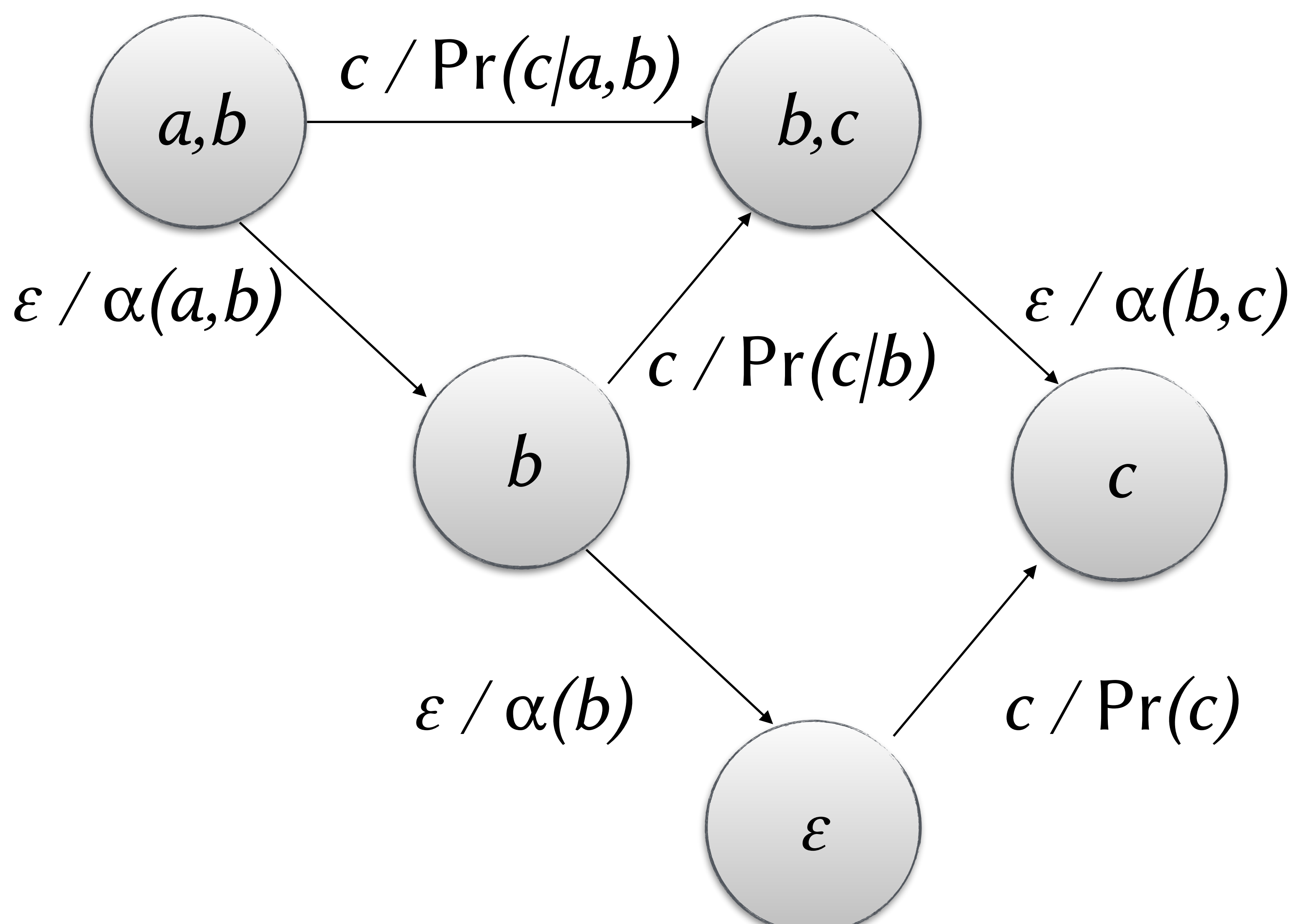
Replaces disambiguation symbols in input alphabet of $\tilde{H}$ with ε

Example G

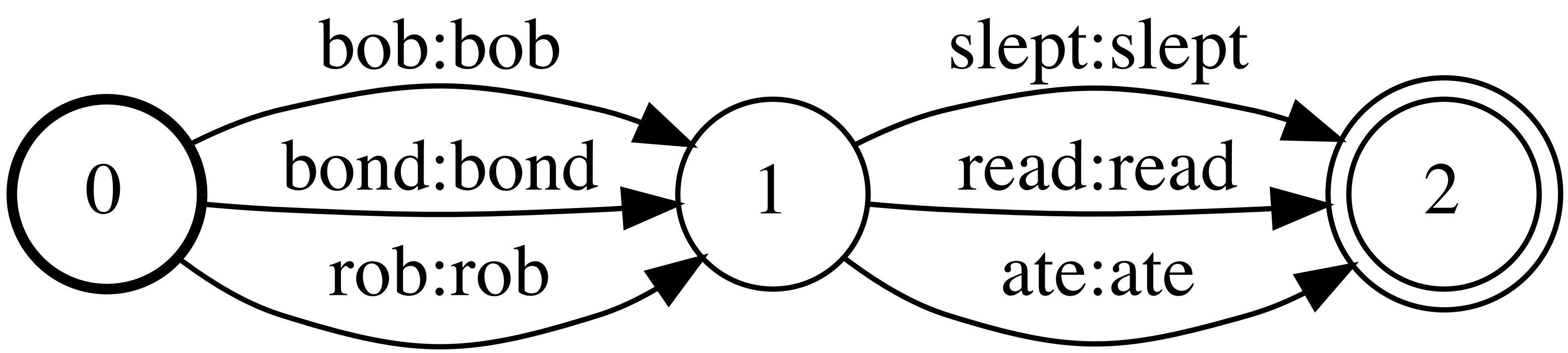


Compact language models (G)

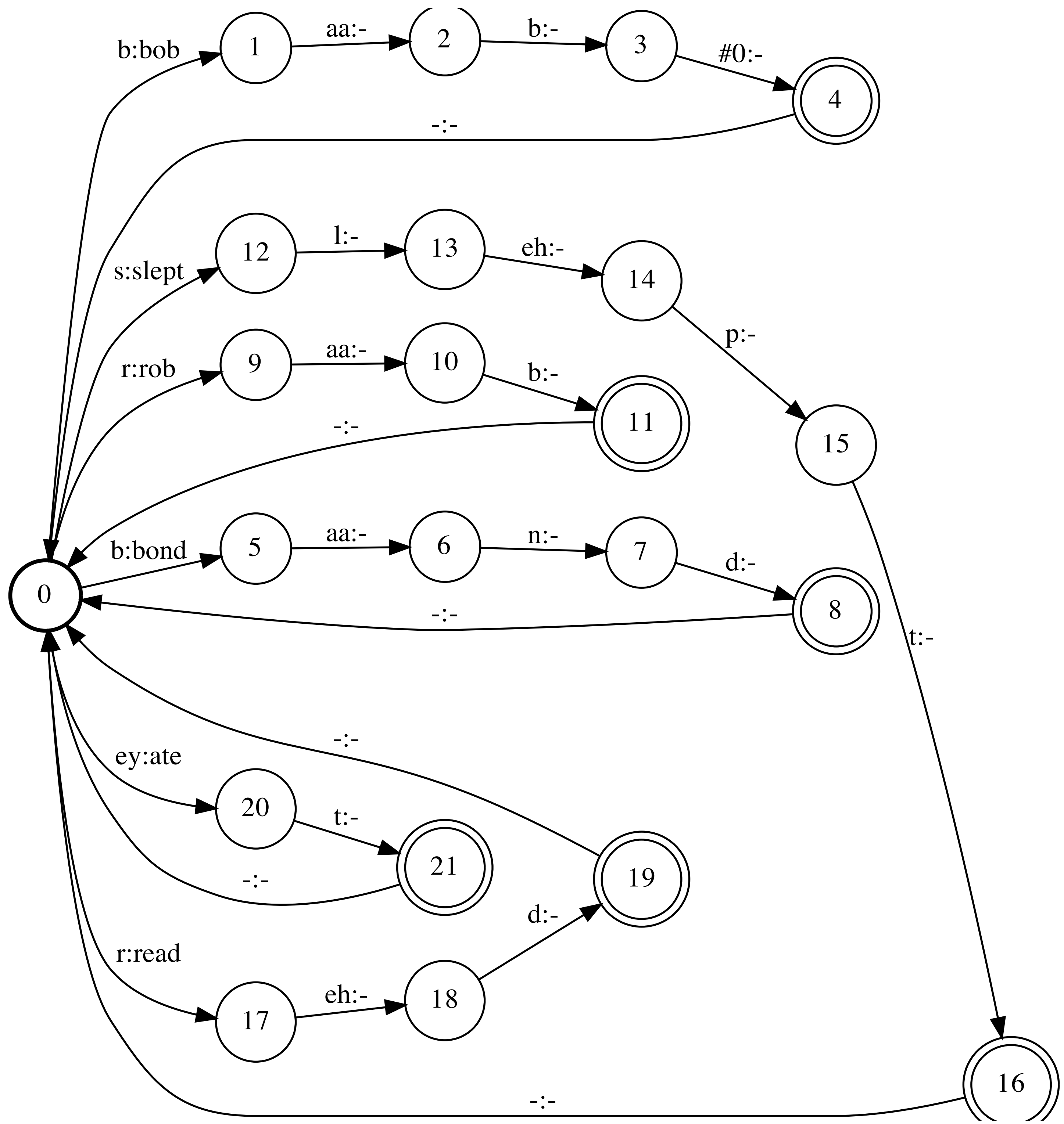
- Use Backoff Ngram language models for G



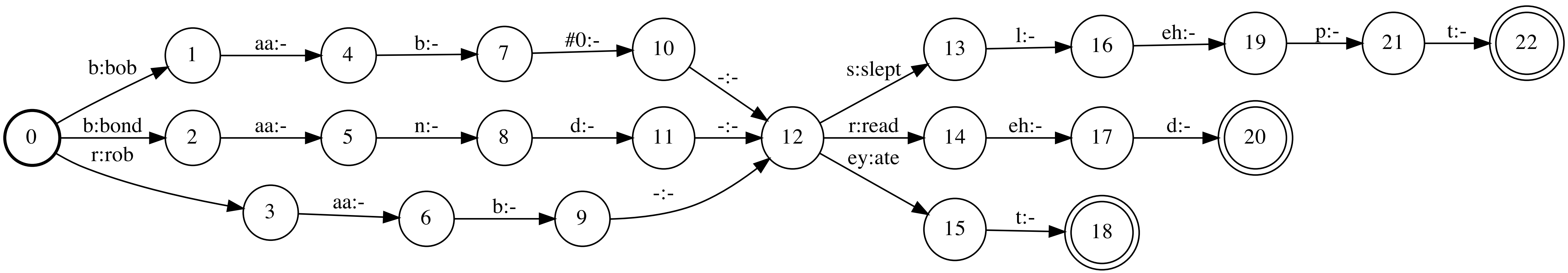
Example G



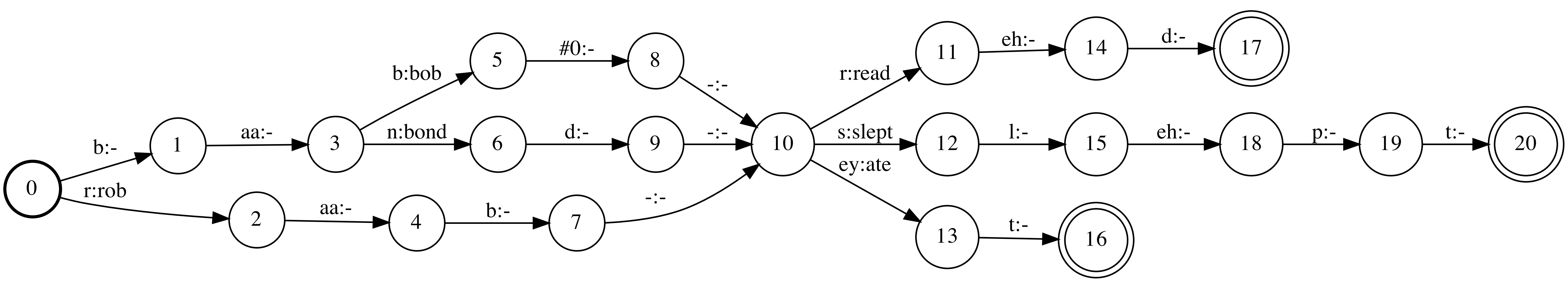
Example L̃ :Lexicon with disambig symbols



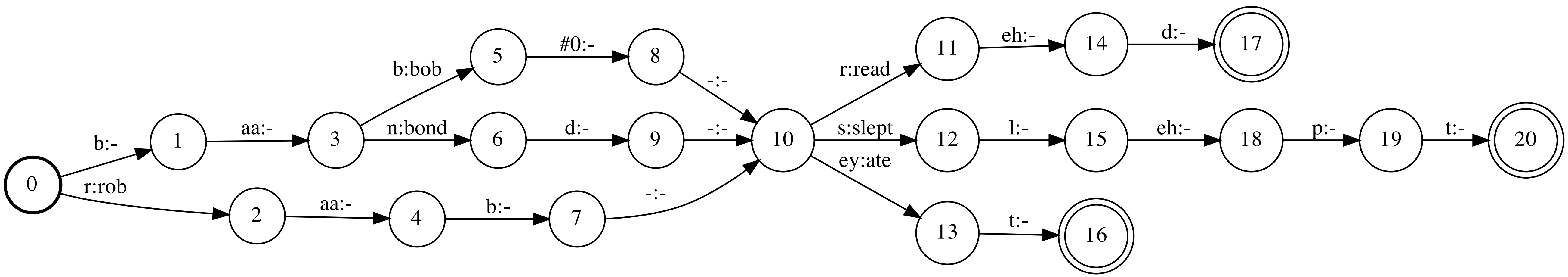
$\tilde{L} \circ G$



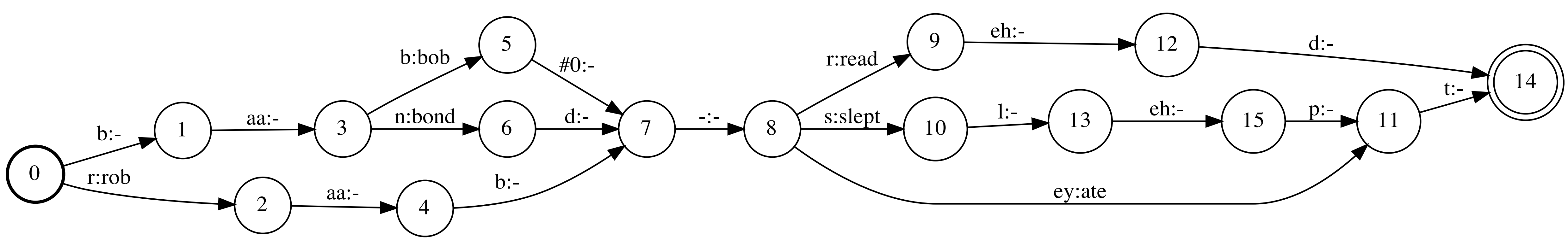
$\det(\tilde{L} \circ G)$



det($\tilde{L} \circ G$)



min(det($\tilde{L} \circ G$))



Viterbi search over the large trellis

- Exact search is infeasible for large vocabulary tasks
  - Unknown word boundaries
  - Ngram language models greatly increase the search space
- Solutions
  - Compactly represent the search space using WFST-based optimisations
  - Beam search: Prune away parts of the search space that aren't promising

Beam pruning

- At each time-step t, only retain those nodes in the time-state trellis that are within a fixed threshold δ (beam width) of the best path

- Given active nodes from the last time-step:

  - Examine nodes in the current time-step …

  - … that are reachable from active nodes in the previous time-step

  - Get active nodes for the current time-step by only retaining nodes with hypotheses that score close to the score of the best hypothesis

Beam search

- Beam search at each node keeps only hypotheses with scores that fall within a threshold of the current best hypothesis

- Hypotheses with $Q(t, s) < \delta \cdot \max Q(t, s')$ are pruned

  here, δ controls the *beam width*

- Search errors could occur if the most probable hypothesis gets pruned

- Trade-off between balancing search errors and speeding up decoding

Static and dynamic networks

- What we've seen so far: *Static* decoding graph
  - $H \circ C \circ L \circ G$
  - Determinize/minimize to make this graph more compact
- Another approach: *Dynamic* graph expansion
  - Dynamically build the graph with active states on the fly
  - Do on-the-fly composition with the language model G
    - $(H \circ C \circ L) \circ G$

Multi-pass search

- Some models are too expensive to implement in first-pass decoding (e.g. RNN-based LMs)
- First-pass decoding: Use simpler model (e.g. Ngram LMs)
  - to find most probable word sequences
  - and represent as a word lattice or an N-best list
- Rescore first-pass hypotheses using complex model to find the best word sequence

Multi-pass decoding with N-best lists

- Simple algorithm: Modify the Viterbi algorithm to return the N-best word sequences for a given speech input

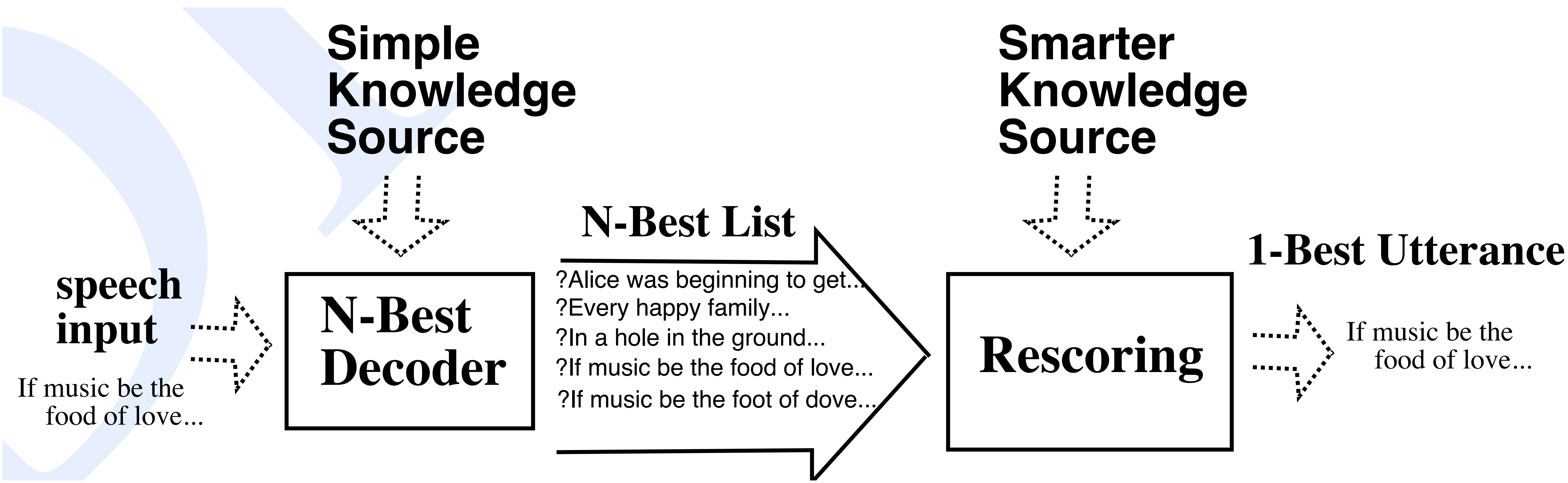


Image from [JM]: Jurafsky & Martin, SLP 2nd edition, Chapter 10

Multi-pass decoding with N-best lists

- Simple algorithm: Modify the Viterbi algorithm to return the N-best word sequences for a given speech input

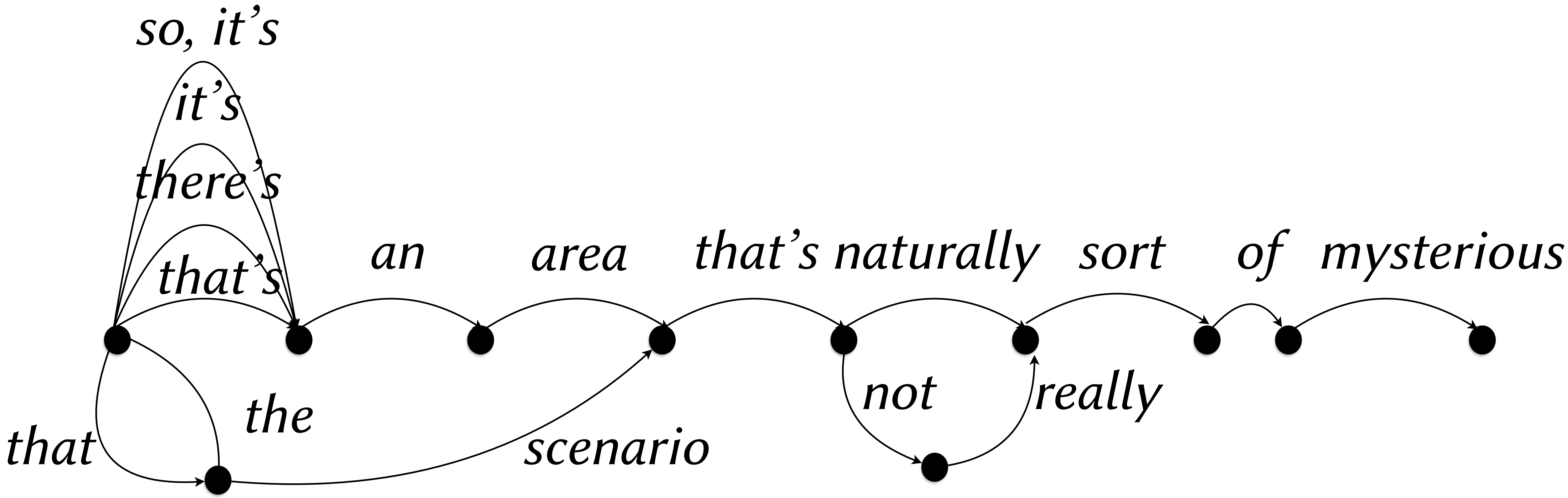
Rank	Path	AM logprob	LM logprob
1.	it's an area that's naturally sort of mysterious	-7193.53	-20.25
2.	that's an area that's naturally sort of mysterious	-7192.28	-21.11
3.	it's an area that's not really sort of mysterious	-7221.68	-18.91
4.	that scenario that's naturally sort of mysterious	-7189.19	-22.08
5.	there's an area that's naturally sort of mysterious	-7198.35	-21.34
6.	that's an area that's not really sort of mysterious	-7220.44	-19.77
7.	the scenario that's naturally sort of mysterious	-7205.42	-21.50
8.	so it's an area that's naturally sort of mysterious	-7195.92	-21.71
9.	that scenario that's not really sort of mysterious	-7217.34	-20.70
10.	there's an area that's not really sort of mysterious	-7226.51	-20.01

- N-best lists aren't as diverse as we'd like. And, not enough information in N-best lists to effectively use other knowledge sources

Image from [JM]: Jurafsky & Martin, SLP 2nd edition, Chapter 10

Multi-pass decoding with lattices

- ASR lattice: Weighted automata/directed graph representing alternate word hypotheses from an ASR system



Multi-pass decoding with lattices

- *Confusion networks/sausages*: Lattices that show competing/ confusable words and can be used to compute posterior probabilities at the word level