ACT-R 5.0 Subsymbolic

Christian Lebiere
Human-Computer Interaction Institute
Carnegie Mellon University
cl+@cmu.edu

Parameters and Equations

- Reduce number of parameters and rationalize the equations
- Increase symmetry between declarative and procedural parts
- Changes:
  - Simplification of utilities parameters from q, a, r and b to p and c
  - Production strength is eliminated from retrieval time equation
  - Generalized base-level activation provides precision and efficiency
  - Noise added to retrieval (and utility) threshold unifies chunk choice and retrieval probability equations into single equation
  - Competitive latency equation is symmetrical to chunk choice equation and yields fan effect and number and similarity of distractors
  - Similarities and associations are left basically as is for now (with a minor reworking) but similarities become the default associations

Equations

Declarative	**Procedural**
1. $A_i = B_i + \sum_j W_j S_{ji} + \sum_k P_k M_{ki}$ j sources and k probes	1. $U_i = P_i G - C_i$ note G $\approx$ 20
2. $P(i) = \dfrac{e^{A_i / t_a}}{\sum_j e^{A_j / t_a} + e^{t_a / t_a}}$	2. $P(i) = \dfrac{e^{U_i / t_u}}{\sum_j e^{U_j / t_u} + e^{t_u / t_u}}$
3. $T(i) = F \dfrac{\sum e^{A_j / t} + e^{t_a / t}}{e^{A_i / t}}$	3. $T(i) = 50\ ms$?

Production Utility

- Get rid of q, a, r and b and instead use simply p and c
- Solves problem with q and with too many parameters
- Chance and effort used as before but correspond to p and c
- One of the few changes that is not backward-compatible
- Consistent with loss of goal stack as architectural primitive
- No subgoal discounting: G remains fixed (unit task level?)
- Lack of structured learning, i.e. “seeing across subgoals”
- No architectural role in goal selection?
- Automatic success if goal changes w/o explicit outcome?
- Popping upon failure is off by default

Strength and Approximation

- Production strength is out
  - Misfit with activation scale for chunks and utility for productions
  - Contradiction between effect on retrieval speed but not probability
  - Still used in old latency equation and can be printed on demand
- Generalized base-level approximation
  - OL can be either nil, t or a positive number m (recent references)

$$B = \log\left(\sum_{i=1}^{m} t_{n-i}^{-d} + \frac{(n-m)\left(t_0^{1-d} - t_{n-m}^{1-d}\right)}{(1-d)\left(t_0 - t_{n-m}\right)}\right)$$

  - Combines precision of exact formula with efficiency of approximation
  - But is the short-term boost really right (see last year's discussion)?

Noisy Threshold

- Add noise to retrieval threshold (and utility threshold too)
- Unifies retrieval probability and chunk choice equation into a new generalized equation in which the threshold is just another competitor:

$$P(i) = \frac{e^{A_i / t_a}}{\sum_j e^{A_j / t_a} + e^{t_a / t_a}}$$

- Increases variability in behavior (probability and latency)
- Only transient activation noise (interpretation of threshold)
- Different noise every cycle or retrieval (no difference in 5.0)
- Requires minor parameter adjustment to noise value (s-> t)

Competitive Latency

- Retrieval latency is symmetric with chunk choice equation:

$$T(i) = F\frac{\sum e^{A_j/\tau} + e^{\tau_a/\tau}}{e^{A_i/\tau}}$$

- Equivalent to activation scaling to enforce log odds rule
- Factors in the number, similarity and priming of distractors
- Can account for systematic varying of parameters (F,τ)
- Result from basic inhibition between competing chunks
- Works best with indirect partial matches
  - Direct matches: **only 1 considered** (or partial match header?)
  - Exact matching: **usually 1 match** (or consider all chunks?)
- Old latency equation is still available as option

Similarities and Associations

- Associations and similarities both provide context-sensitivity
- More similar than different: unify into single mechanism
- Little support for strengths of association as currently defined
- Main function of strengths assumed by competitive latency
- Current definition of strengths violates integration constraint
- Very few uses of associative learning compared to others
- Co-occurrences learned from source to any slot in any chunk
- Most stringent test to date yielded definite predictions
- They were consistently not supported by the data
- There are other pathologies

Prospective Memory Example

- ? **(sdm isa lexicon word spread)**
- Lexicon0 2.577
- isa LEXICON
- context nil
- word Spread
- spelling S-P-R-E-A-D
- (LEXICON0)
- ? **(sdp lexicon0)**
- Parameters for chunk Lexicon0:
- :Activation 2.577
- :Source nil
- :Base-Level 2.191
- :Creation-Time 0.000
- :References (1.0)
- :Source-Spread 0.000
- :IAs ((S-P-R-E-A-D . 3.466) (Spread . 3.466) (Lexicon0 . 4.159))
- :Creation-Cycle 0.000
- :Needed 0.000
- :Contexts 0.000
-
- (Lexicon0)
- ? **(inspect-activation)**
- SETTING-TABLE BLL CNTXT
  SPELL
- SPREAD 2.191 0.000 3.466
- TABLECLOTH 2.191 0.000
  3.466
- DISTRIBUTE 2.191 0.000
  3.466
- CUTLERY 2.191 0.000 3.466
- POLISH 2.191 0.000 3.466
- GLASSES 2.191 0.000 3.466
- FOLD 2.191 0.000 3.466
- NAPKINS 2.191 0.000 3.466
- LIGHT 2.191 0.000 3.466
- CANDLES 2.191 0.000 3.466

- CLEAR-DESK BLL CNTXT
  SPELL
- OPEN 2.191 0.000 3.466
- FOLDER 2.191 0.000 3.466
- PUT 2.191 0.000 3.466
- FILES 2.191 0.000 3.466
- SHARPEN 2.191 0.000 3.466
- PENCIL 2.191 0.000 3.466
- SORT 2.191 0.000 3.466
- CARDS 2.191 0.000 3.466
- STACK 2.191 0.000 3.466
- ARTICLES 2.191 0.000 3.466

? **(study *prospective*)**
0.5243
? **(inspect-activation)**

SETTING-TABLE	BLL	CNTXT	SPELL
SPREAD	0.558	0.000	2.853
CLEAR-DESK	BLL	CNTXT	SPELL
OPEN	-0.135	0.000	3.466

? **(study *neutral*)**
0.6466
? **(inspect-activation)**

SETTING-TABLE	BLL	CNTXT	SPELL
SPREAD	0.156	0.669	3.178
CLEAR-DESK	BLL	CNTXT	SPELL
OPEN	0.156	0.669	3.178

? **(study *neutral*)**
0.53
? **(inspect-activation)**

SETTING-TABLE	BLL	CNTXT	SPELL
SPREAD	-0.031	1.066	3.423
CLEAR-DESK	BLL	CNTXT	SPELL
OPEN	0.375	0.397	2.912

? **(study *neutral*)**
0.5216000000000001
? **(inspect-activation)**

SETTING-TABLE	BLL	CNTXT	SPELL
SPREAD	-0.164	1.350	3.620
CLEAR-DESK	BLL	CNTXT	SPELL
OPEN	0.529	0.284	2.773

Activation Equation

$$A_i = B_i + \sum_j W_j S_{ji} + \sum_k P_k M_{ki}$$

- Similarities M_{ki} scaled from *max-sim* to *max-dif*
- *max-sim* is positive (1.0) and provides boost of activation
- *max-dif* is very negative (-10.0) and provides inhibition
- *mismatch-penalty* P_k defaults to 1.0
- *goal-activation* W_j defaults to 0.0
- Divide P by n as with W?
- Ultimately (6.0?) P becomes W and M_{ki} become strengths S_{ji}