Speech Recognition
Lecture 11: Search.

Mehryar Mohri
Courant Institute of Mathematical Sciences
mohri@cims.nyu.edu

Speech Recognition Components

- Acoustic and pronunciation model:

$$\Pr(o \mid w) = \sum_{d,c,p} \Pr(o \mid d) \Pr(d \mid c) \Pr(c \mid p) \Pr(p \mid w).$$

acoustic model

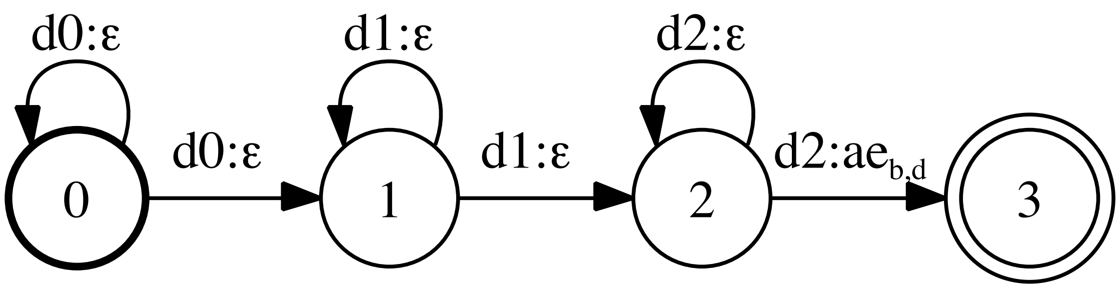
- $\Pr(o \mid d)$: observation seq. $\leftarrow$ distribution seq.
- $\Pr(d \mid c)$: distribution seq. $\leftarrow$ CD phone seq.
- $\Pr(c \mid p)$: CD phone seq. $\leftarrow$ phoneme seq.
- $\Pr(p \mid w)$: phoneme seq. $\leftarrow$ word seq.

- Language model: $\Pr(w)$, distribution over word seq.

Continuous Speech Models

(Rabiner and Juang, 1993)

- Graph topology: 3-state HMM model: for each CD phone $ae_{b,d}$.



  - Interpretation: beginning, middle, and end of CD phone.

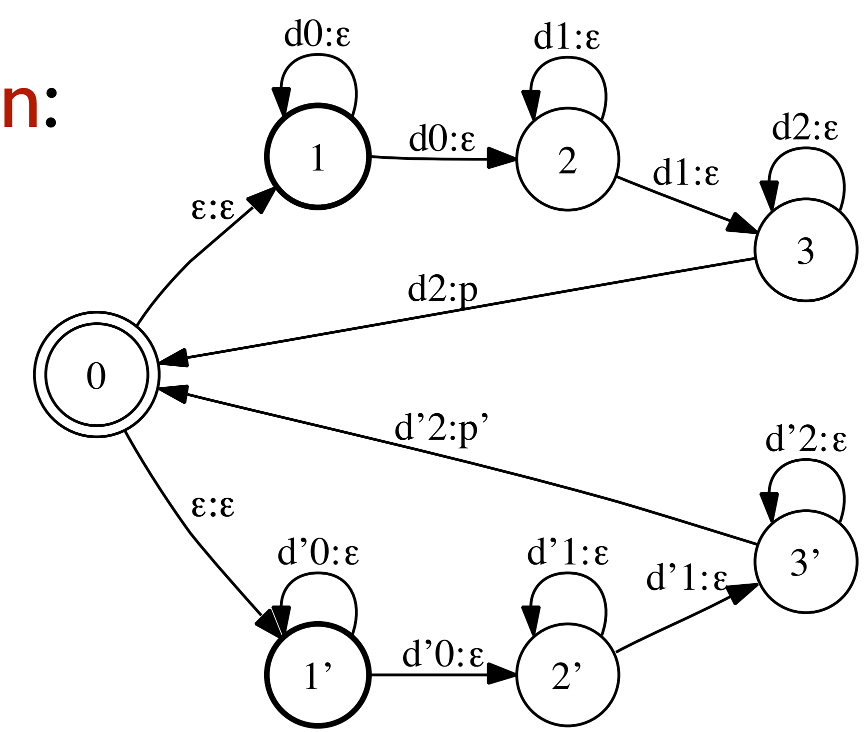
- Continuous case: transition weights based on distributions over feature vectors in $\mathbb{R}^N$, typically with $N = 39$.

HMM model - Representation

- Composite model: obtained by taking the union and closure of all CD phone models.

$$\left(\sum_{p=1}^{P} H_i\right)^*.$$

- Illustration:

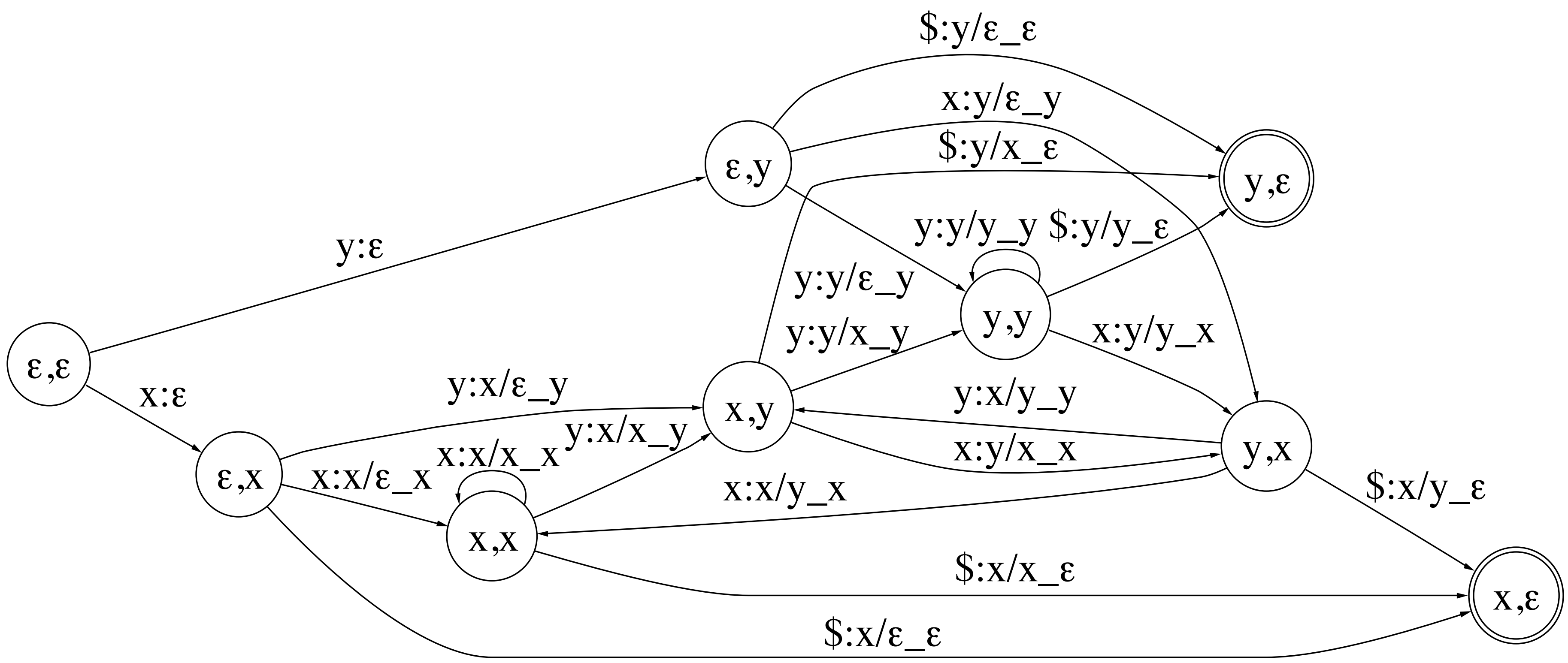


Tying can reduce the size.

CD Model Representation

(MM, Pereira, Riley, 2007)

- Deterministic transducer representation

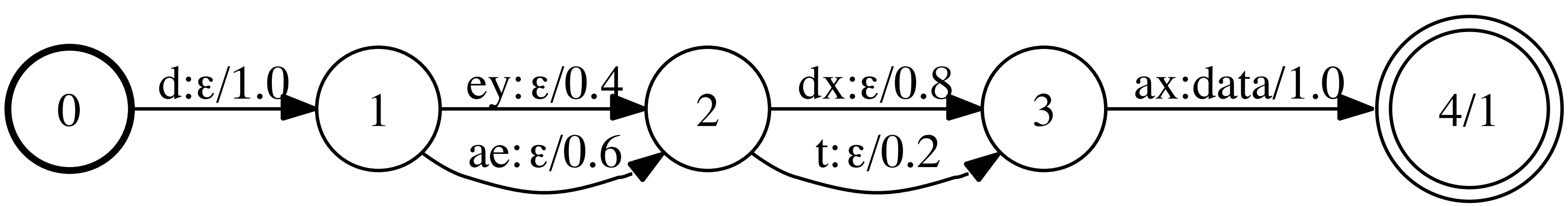


Pronunciation Dictionary

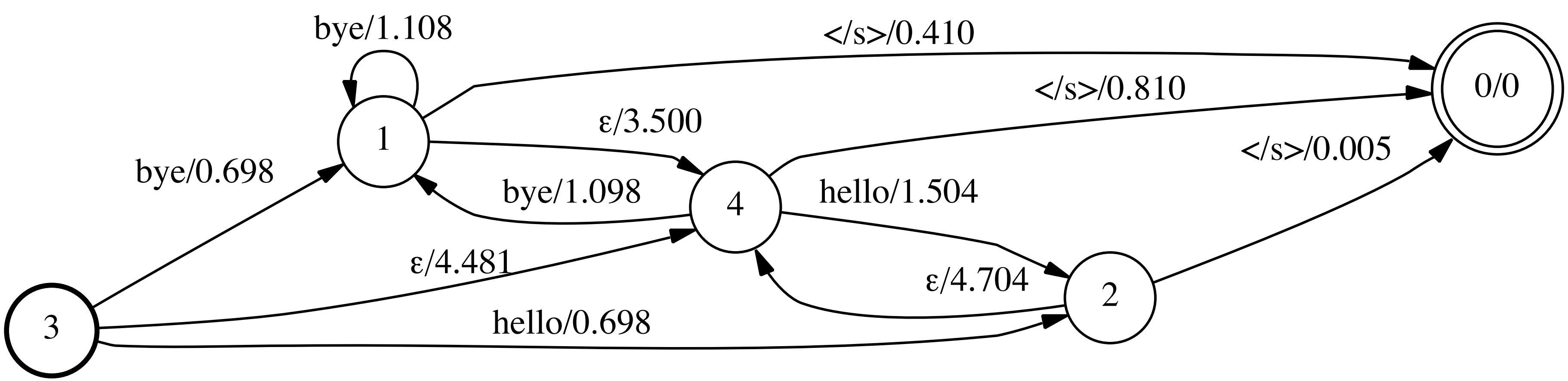
- Phonemic transcription
  - Example: word *data* in American English.

data	D ey dx ax	0.32
data	D ey t ax	0.08
data	D ae dx ax	0.48
data	D ae t ax	0.12

- Representation

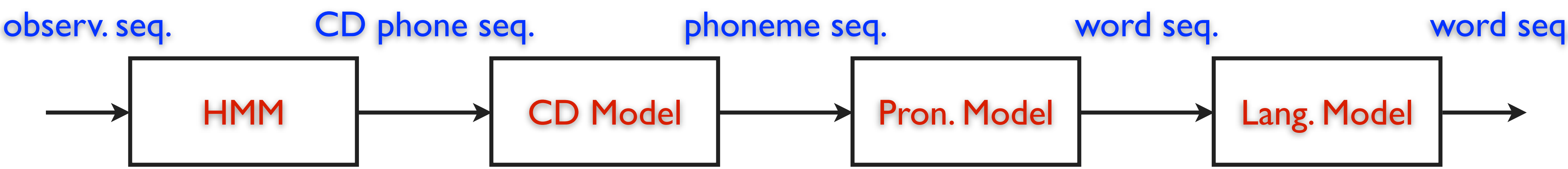


N-Gram Models - Representation



Recognition Cascade

- Combination of components



- Viterbi approximation

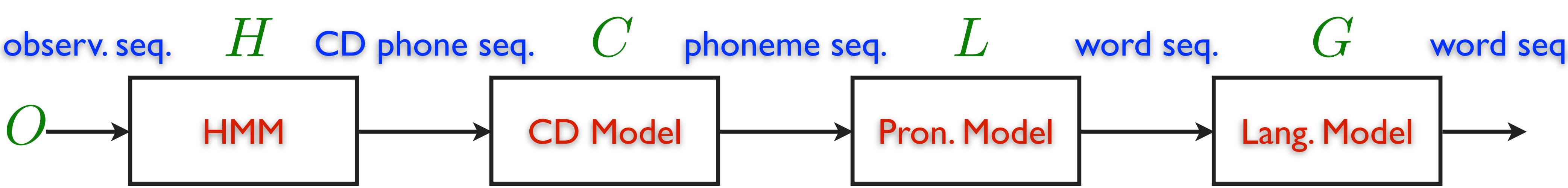
$$\hat{w} = \operatorname*{argmax}_{w} \sum_{d,c,p} \Pr[o \mid d] \Pr[d \mid c] \Pr[c \mid p] \Pr[p \mid w] \Pr[w]$$

$$\approx \operatorname*{argmax}_{w} \max_{d,c,p} \Pr[o \mid d] \Pr[d \mid c] \Pr[c \mid p] \Pr[p \mid w] \Pr[w].$$

Model Combination

- Steps:
  - models represented by weighted transducers.
  - Viterbi approximation: semiring change.
  - composition of weighted transducers.

$$w = \operatorname*{argmin}_{w} \Pi_2 \left[O \star H \circ C \circ L \circ G \right].$$



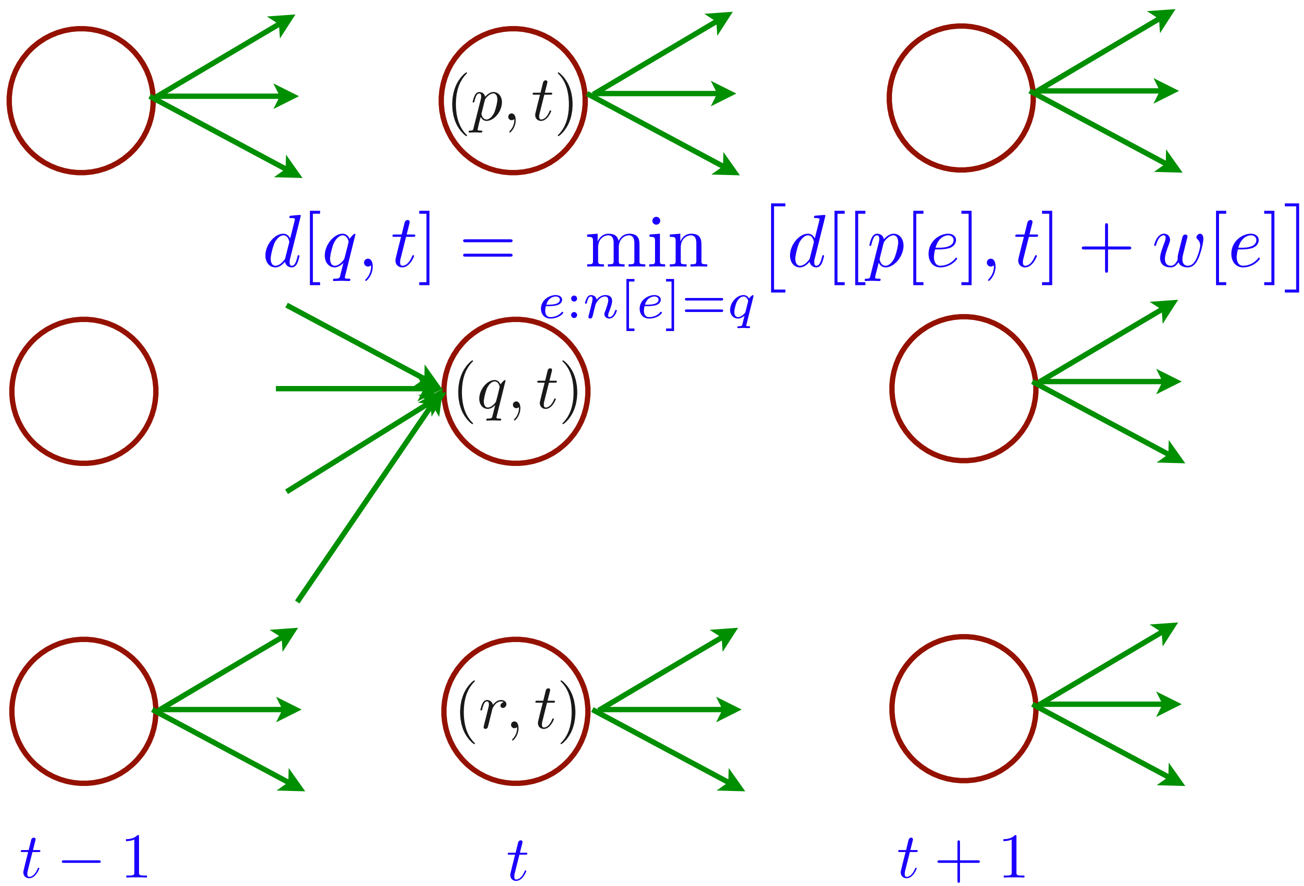
Search Problem

- Problem:
  - size of composed transducer prohibitively large.
  - visiting all states and transitions impractical.
  - how to combine models efficiently and return the best transcription?
- Consequences:
  - pruning.
  - search errors.

Viterbi Algorithm

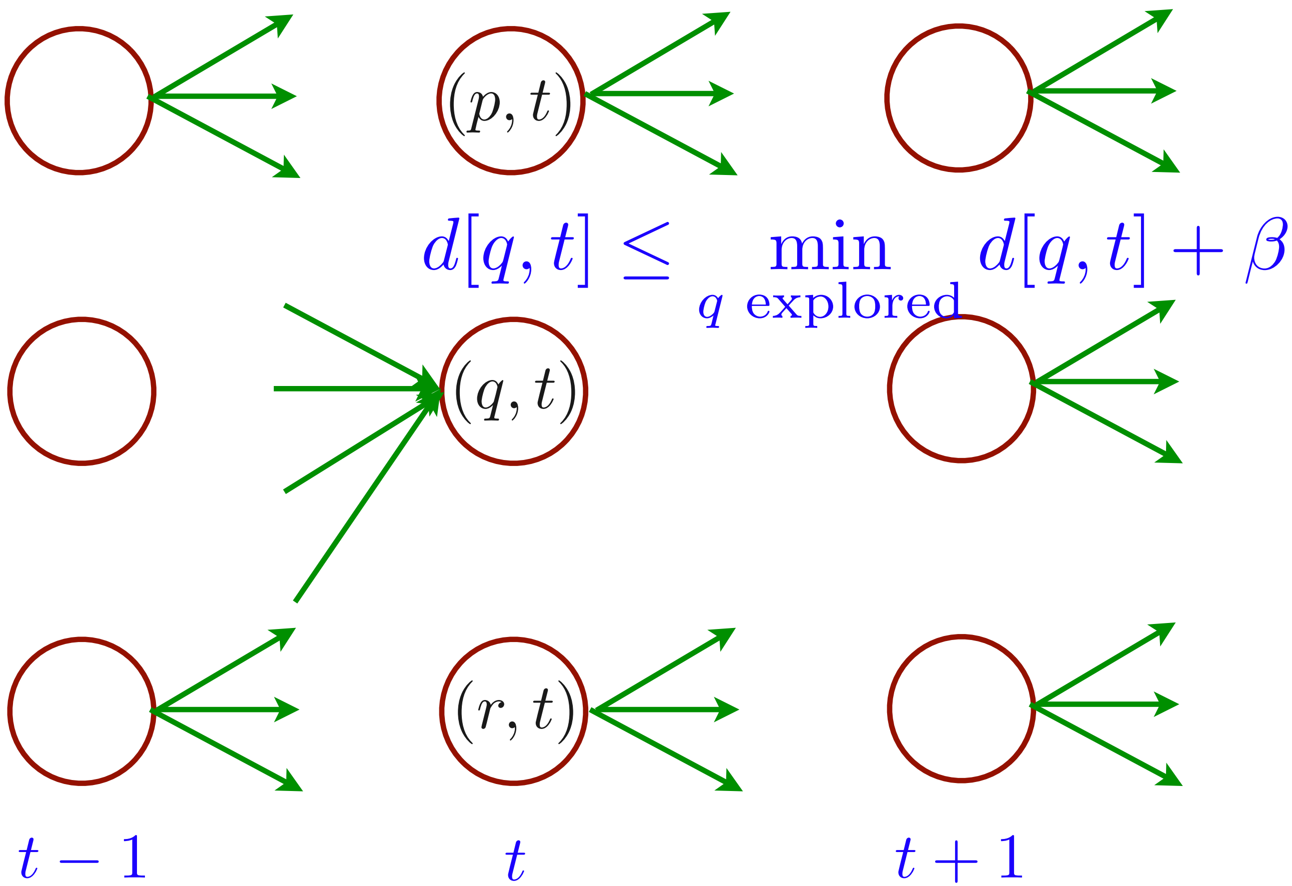
- Specific shortest-distance algorithm

$$d[q,t] = \min_{e:n[e]=q} \left[d[[p[e],t] + w[e]] \right]$$



Beam Pruning

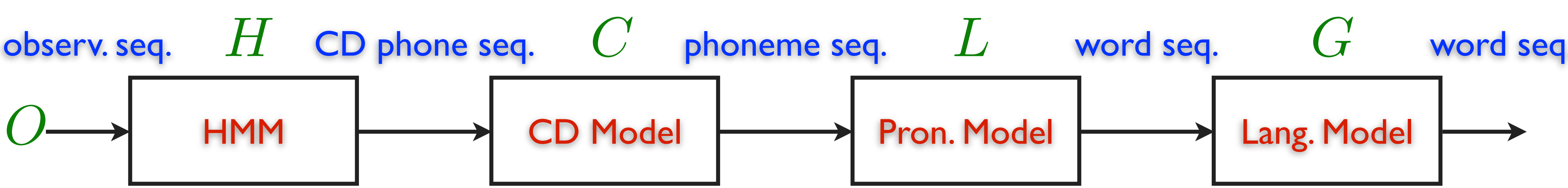
- Time-synchronous beam search: at each time t keep only states within a fixed threshold β of the best.



$$d[q,t] \leq \min_{q \text{ explored}} d[q,t] + \beta$$

Search Modes

- On-the-fly composition: $H \circ C \circ L \circ G$.
  - advantages: components can be modified, e.g., dynamic grammars. Memory usage.
- Off-line composition: full $H \circ C \circ L \circ G$ or parts.
  - advantage: recognition transducer optimization.



Key Optimization Ideas

- General algorithms: as opposed to *ad hoc* solutions.
  - Recognition transducer redundancy: use determinization to reduce or eliminate redundancy. But: not all weighted transducers are determinizable.
  - Recognition transducer size: use minimization to reduce space.
  - Recognition transducer weight distribution: use weight pushing to standardize weight distribution.

Disambiguation & Determinizability

- Determinizability of $L \circ G$: use auxiliary symbols to deal with homophones and unbounded delay. Transformation $L \to \tilde{L}$ according to:

$$\begin{array}{ll} \texttt{r eh d } \#_0 & \textit{read} \\ \texttt{r eh d } \#_1 & \textit{red} \end{array}$$

- Determinizability of $C \circ L \circ G$: self-loops used to propagate auxiliary symbols to context-dependency level, $C \to \tilde{C}$.
- Determinizability of $H \circ C \circ L \circ G$: self-loops at initial state, auxiliary CD symbols mapped to new distinct distribution names, $H \to \tilde{H}$.

Recognition Transducer Optimization

(MM and Riley, 2001; MM, Pereira and Riley, 2007)

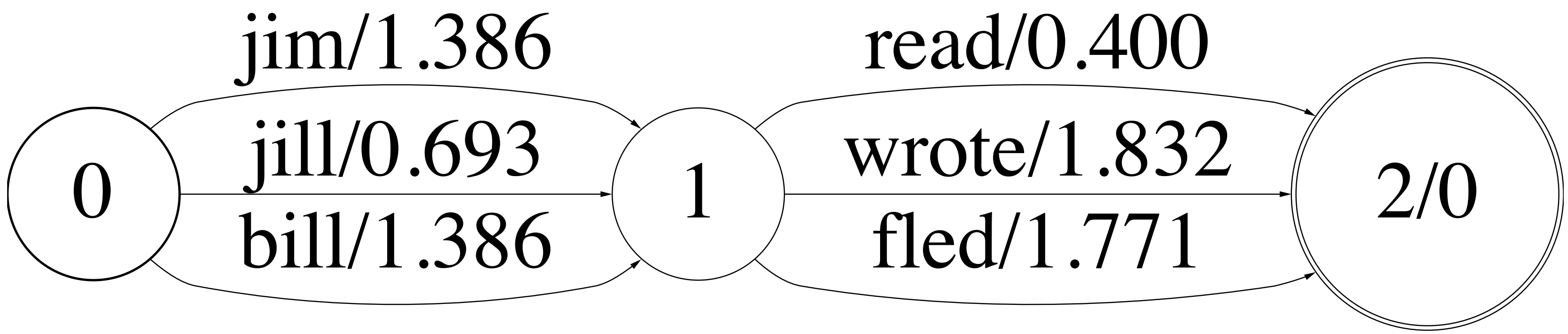
- Optimization cascade:

$$N = push(\sigma_\epsilon(min(det(\tilde{H} \circ det(\tilde{C} \circ det(\tilde{L} \circ G)))))).$$

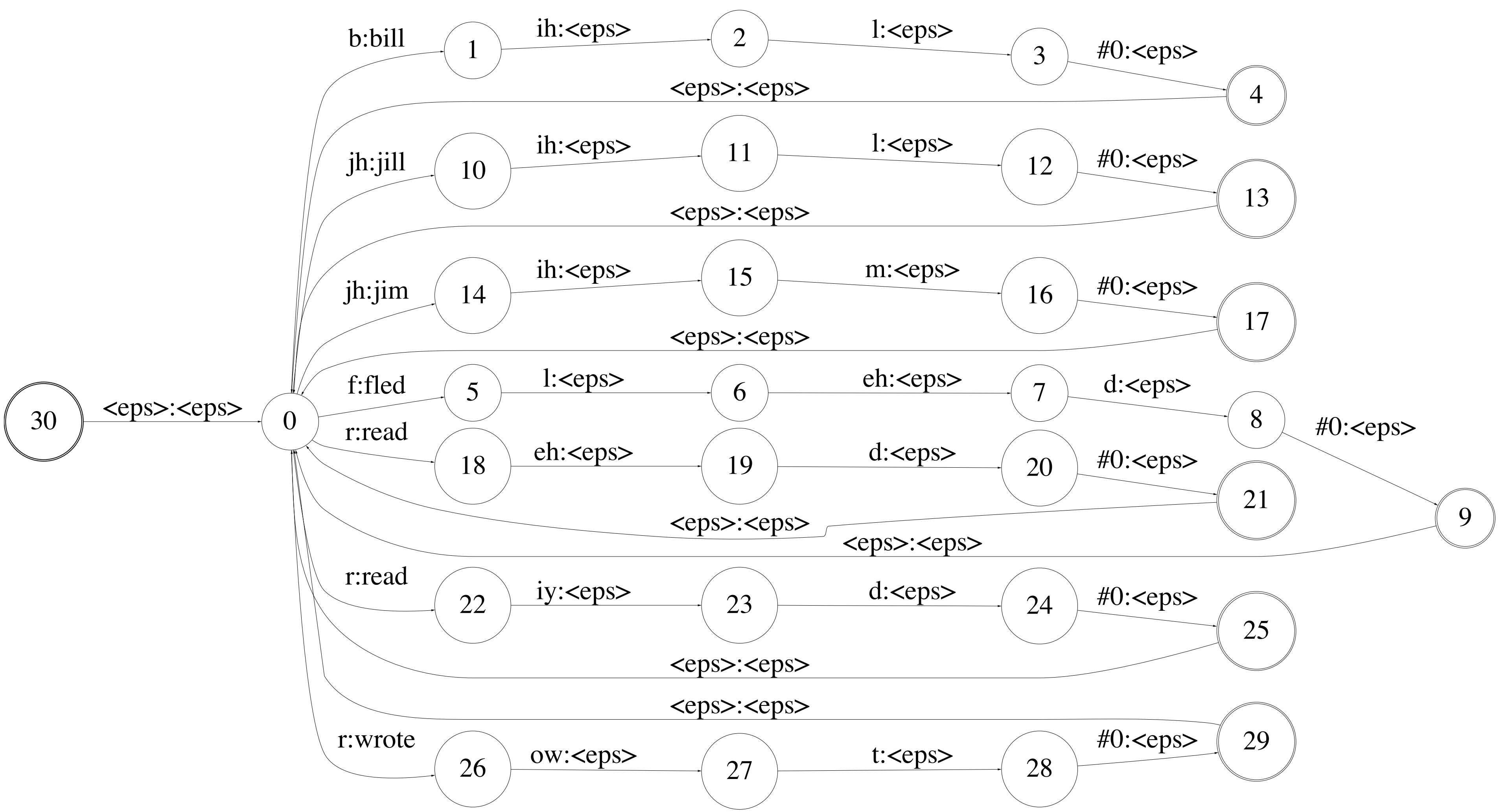
replace auxiliary symbols by ϵ

- Other more general methods for making weighted transducers determinizable (Allauzen and MM, 2004).

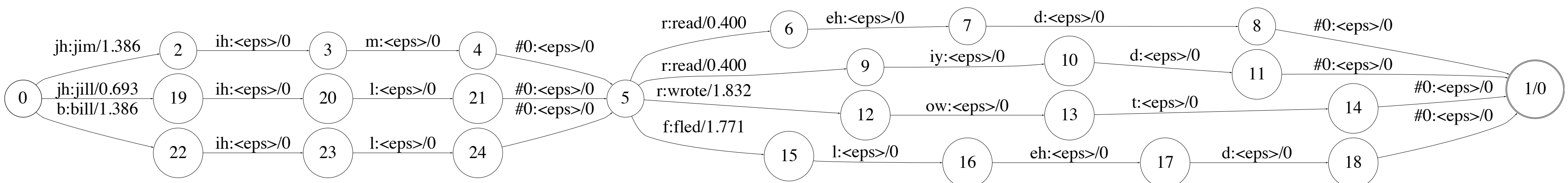
Example - G


jim/1.386
jill/0.693
bill/1.386
read/0.400
wrote/1.832
fled/1.771
0
1
2/0

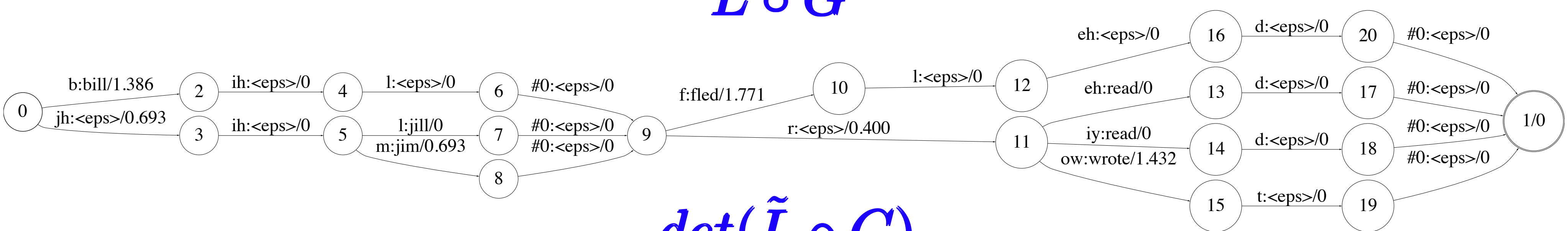
Example - *L*


30
<eps>:<eps>
0
b:bill
1
ih:<eps>
2
l:<eps>
3
#0:<eps>
4
<eps>:<eps>
jh:jill
10
ih:<eps>
11
l:<eps>
12
#0:<eps>
13
<eps>:<eps>
jh:jim
14
ih:<eps>
15
m:<eps>
16
#0:<eps>
17
<eps>:<eps>
f:fled
5
l:<eps>
6
eh:<eps>
7
d:<eps>
8
#0:<eps>
9
r:read
18
eh:<eps>
19
d:<eps>
20
#0:<eps>
21
<eps>:<eps>
<eps>:<eps>
r:read
22
iy:<eps>
23
d:<eps>
24
#0:<eps>
25
<eps>:<eps>
<eps>:<eps>
r:wrote
26
ow:<eps>
27
t:<eps>
28
#0:<eps>
29

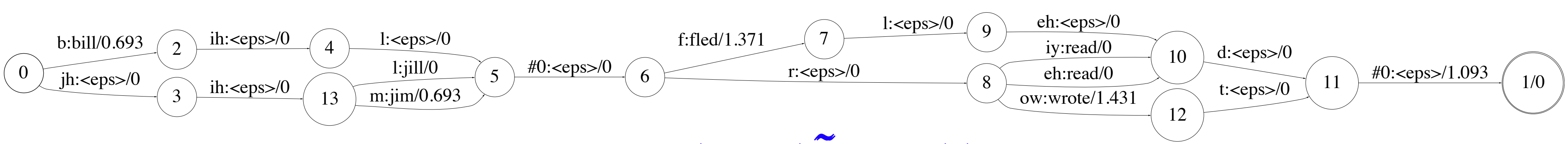
Example



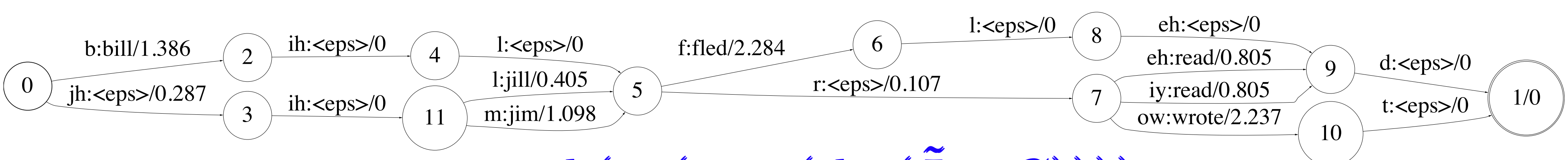
$\tilde{L} \circ G$



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



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



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

Recognition Transducer Standardization

- Minimal deterministic weighted transducers: unique up to state renumbering and to any weight and output label redistribution that preserves the total path weights and output strings.
- Weight-pushed transducer: selects a specific weight distribution along paths while preserving total path weights.
- Result is a standardized recognition transducer.

Factoring

- **Idea**:
  - decoder feature: separate representation for variable-length HMMs (time and space efficiency).
  - To take advantage of this feature, factor integrated transducer $N = H' \circ F$.
- **Algorithm**:
  - Replace input of each linear path in N by a single label naming an *n*-state HMM.
  - Define gain of the replacement of linear path:

$$G(\sigma) = \sum_{\pi \in \mathrm{Lin}(N), i[\pi]=\sigma} |\sigma| - |o[\pi]| - 1.$$

1st-Pass Recognition Networks
40K NAB Task

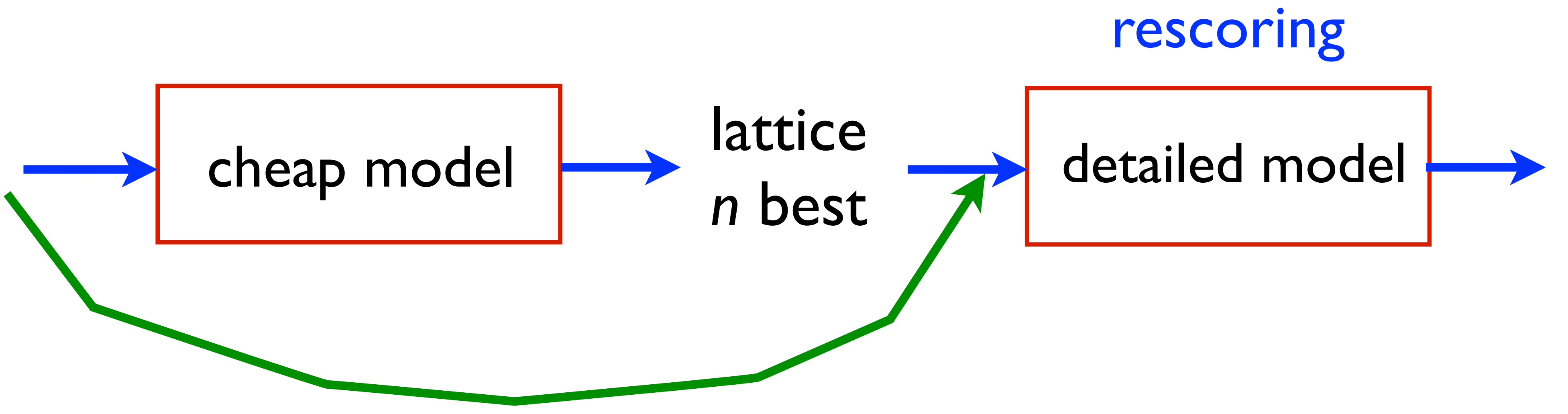
network	states	transitions
G	1,339,664	3,926,010
$L \circ G$	8,606,729	11,406,721
$det(L \circ G)$	7,082,404	9,836,629
$C \circ det(L \circ G))$	7,273,035	10,201,269
$det(H \circ C \circ L \circ G)$	18,317,359	21,237,992
F	3,188,274	6,108,907
$min(F)$	2,616,948	5,497,952

1st-Pass Recognition Networks 40K NAB Eval '95

transducer	x real-time
$C \circ L \circ G$	12.5
$C \circ det(L \circ G)$	1.2
$det(H \circ C \circ L \circ G)$	1.0
$push(min(F))$	0.7

Recognition speed of the first-pass networks in the NAB 40,000-word vocabulary task at 83% word accuracy.

Rescoring

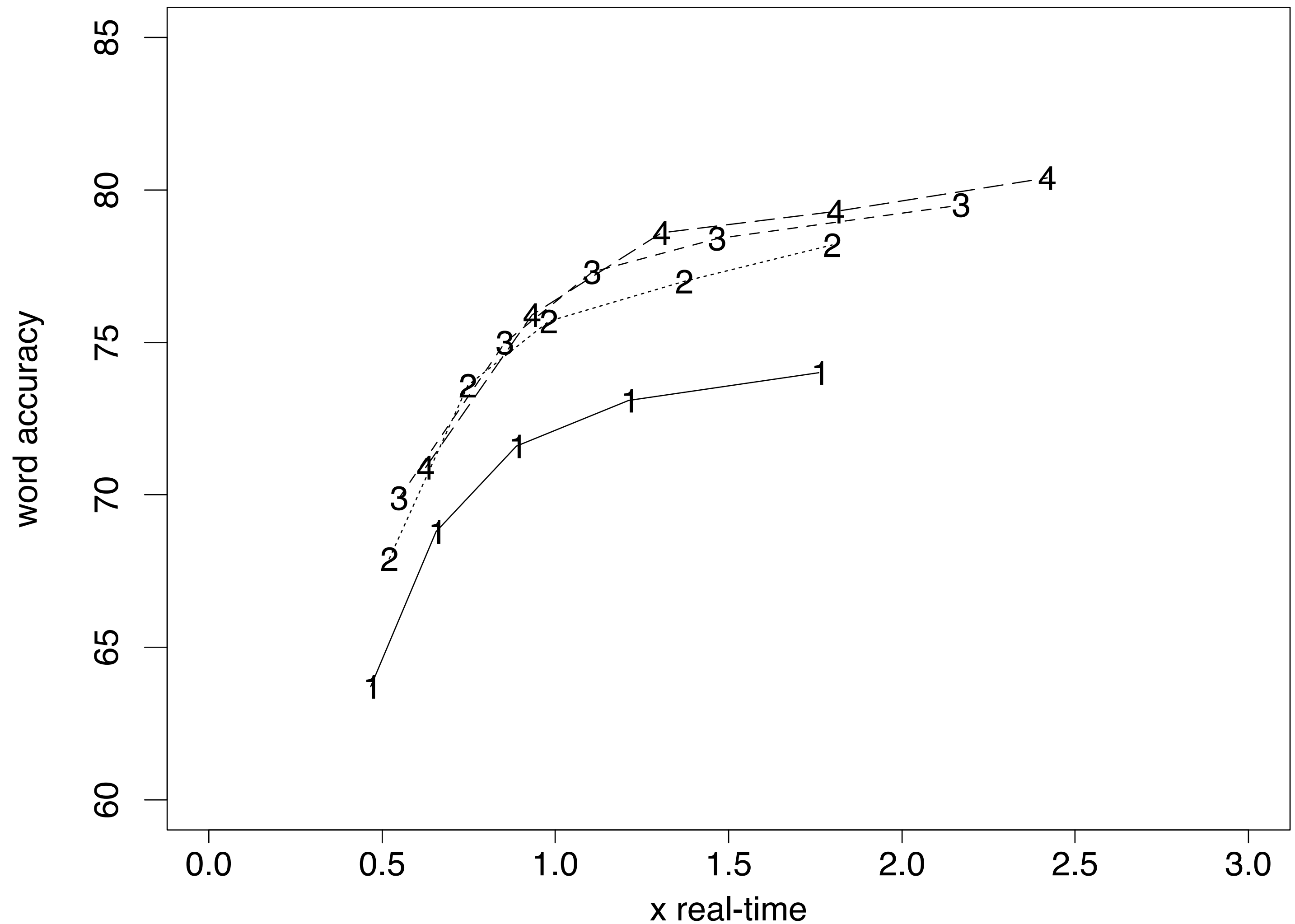


2nd-Pass Recognition Speed 160K NAB Eval '95

network	x real-time
$C \circ L \circ G$	.18
$C \circ det(L \circ G)$	.13
$C \circ push(min(det(L \circ G)))$	.02

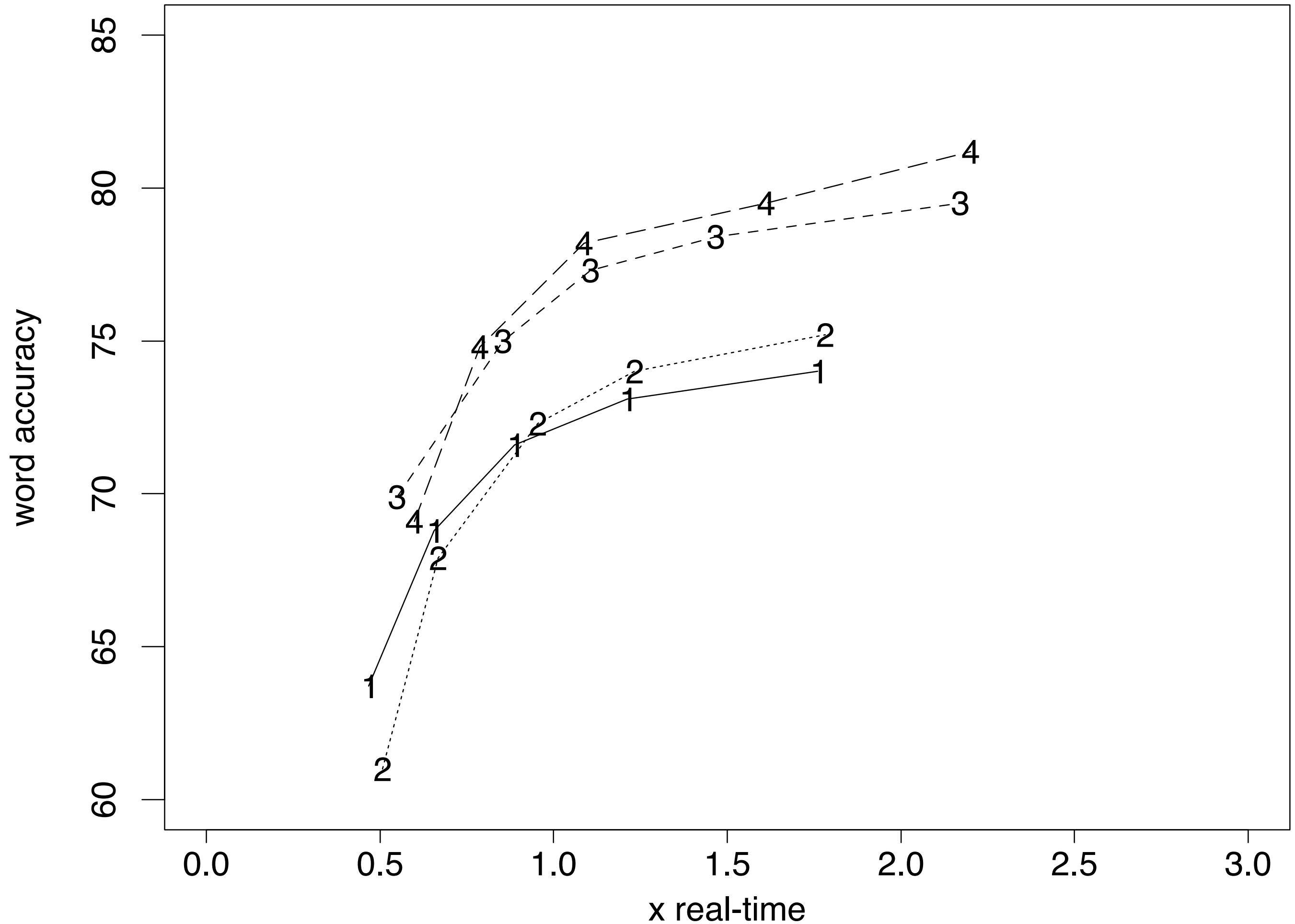
Recognition speed of the second-pass networks in the NAB 160,000-word vocabulary task at 88%.

Effect of Vocabulary Size NAB Eval '95



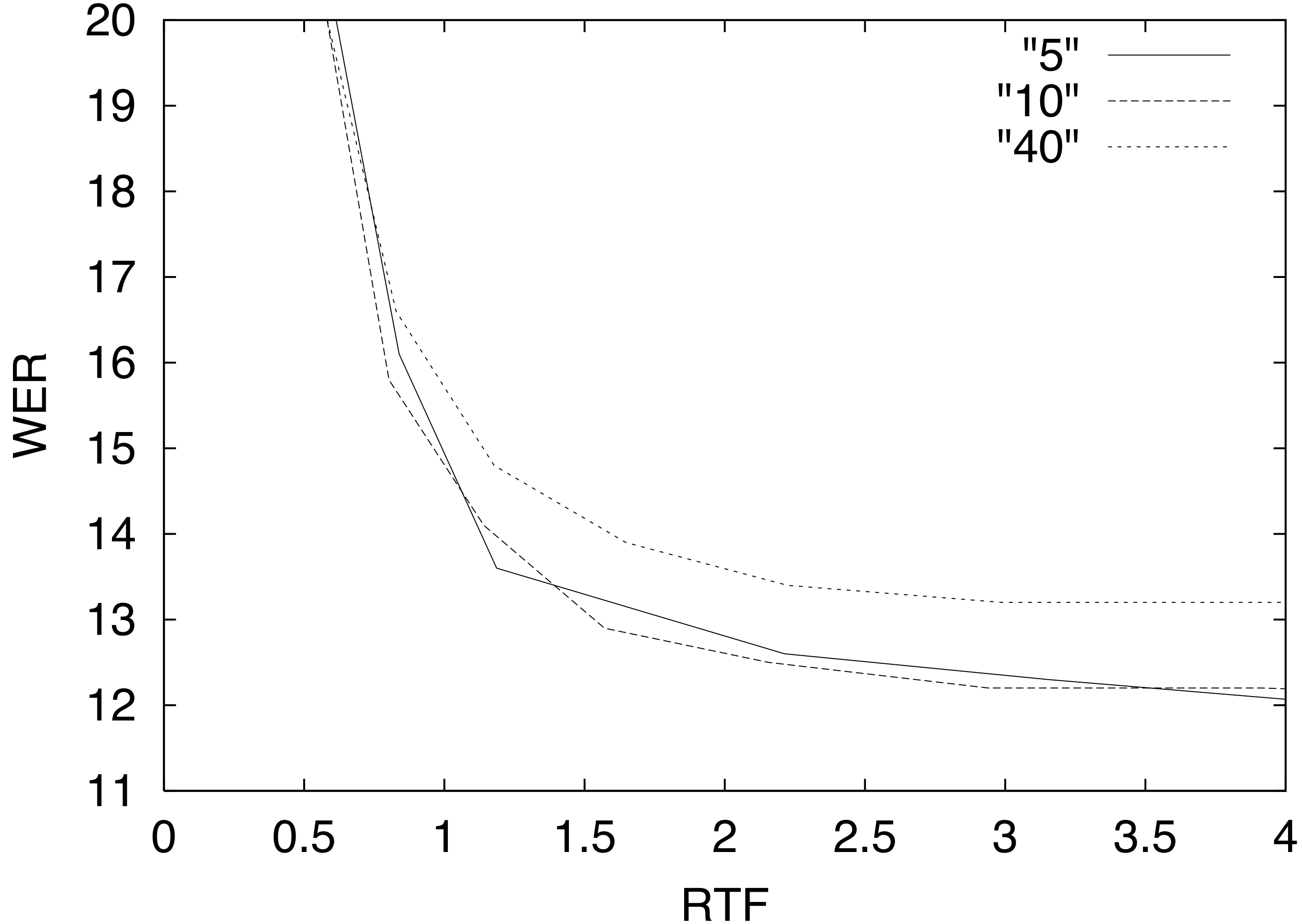
Bigram recognition results for vocabularies of (1) 10,000 words, (2) 20,000 words, (3) 40,000 words, and (4) 160,000 words. (LG Optimized Only.)

Effect of N-gram Order
NAB Eval '95



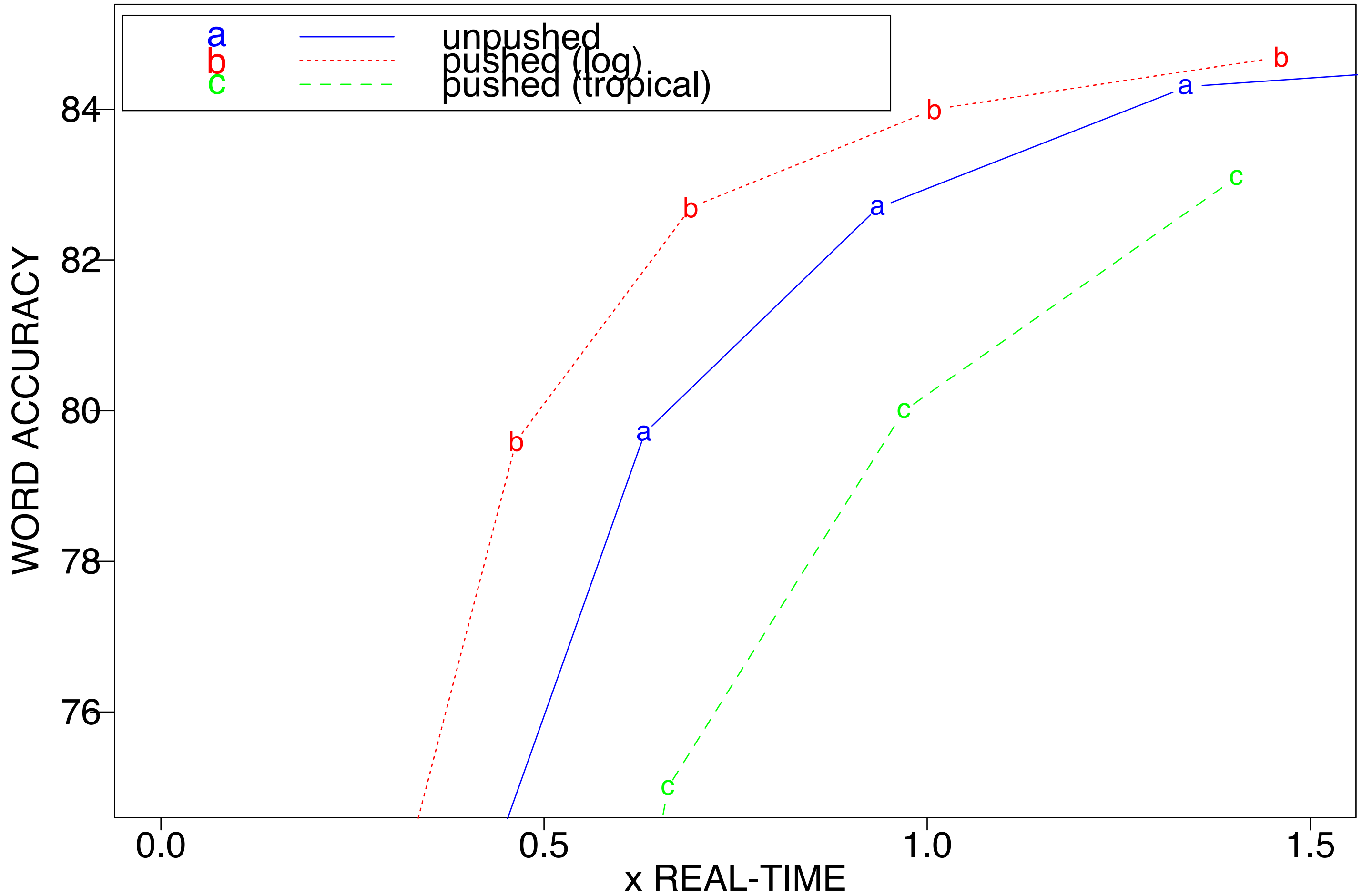
Recognition results for a (1) 10,000 word bigram, (2) 10,000 word trigram, (3) 40,000 word bigram, and (4) 40,000 word trigram. (LG Optimized Only.)

Effect of Shrink Parameter NAB Eval '95



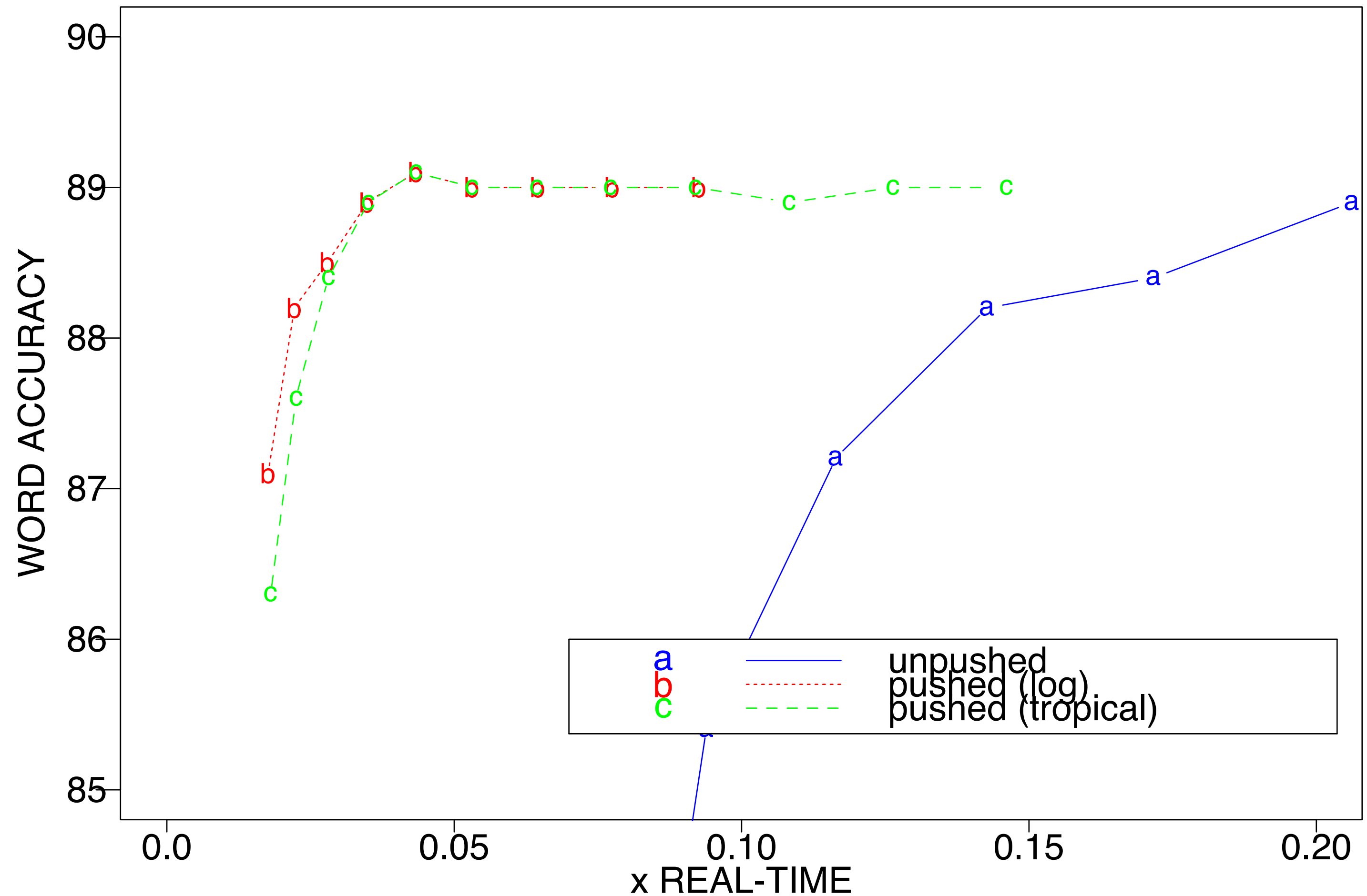
Recognition results for shrink factors (Seymore & Rosenfeld, 1996) of 5, 10, and 40.

Effect of Pushing
1st Pass, 40K NAB Eval '95



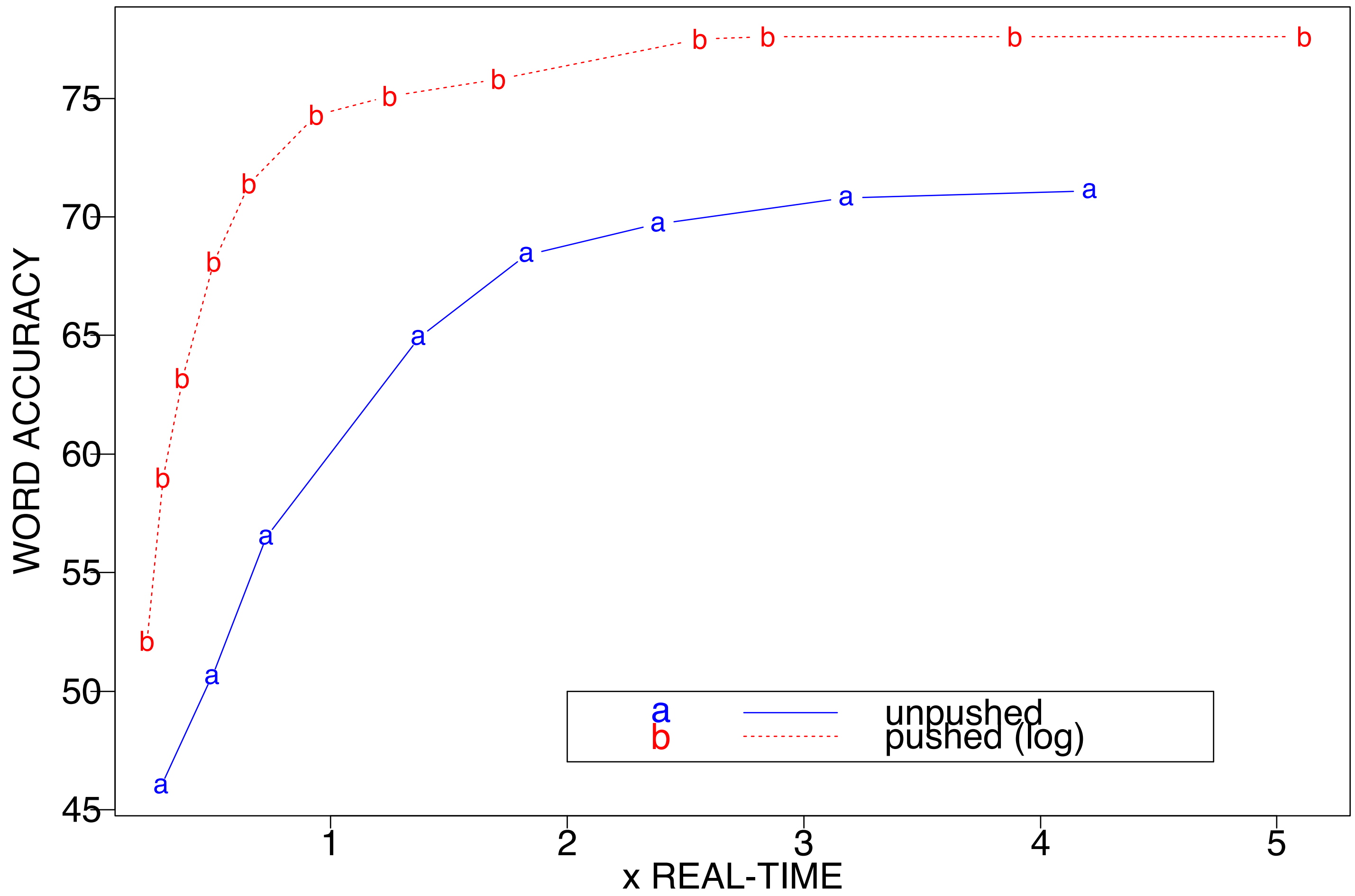
40K-word NAB 1st-pass recognition, 6,108,907-transition determinized and factored HMM-to-word transducer [Alpha 21284].

Effect of Pushing
2nd-Pass, 160K NAB Eval '95



160,000-word vocabulary NAB task, weight-pushing determinized HMM-to-word transducer lattices [Alpha 21284].

100K Names Recognition



100K names recognition, the effect of weight-pushing [SGI~Origin~2000].

Model Combination by Lattice Intersection - SWBD Eval '00

Word Error Rate (%)						
Model/pass	Mod1	Mod2	Mod3	Mod4	Mod5	Mod6
MLLR	30.3	30.2	30.8	30.7	31.4	32.6
Combined	30.3	29.6	28.9	28.8	28.7	28.6

References

- Cyril Allauzen and Mehryar Mohri. An Optimal Pre-Determinization Algorithm for Weighted Transducers. Theoretical Computer Science, 328(1-2):3-18, November 2004.
- Cyril Allauzen, Mehryar Mohri, Brian Roark, and Michael Riley. A Generalized Construction of Integrated Speech Recognition Transducers. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004). Montréal, Canada, May 2004.
- Mehryar Mohri. Statistical Natural Language Processing. In M. Lothaire, editor, Applied Combinatorics on Words. Cambridge University Press, 2005.
- Mehryar Mohri, Michael Riley, Don Hindle, Andrej Ljolje, and Fernando C. N. Pereira. Full Expansion of Context-Dependent Networks in Large Vocabulary Speech Recognition. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP '98). Seattle, Washington, 1998.
- Mehryar Mohri and Michael Riley. Integrated Context-Dependent Networks in Very Large Vocabulary Speech Recognition. In Proceedings of the 6th European Conference on Speech Communication and Technology (Eurospeech '99). Budapest, Hungary, 1999.

References

- Mehryar Mohri and Michael Riley. A Weight Pushing Algorithm for Large Vocabulary Speech Recognition. In Proceedings of the 7th European Conference on Speech Communication and Technology (Eurospeech '01). Aalborg, Denmark, September 2001.

- Mehryar Mohri, Fernando C. N. Pereira, and Michael Riley. Speech Recognition with Weighted Finite-State Transducers. In Larry Rabiner and Fred Juang, editors, Handbook on Speech Processing and Speech Communication, Part E: Speech recognition. volume to appear. Springer-Verlag, Heidelberg, Germany, 2007.

- Michael Riley and Andrej Ljolje. Lexical access with a statistically-derived phonetic network. In Proceedings of the European Conference on Speech Communication and Technology, pages 585-588, 1991.