LING5702: Lecture Notes 14
A Model of Memory Bounds as Interference

Contents

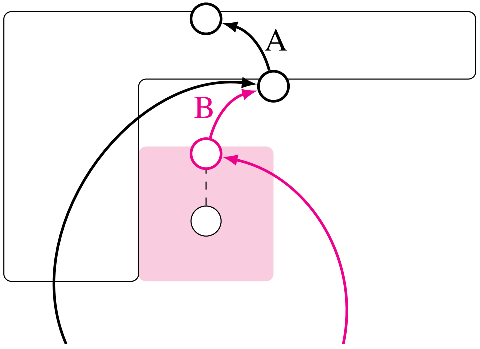
14.1 Review: parser operations use different amounts of cued associations 1
14.2 More cued associations mean more risk of interference 2
14.3 Simulation model [Rasmussen & Schuler, 2018] 3
14.4 Simulation results [Rasmussen & Schuler, 2018] 6

14.1 Review: parser operations use different amounts of cued associations

Comprehension proceeds as follows, using modified terminal and nonterminal decisions:

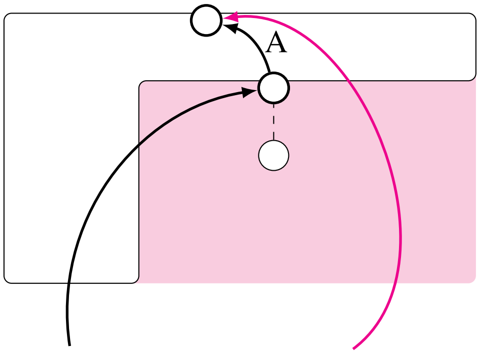
1. a **terminal** decision is made about whether to **match** store elements at the next word, and

   (a) **no** terminal (lexical) match:

   — **no** associations cued before any form

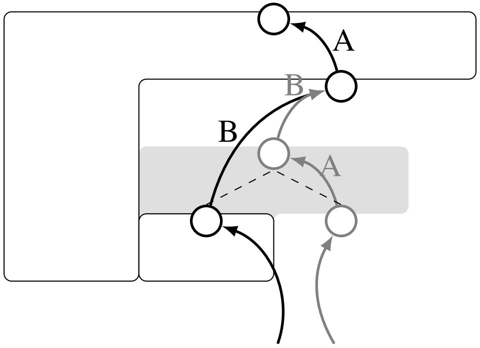
   (b) **yes** terminal (lexical) match:

   — **one** association cued before any form

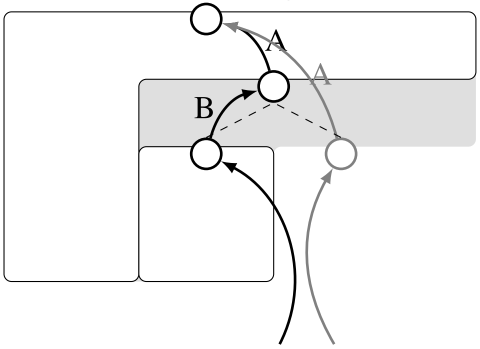
2. a **non-terminal** decision is made about whether to **match** store elements at the next rule,

   (c) **no** non-terminal (grammatical) match:

   — **one** association cued before any form

   (d) **yes** non-terminal (grammatical) match:

   — **two** associations cued before any form

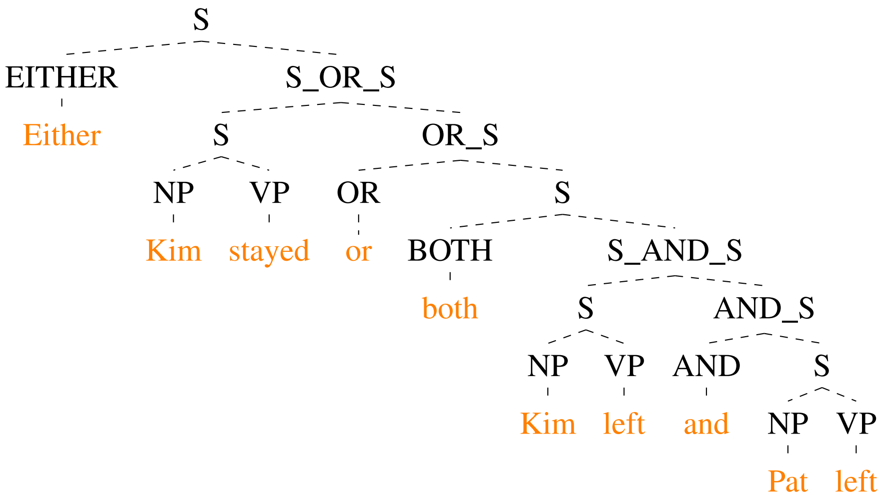
14.2 More cued associations mean more risk of interference

As cued associations for the same sentence are added, the risk of interference increases.

Perfect cueing of each target must avoid all other interfering cues.

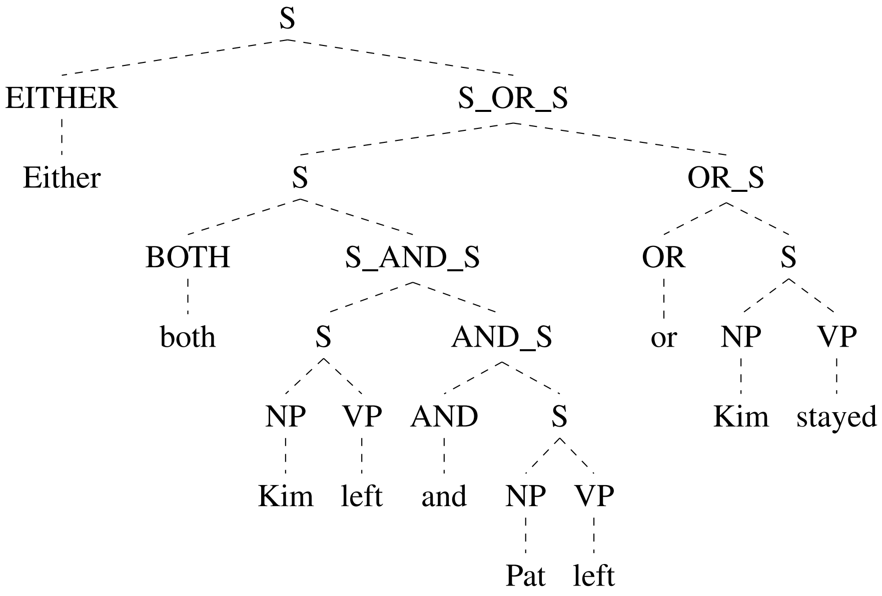
Operations that involve more cueing should happen earlier, to avoid interference.

1. For example, a right branching structure does cueing early, encounters less clutter:



	singly center-embedded sentence			
step	operations	avoidances	cumu	resulting store ; remaining input
0	(initial)			T/T ; *Either Kim stayed or . . .*
1	T:no N:no	0 × 1 =	0	T/T, S/S_OR_S ; *Kim . . .*
2	T:no N:no	+ 1 × 1 =	1	T/T, S/S_OR_S, S/VP ; *stayed . . .*
3	T:yes N:yes	+ 2 × 3 =	7	T/T, S/OR_S ; *or . . .*
4	T:no N:yes	+ 3 × 2 =	13	T/T, S/S ; *both . . .*
5	T:no N:yes	+ 4 × 2 =	21	T/T, S/S_AND_S ; *Kim . . .*
6	T:no N:no	+ 5 × 1 =	26	T/T, S/S_AND_S, S/VP ; *left . . .*
7	T:yes N:yes	+ 6 × 3 =	44	T/T, S/AND_S ; *and . . .*
8	T:no N:yes	+ 7 × 2 =	58	T/T, S/S ; *Pat . . .*
9	T:no N:yes	+ 8 × 2 =	74	T/T, S/VP ; *left*
10	T:yes N:yes	+ 9 × 3 =	101	T/T ;

2. A center embedded structure does cueing later, encounters more clutter:



	doubly center-embedded sentence			
step	operations	avoidances	cumu	resulting store ; remaining input
0	(initial)			T/T ; *Either both Kim left and ...*
1	T:no N:no	$0 \times 1 =$	0	T/T, S/S_OR_S ; *both ...*
2	T:no N:no	$+ 1 \times 1 =$	1	T/T, S/S_OR_S, S/S_AND_S ; *Kim ...*
3	T:no N:no	$+ 2 \times 1 =$	3	T/T, S/S_OR_S, S/S_AND_S, S/VP ; *left ...*
4	T:yes N:yes	$+ 3 \times 3 =$	12	T/T, S/S_OR_S, S/AND_S ; *and ...*
5	T:no N:yes	$+ 4 \times 2 =$	20	T/T, S/S_OR_S, S/S ; *Pat ...*
6	T:no N:yes	$+ 5 \times 2 =$	30	T/T, S/S_OR_S, S/VP ; *left ...*
7	T:yes N:yes	$+ 6 \times 3 =$	48	T/T, S/OR_S ; *or ...*
8	T:no N:yes	$+ 7 \times 2 =$	62	T/T, S/S ; *Kim ...*
9	T:no N:yes	$+ 8 \times 2 =$	78	T/T, S/VP ; *stayed*
10	T:yes N:yes	$+ 9 \times 3 =$	105	T/T ;

14.3 Simulation model [Rasmussen & Schuler, 2018]

Notation:

1. Diagonalize (turn a vector into a simple filter): $\mathrm{diag}(\mathbf{v})$
2. Renormalization (rescale **v** to have unit magnitude): $\frac{\mathbf{v}}{||\mathbf{v}||}$

Initialization. Before it begins processing, the model:

1. **randomly generates** initial a top-level derivation fragment and category 'T':

$$\mathbf{a}_0 \in \mathbb{R}^d$$
$$\mathbf{b}_0 \in \mathbb{R}^d$$
$$\mathbf{c}_0 \in \mathbb{R}^d$$

2. **associates** the new signs and category in (time-subscripted) associative memory:

$$\mathbf{A}_0 = \mathbf{a}_0\, \mathbf{b}_0^\top \tag{1}$$
$$\mathbf{B}_0 = \mathbf{0}\, \mathbf{0}^\top \tag{2}$$
$$\mathbf{C}_0 = \mathbf{c}_0\, \mathbf{a}_0^\top + \mathbf{c}_0\, \mathbf{b}_0^\top \tag{3}$$

3. **associates** categories with m words and n grammar rules (as parent, left child, right child):

$$\mathbf{L} = \sum_{m=1}^{M} \mathbf{c}_m\, \mathbf{w}_m^\top \tag{4}$$

$$\mathbf{G}_\mathrm{P} = \sum_{n=1}^{N} \mathbf{r}_n\, \mathbf{c}_n^\top \tag{5a}$$
$$\mathbf{G}_\mathrm{L} = \sum_{n=1}^{N} \mathbf{r}_n\, \mathbf{c}_n'^\top \tag{5b}$$
$$\mathbf{G}_\mathrm{R} = \sum_{n=1}^{N} \mathbf{r}_n\, \mathbf{c}_n''^\top \tag{5c}$$

4. **associates** categories with categories of left- and right-recursive descendants:

$$\mathbf{D}_0' = \mathrm{diag}(\mathbf{1}) \tag{6a}$$
$$\mathbf{D}_0 = \mathrm{diag}(\mathbf{0}) \tag{6b}$$
$$\mathbf{D}_k' = \mathbf{G}_\mathrm{L}^\top\, \mathbf{G}_\mathrm{P}\, \mathbf{D}_{k-1}' \tag{6c}$$
$$\mathbf{D}_k = \mathbf{D}_{k-1} + \mathbf{D}_k' \tag{6d}$$

$$\mathbf{E}_0' = \mathrm{diag}(\mathbf{1}) \tag{7a}$$
$$\mathbf{E}_0 = \mathrm{diag}(\mathbf{0}) \tag{7b}$$
$$\mathbf{E}_k' = \mathbf{G}_\mathrm{R}^\top\, \mathbf{G}_\mathrm{P}\, \mathbf{E}_{k-1}' \tag{7c}$$
$$\mathbf{E}_k = \mathbf{E}_{k-1} + \mathbf{E}_k' \tag{7d}$$

   iterating to a maximum depth of $k = 20$, so $\mathbf{D} = \mathbf{D}_{20}$ and $\mathbf{E} = \mathbf{E}_{20}$.

Terminal phase. At every word t, the model:

1. **cues** a new apex sign:
$$\mathbf{a}_{t-1} = \mathbf{A}_{t-1}\,\mathbf{b}_{t-1} \tag{8}$$
2. **randomly generates** new signs for yes-match and no-match results:
$$\mathbf{a}_{t-.5,\text{yes}} \in \mathbb{R}^d$$
$$\mathbf{a}_{t-.5,\text{no}} \in \mathbb{R}^d$$
3. **filters** a category label for each match result:
$$\mathbf{c}_{t-.5,\text{yes}} = \text{diag}(\mathbf{L}\,\mathbf{w}_t)\,\mathbf{C}_{t-1}\,\mathbf{b}_{t-1} \tag{9a}$$
$$\mathbf{c}_{t-.5,\text{no}} = \text{diag}(\mathbf{L}\,\mathbf{w}_t)\,\mathbf{D}\,\mathbf{C}_{t-1}\,\mathbf{b}_{t-1} \tag{9b}$$
4. **superposes** the possibile signs in attentional focus, weighted by magnitudes of categories:
$$\mathbf{a}_{t-.5} = \frac{(||\mathbf{c}_{t-.5,\text{yes}}||\,\mathbf{a}_{t-.5,\text{yes}}) + (||\mathbf{c}_{t-.5,\text{no}}||\,\mathbf{a}_{t-.5,\text{no}})}{||\,(||\mathbf{c}_{t-.5,\text{yes}}||\,\mathbf{a}_{t-.5,\text{yes}}) + (||\mathbf{c}_{t-.5,\text{no}}||\,\mathbf{a}_{t-.5,\text{no}})\,||} \tag{10}$$
5. **associates** the new signs with categories and with the remainder of the analysis:
$$\mathbf{C}_{t-.5} = \mathbf{C}_{t-1} + \frac{\mathbf{c}_{t-.5,\text{no}}}{||\,\mathbf{c}_{t-.5,\text{no}}\,||}\,\mathbf{a}_{t-.5,\text{no}}{}^\top + \frac{\text{diag}(\mathbf{C}_{t-1}\,\mathbf{a}_{t-1})\,\mathbf{E}^\top\mathbf{c}_{t-.5,\text{yes}}}{||\,\text{diag}(\mathbf{C}_{t-1}\,\mathbf{a}_{t-1})\,\mathbf{E}^\top\mathbf{c}_{t-.5,\text{yes}}\,||}\,\mathbf{a}_{t-.5,\text{yes}}{}^\top \tag{11}$$
$$\mathbf{B}_{t-.5} = \mathbf{B}_{t-1} + \mathbf{b}_{t-1}\,\mathbf{a}_{t-.5,\text{no}}{}^\top + \mathbf{B}_{t-1}\,\mathbf{a}_{t-1}\,\mathbf{a}_{t-.5,\text{yes}}{}^\top \tag{12}$$

Non-terminal phase. Similarly, after each terminal phase, the model:

1. **cues** a new base sign:
$$\mathbf{b}_{t-.5} = \mathbf{B}_{t-.5}\,\mathbf{a}_{t-.5} \tag{13}$$
2. **randomly generates** a new sign for the no-match case (yes- is just old apex), and new base:
$$\mathbf{a}_{t,\text{no}} \in \mathbb{R}^d$$
$$\mathbf{b}_t \in \mathbb{R}^d$$
3. **filters** a grammar rule for each match result:
$$\mathbf{r}_{t,\text{yes}} = \text{diag}(\mathbf{G}_\text{L}\,\mathbf{C}_{t-.5}\,\mathbf{a}_{t-.5})\,\mathbf{G}_\text{P}\,\mathbf{C}_{t-.5}\,\mathbf{b}_{t-.5} \tag{14a}$$
$$\mathbf{r}_{t,\text{no}} = \text{diag}(\mathbf{G}_\text{L}\,\mathbf{C}_{t-.5}\,\mathbf{a}_{t-.5})\,\mathbf{G}_\text{P}\,\mathbf{D}\,\mathbf{C}_{t-.5}\,\mathbf{b}_{t-.5} \tag{14b}$$
4. **superposes** the two possible signs as a new apex, weighted by magnitude of grammar rules:
$$\mathbf{a}_t = \frac{(||\mathbf{r}_{t,\text{yes}}||\,\mathbf{A}_{t-.5}\,\mathbf{b}_{t-.5}) + (||\mathbf{r}_{t,\text{no}}||\,\mathbf{a}_{t,\text{no}})}{||\,(||\mathbf{r}_{t,\text{yes}}||\,\mathbf{A}_{t-.5}\,\mathbf{b}_{t-.5}) + (||\mathbf{r}_{t,\text{no}}||\,\mathbf{a}_{t,\text{no}})\,||} \tag{15}$$
5. **associates** the possible signs with categories and the remainder of the analysis:
$$\mathbf{A}_t = \mathbf{A}_{t-1} + \mathbf{a}_t\,\mathbf{b}_t^\top \tag{16}$$
$$\mathbf{B}_t = \mathbf{B}_{t-.5} + \mathbf{b}_{t-.5}\,\mathbf{a}_{t,\text{no}}{}^\top \tag{17}$$
$$\mathbf{C}_t = \mathbf{C}_{t-.5} + \frac{\mathbf{G}_\text{P}^\top\mathbf{r}_{t,\text{no}}}{||\,\mathbf{G}_\text{P}^\top\mathbf{r}_{t,\text{no}}\,||}\,\mathbf{a}_{t,\text{no}}{}^\top + \frac{\mathbf{G}_\text{R}^\top\mathbf{r}_{t,\text{yes}} + \mathbf{G}_\text{R}^\top\mathbf{r}_{t,\text{no}}}{||\,\mathbf{G}_\text{R}^\top\mathbf{r}_{t,\text{yes}} + \mathbf{G}_\text{R}^\top\mathbf{r}_{t,\text{no}}\,||}\,\mathbf{b}_t^\top \tag{18}$$

14.4 Simulation results [Rasmussen & Schuler, 2018]

The model was run on this grammar, measuring the accuracy of retrieving the end category ‘T’:

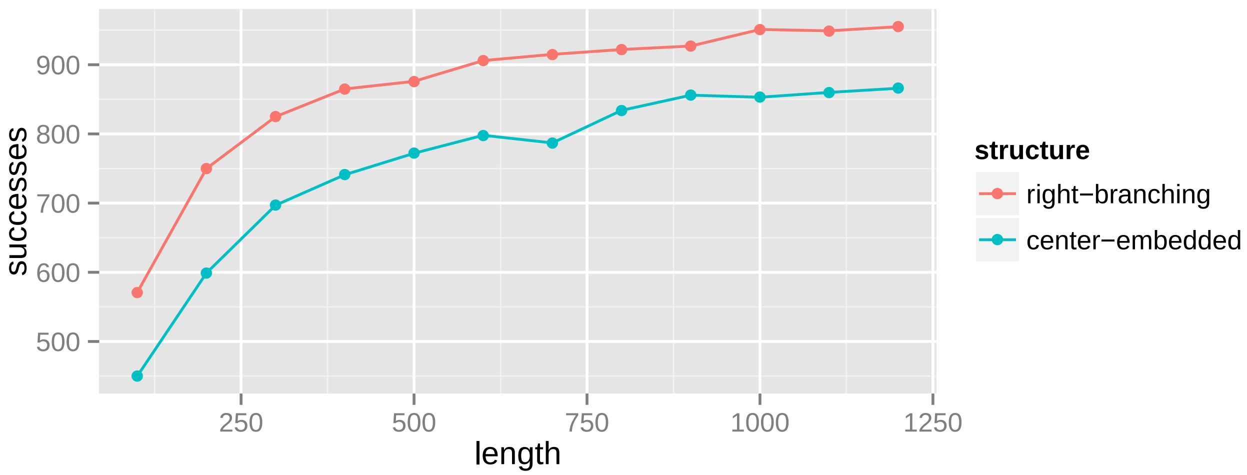
P(S → NP VP) = 0.5
P(S → EITHER S OR S) = 0.25
P(S → BOTH S AND S) = 0.25
P(VP → leaves) = 0.5
P(VP → stays) = 0.5

P(NP → kim) = 0.5
P(NP → pat) = 0.5
P(BOTH → both) = 1.0
P(AND → and) = 1.0
P(EITHER → either) = 1.0
P(OR → or) = 1.0

Like people, it shows higher difficulty for center embedding:

sentence	correct	incorrect
center-embedded	470	530
right-branching	555	445

The effect persists even as the vector size increases, suggesting it’s not just due to capacity bounds:



References

[Rasmussen & Schuler, 2018] Rasmussen, N. E. & Schuler, W. (2018). Left-corner parsing with distributed associative memory produces surprisal and locality effects. *Cognitive Science*, 42(S4), 1009–1042.