Modeling Cognitive Control of Working Memory as a Gated Cortical Network

Jared Sylvester, Jim Reggia, and Scott Weems

Depts. of Computer Science & Psychology, UMIACS, CASL
University of Maryland, College Park, MD

Goal: neurobiologically-inspired computational model of working memory and cognitive control

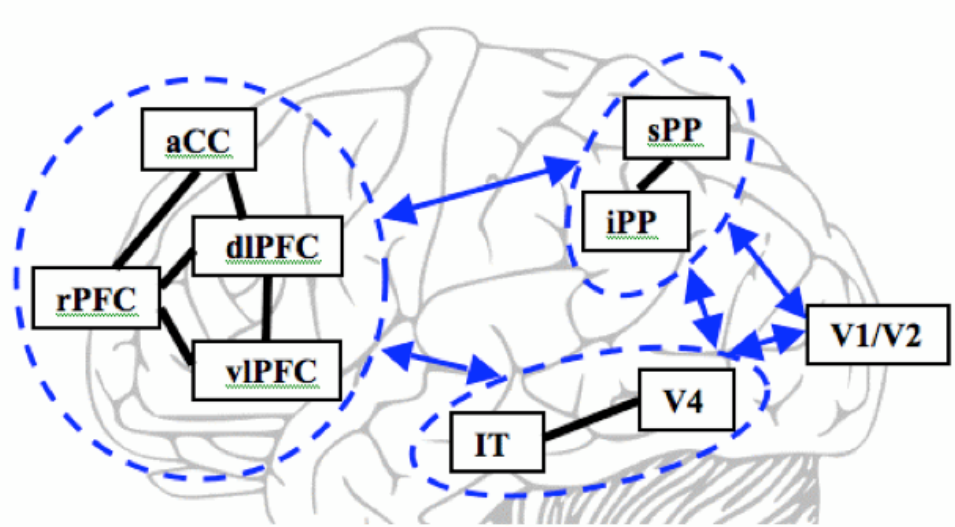
First International Workshop on Cognitive and Working Memory Training
August 25, 2011

Hypothesis

Such a model will help us to …

- interpret results of cognitive & WM training
- relate behavioral data to neurophysiological data
- suggest novel approaches to cognitive training

Top-Level View of Cortical Architecture



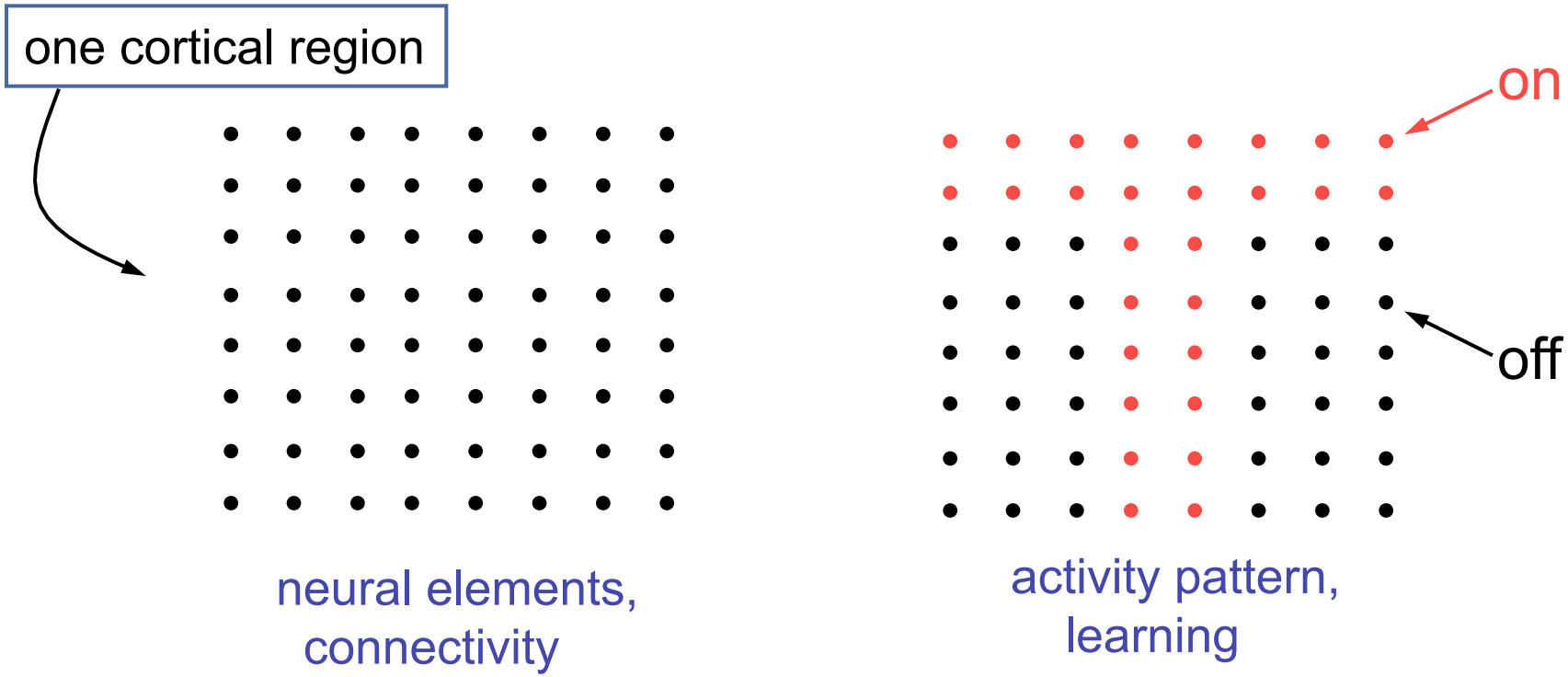
- network of regions & pathways
- regions: - exchange information
  - gate one another
- learns: - WM content
  - task performance

- each region represented as an attractor network

Following: I. Single Region II. Region and Pathway Model

PFC = prefrontal cortex, **CC** = cingulate cortex, **PP** = posterior parietal, **IT** = inferior termporal, **V** = visual

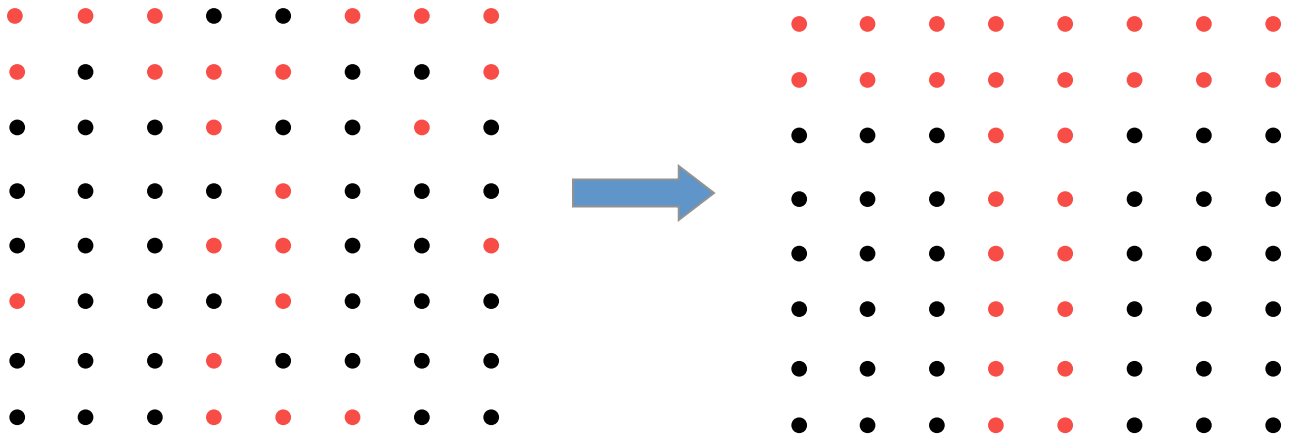
I. Single Region Model



- neural elements with symmetric connections
- store specific patterns (Hebbian learning)
- a basic attractor neural network

Network Behavior

- stored patterns are the attractor states



initial state

final state
fixed

Multiple Patterns Stored Simultaneously

Figure redacted

See: Kohonen, T. *Self-Organization and Associative Memory*. Springer, Berlin. 1984.

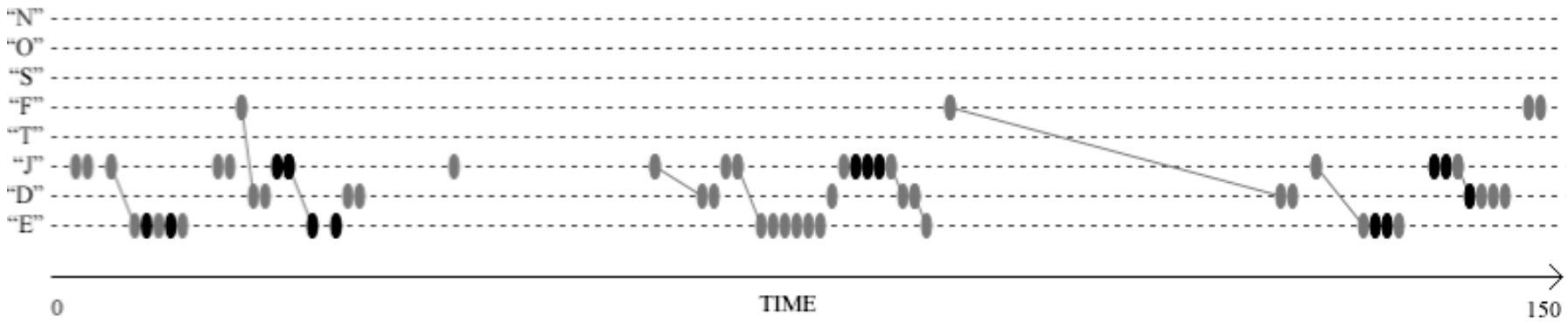
Multiple Patterns Stored Simultaneously

Figure redacted

Missing from basic attractor nets?

- memory decay → add
- multiple simultaneous items in memory → make attractors transient

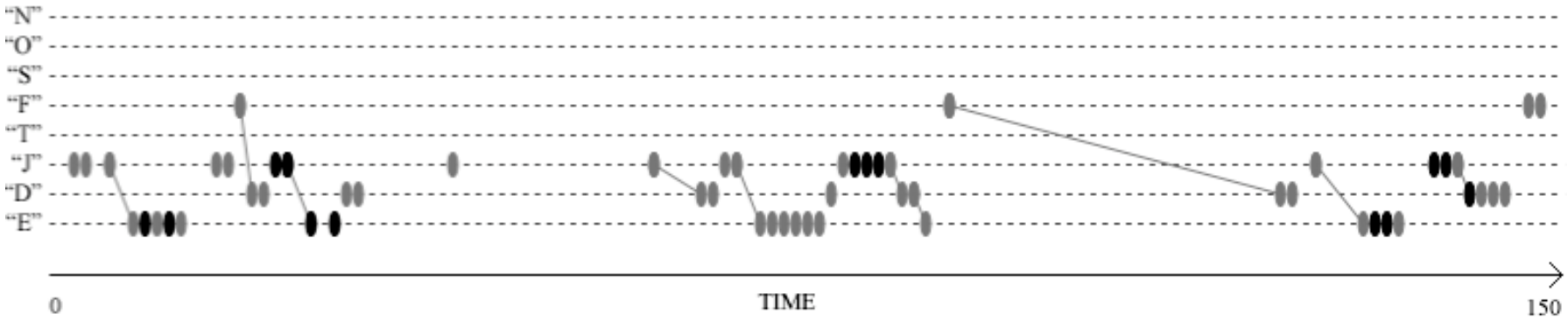
Running Memory Span: Recalled Out of Order



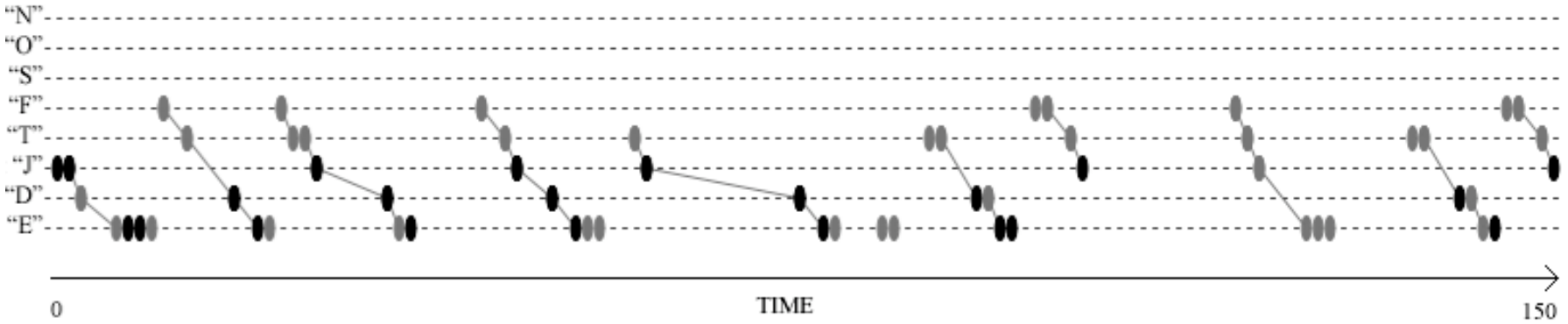
with symmetric weights

Winder R, Reggia J, Weems S, Bunting M. An Oscillatory Hebbian Network Model of Short-Term Memory, Neural Computation, 21, 2009, 741-761.

Running Memory Span: Recalled Out of Order



with symmetric weights



with asymmetric weights

human comparison

Sylvester J, Reggia J, Weems S, Bunting M. A Temporally-Asymmetric Hebbian Network for Sequential Working Memory, Proc. 10th International Conf. on Cognitive Modeling, 2010, pp. 241-246.

II. Regions & Pathways Model

- Moving from single to multiple regions
- Problem:
  - Single-region network has no internal control
- Principles:
  - Network of attractor networks
  - Controlled by gating
  - Learned processing of sequences

Relation to Past Work

- Previous modeling
  - e.g. Anderson (ACT-R); Laird et al. (Soar);
    O'Reilly et al. (PBWM); Dehaene; Polk; Botvinick
- Our approach:
  - Recurrent regional networks, gating, distributed representations, Hebbian learning, learning to do task
- Two types of memory/learning
  - Memory of perceptual stimuli
  - Memory of task procedures

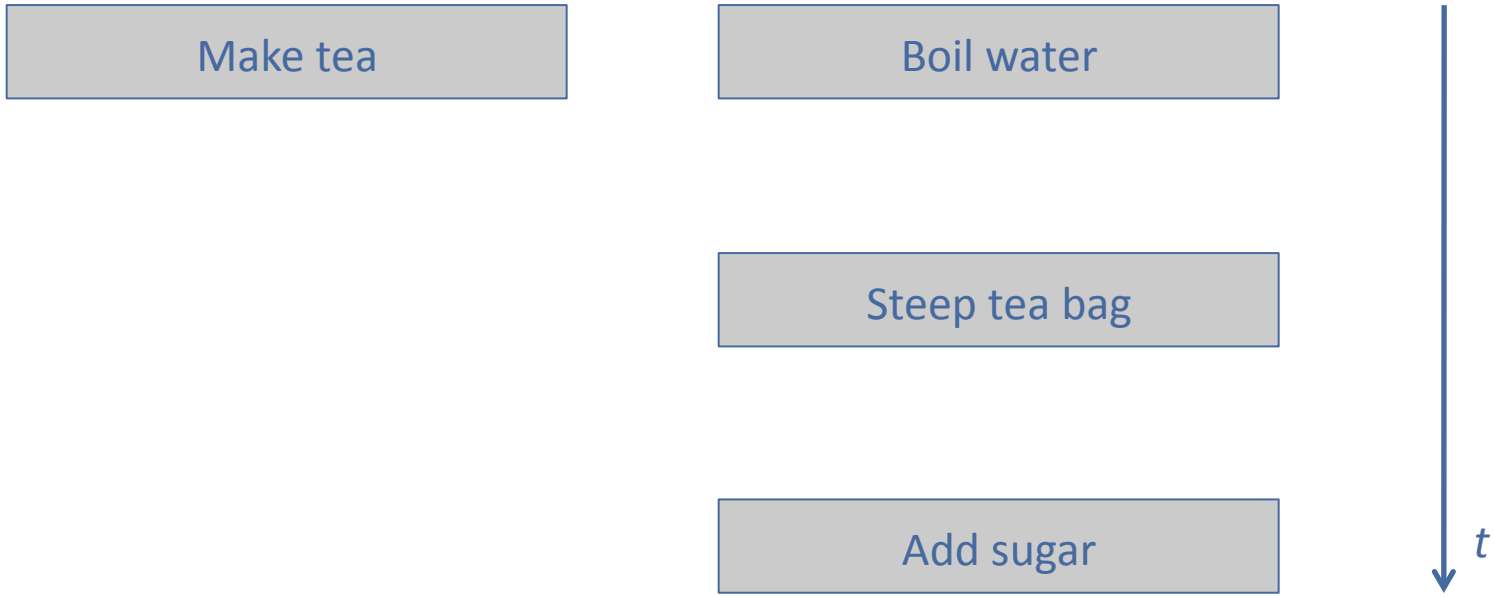
Control Mechanism

- Directs the model by operating gates
- Trained prior to task beginning
- Built around attractor networks
- Core is "instruction memory"

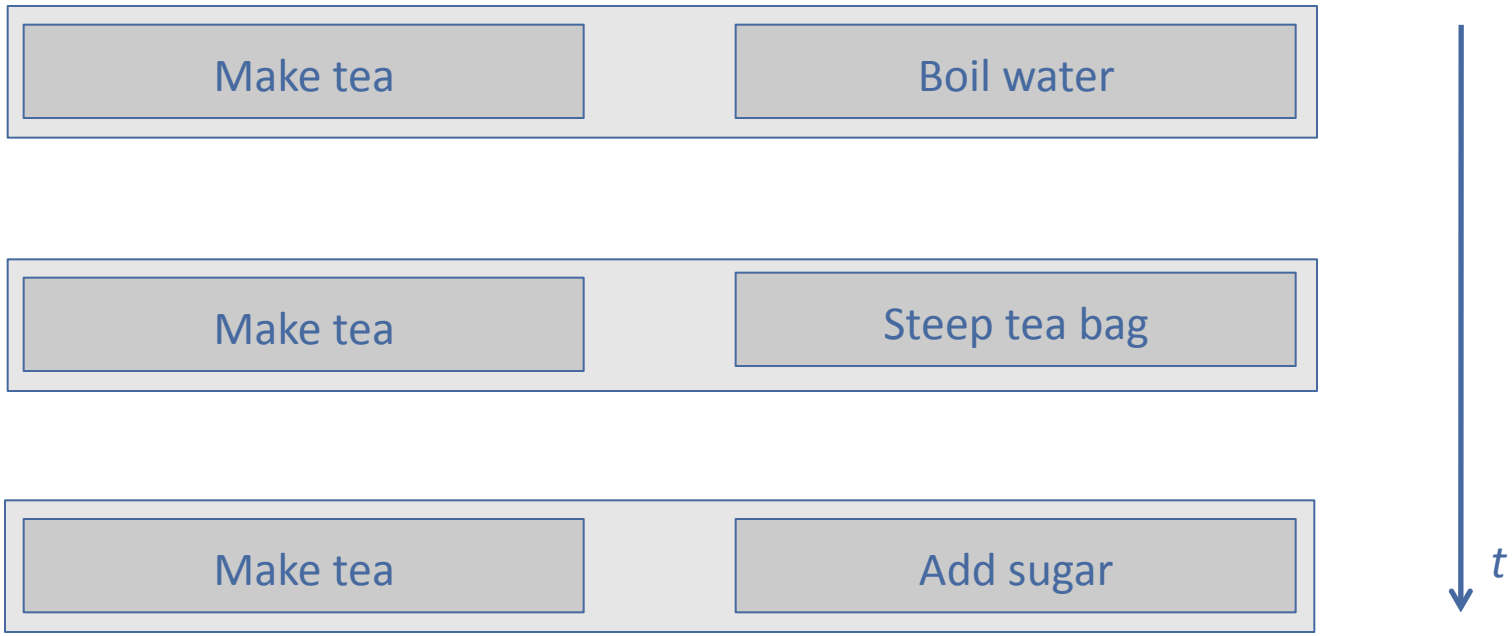
Control Mechanism

- Directs the model by operating gates
- Trained prior to task beginning
- Built around attractor networks
- Core is "instruction memory"
  - Stores sequence of steps to do subtasks
  - Multiple sequences stored simultaneously
  - Divided into cue & response sections

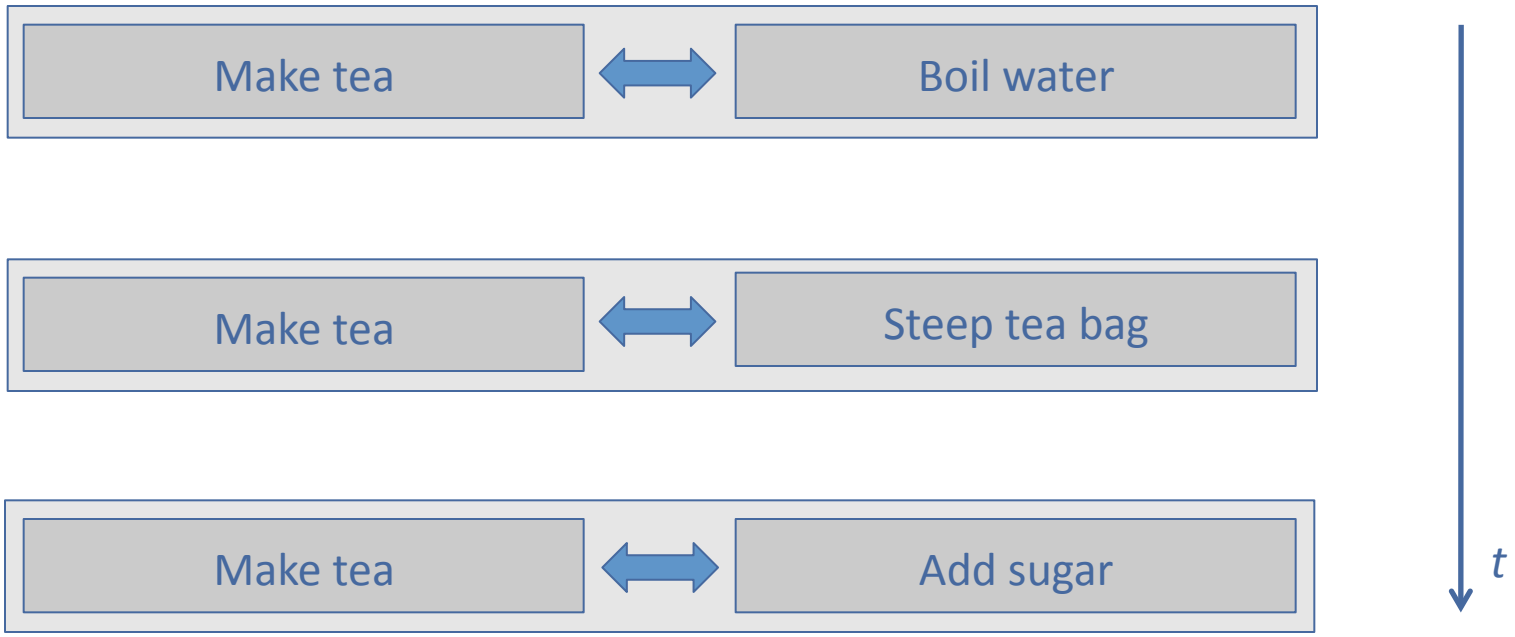
Instruction Memory



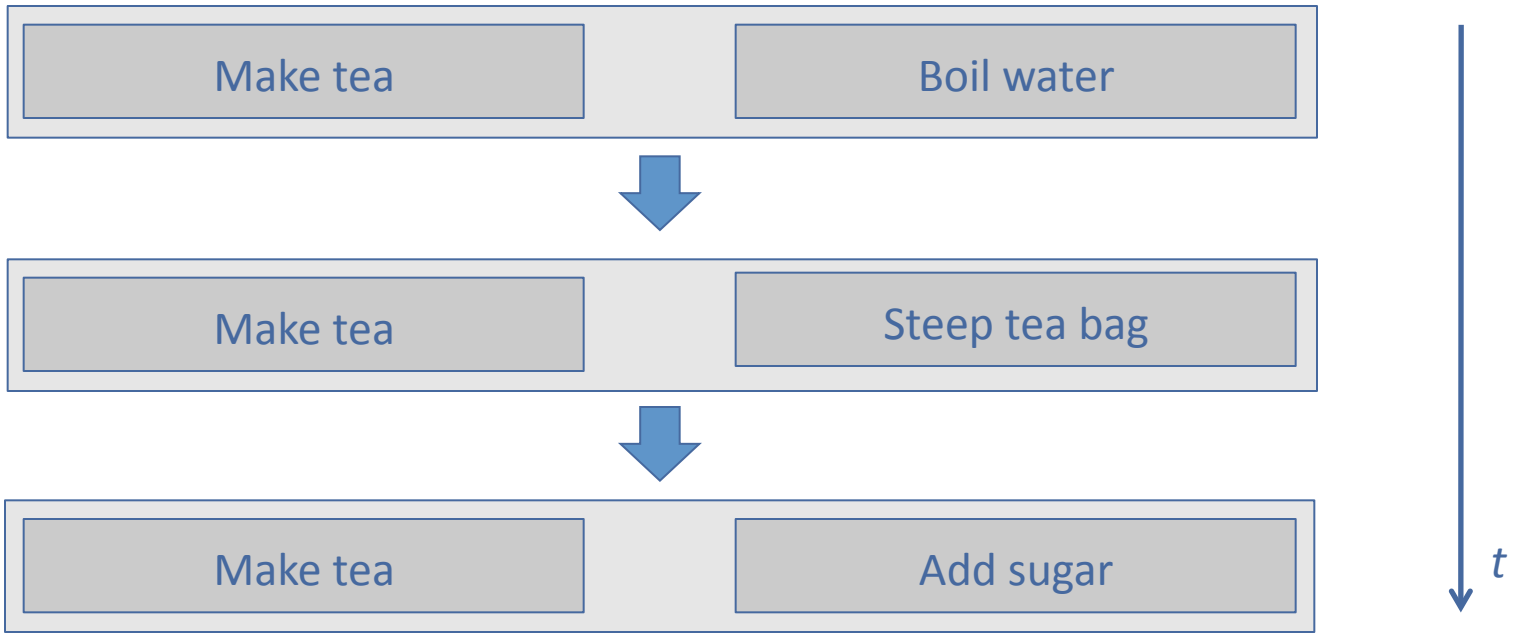
Instruction Memory



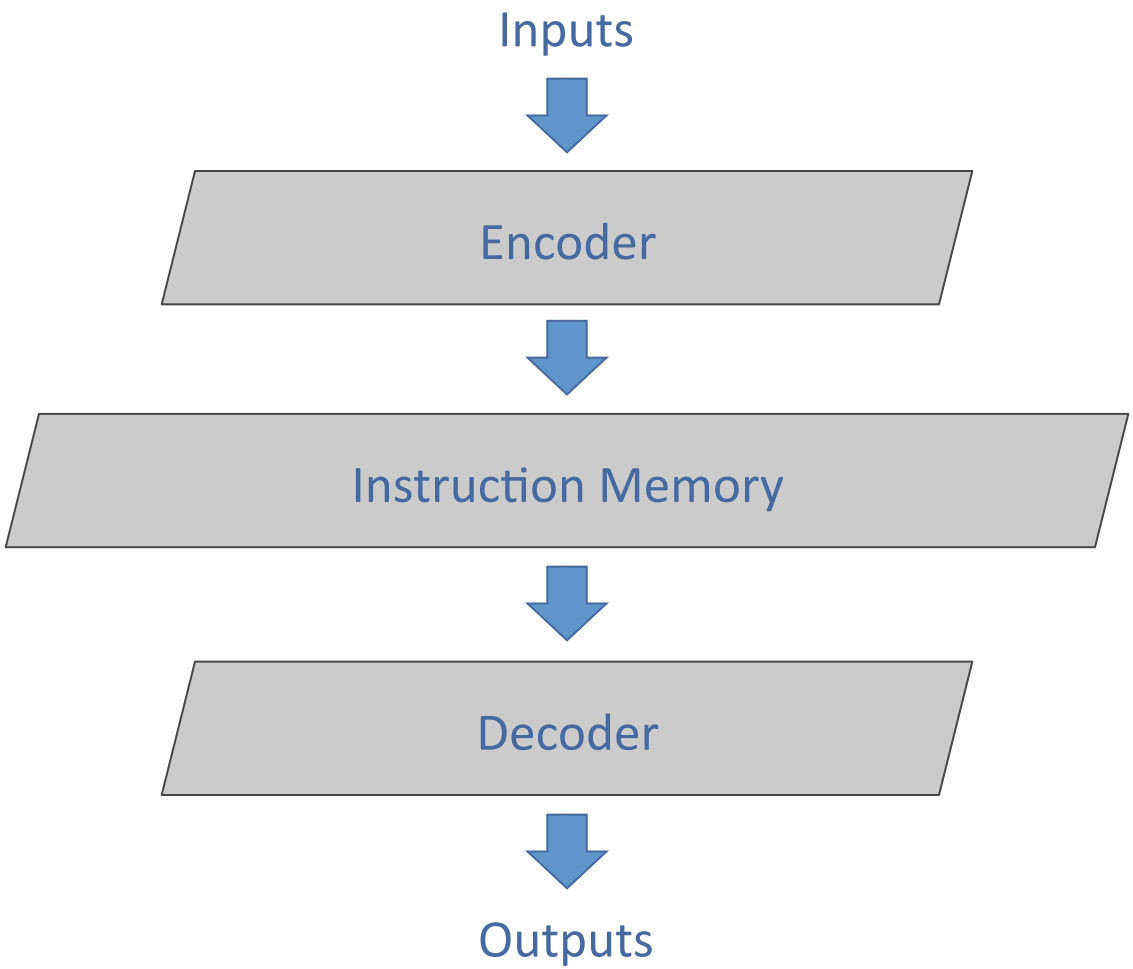
Instruction Memory

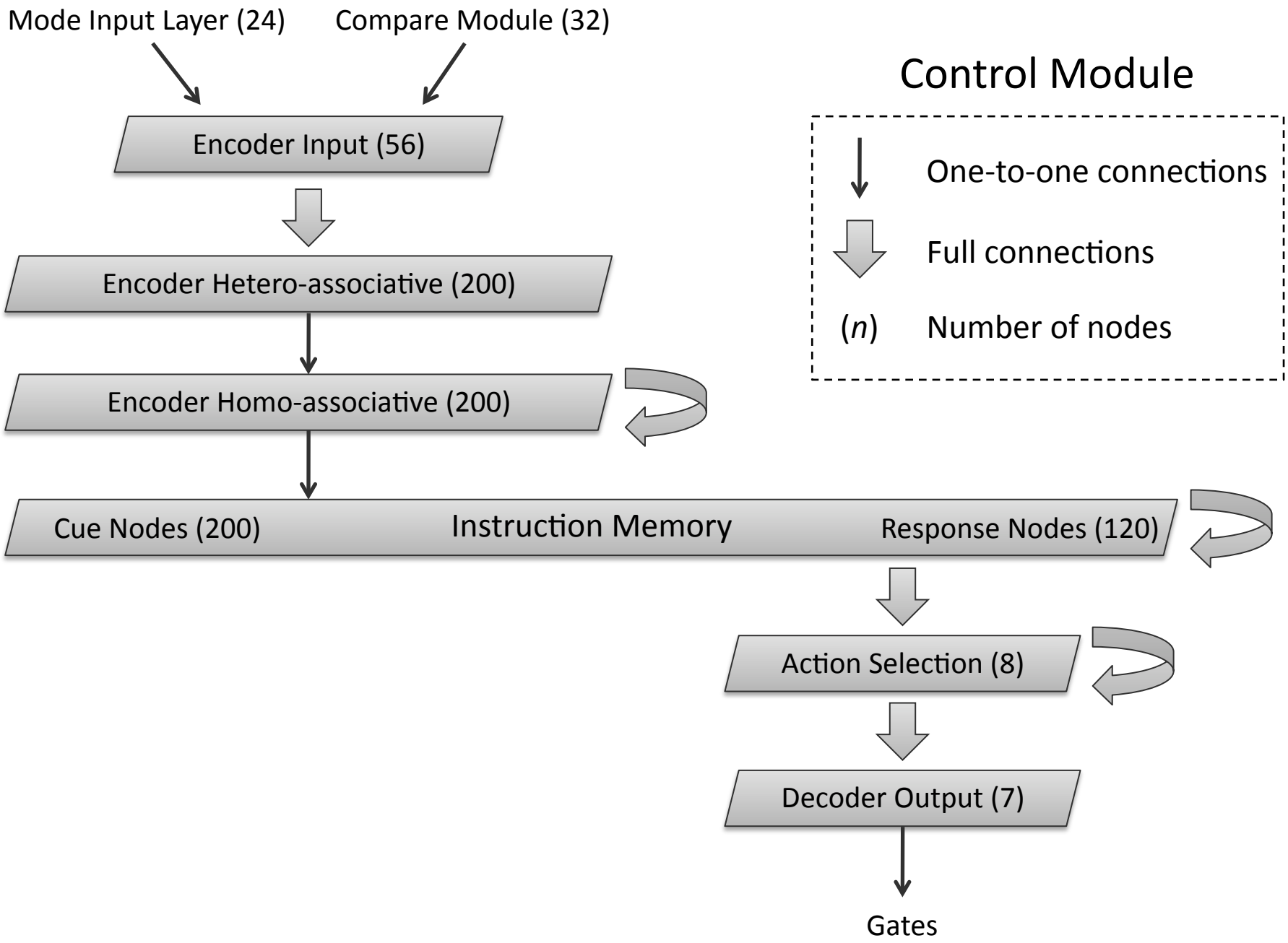


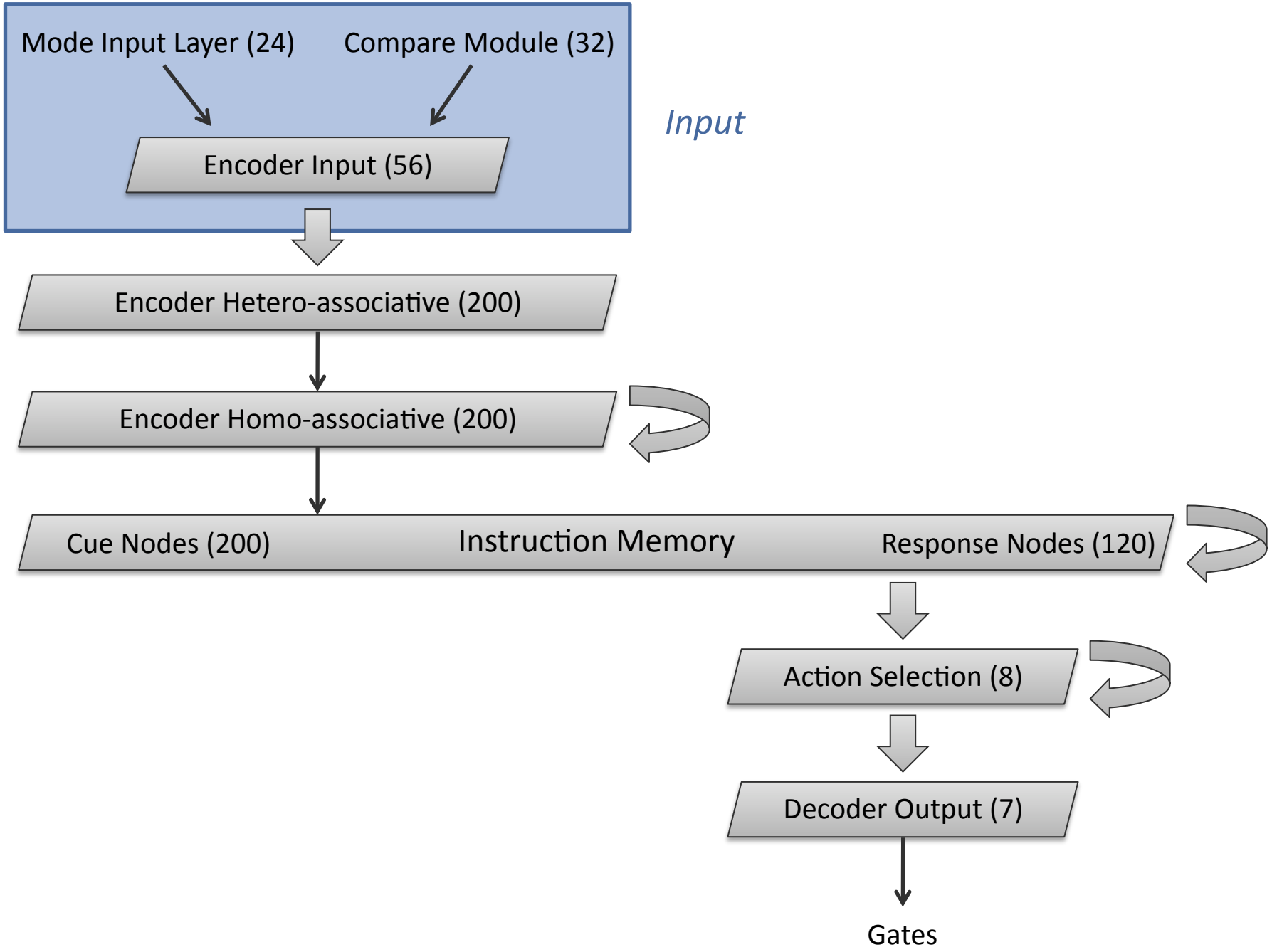
Instruction Memory

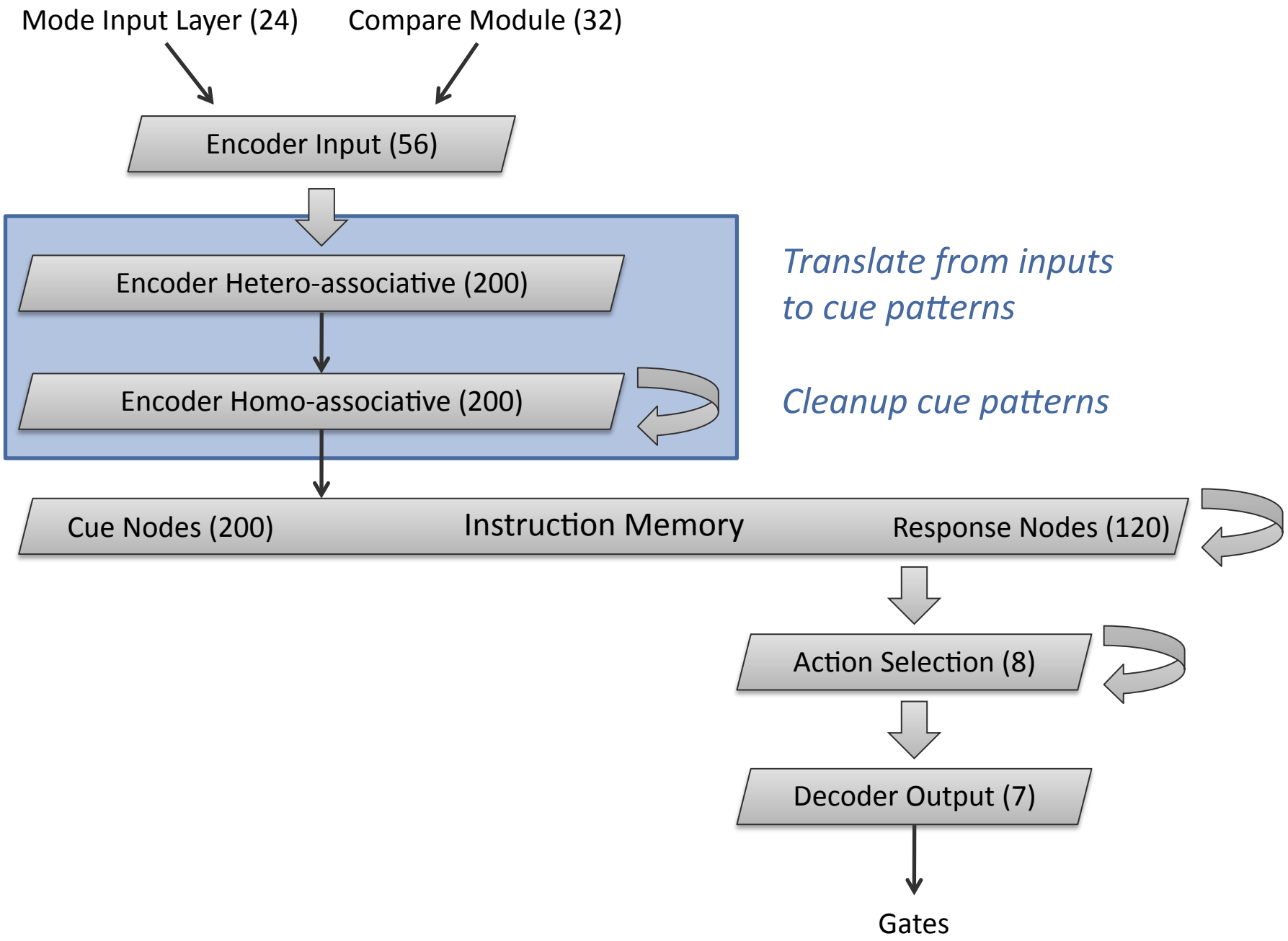


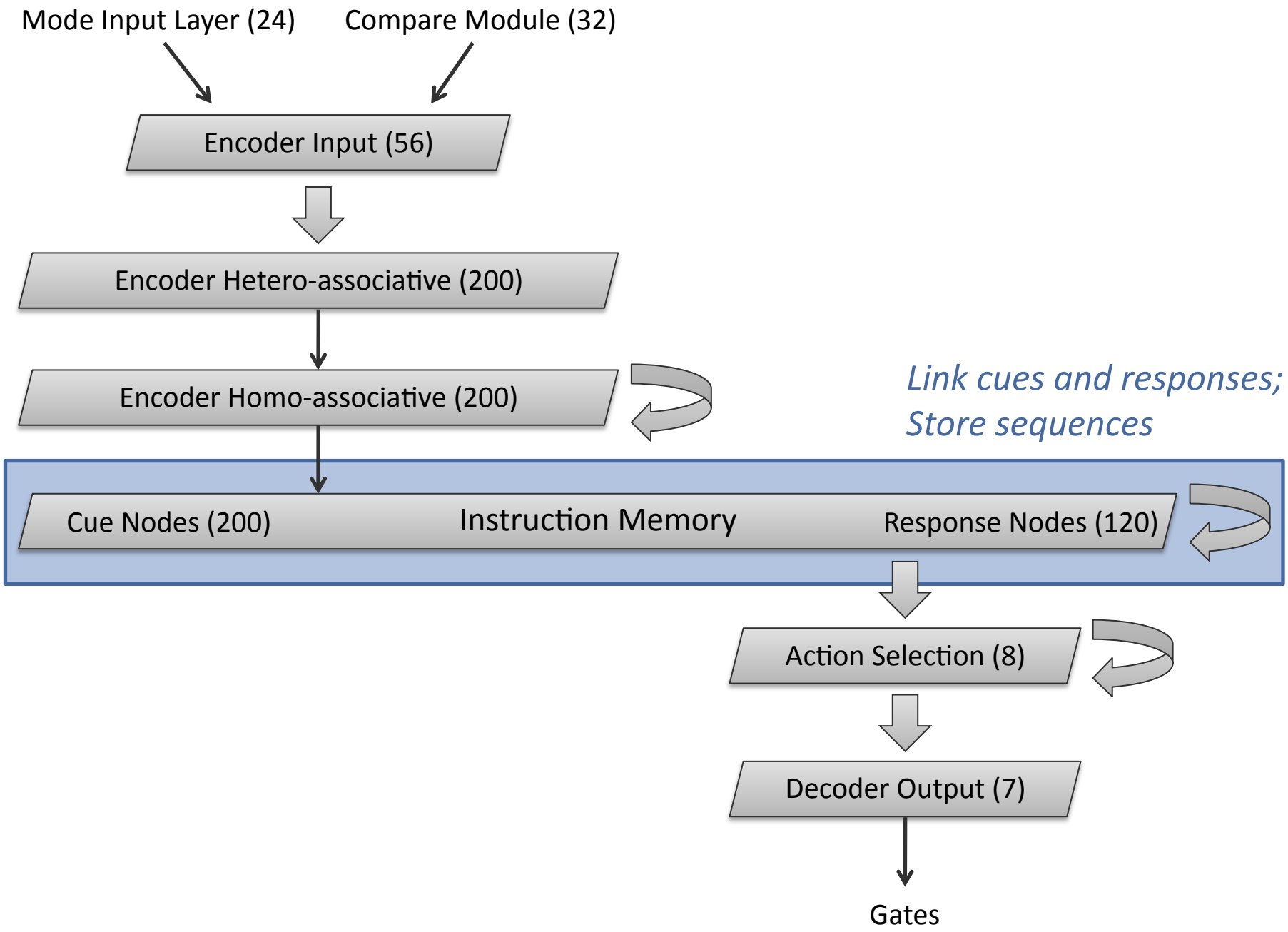
Control Mechanism

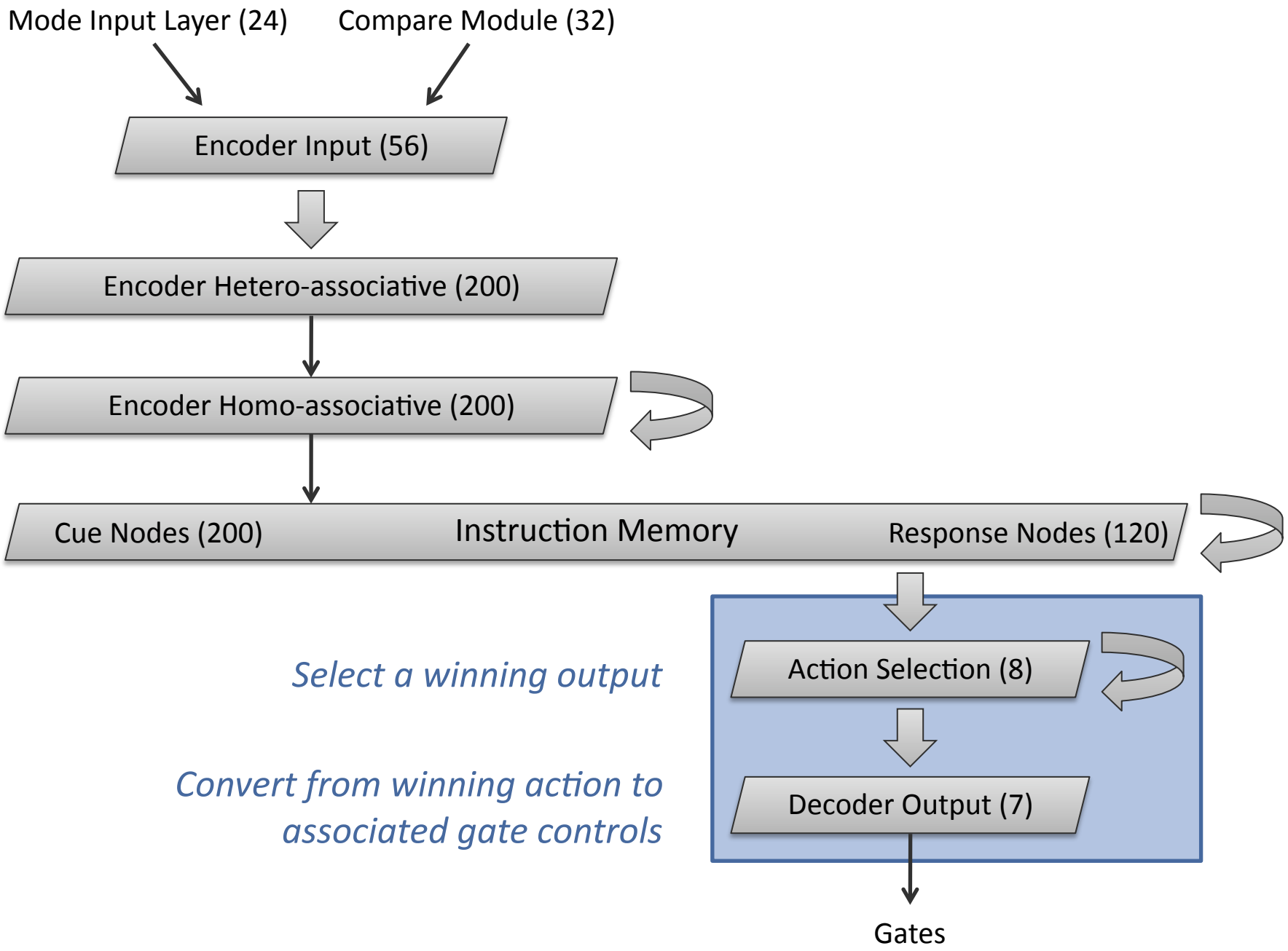



Mode Input Layer (24)
Compare Module (32)
Encoder Input (56)
Encoder Hetero-associative (200)
Encoder Homo-associative (200)
Cue Nodes (200)
Instruction Memory
Response Nodes (120)
Action Selection (8)
Decoder Output (7)
Gates
Control Module
One-to-one connections
Full connections
(n) Number of nodes

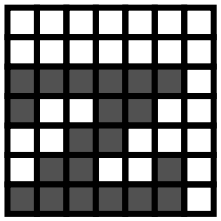

Mode Input Layer (24)
Compare Module (32)
Encoder Input (56)
Input
Encoder Hetero-associative (200)
Encoder Homo-associative (200)
Cue Nodes (200)
Instruction Memory
Response Nodes (120)
Action Selection (8)
Decoder Output (7)
Gates


Mode Input Layer (24)
Compare Module (32)
Encoder Input (56)
Encoder Hetero-associative (200)
Encoder Homo-associative (200)
Translate from inputs to cue patterns
Cleanup cue patterns
Cue Nodes (200)
Instruction Memory
Response Nodes (120)
Action Selection (8)
Decoder Output (7)
Gates

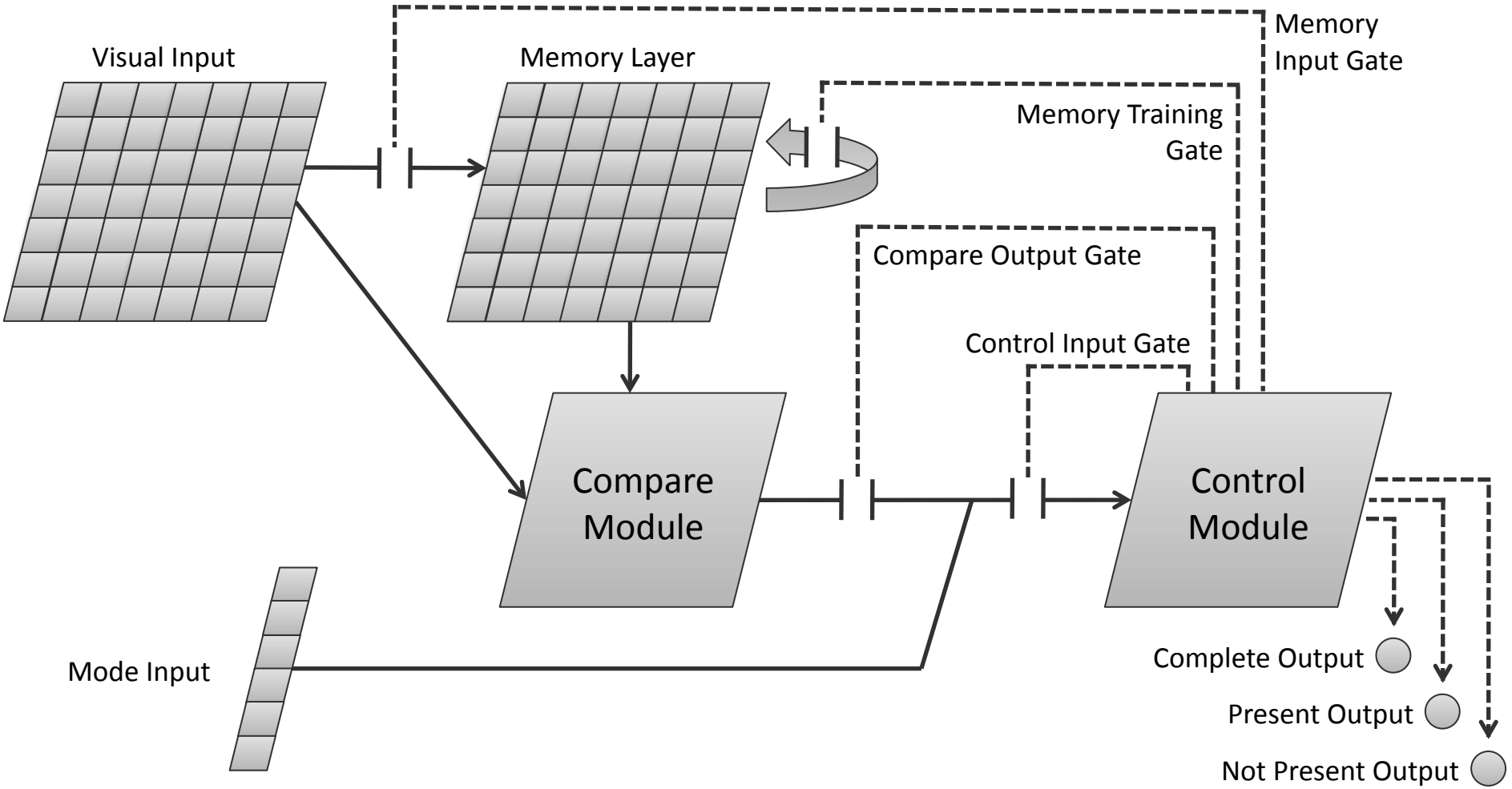

Mode Input Layer (24)
Compare Module (32)
Encoder Input (56)
Encoder Hetero-associative (200)
Encoder Homo-associative (200)
Link cues and responses;
Store sequences
Cue Nodes (200)
Instruction Memory
Response Nodes (120)
Action Selection (8)
Decoder Output (7)
Gates

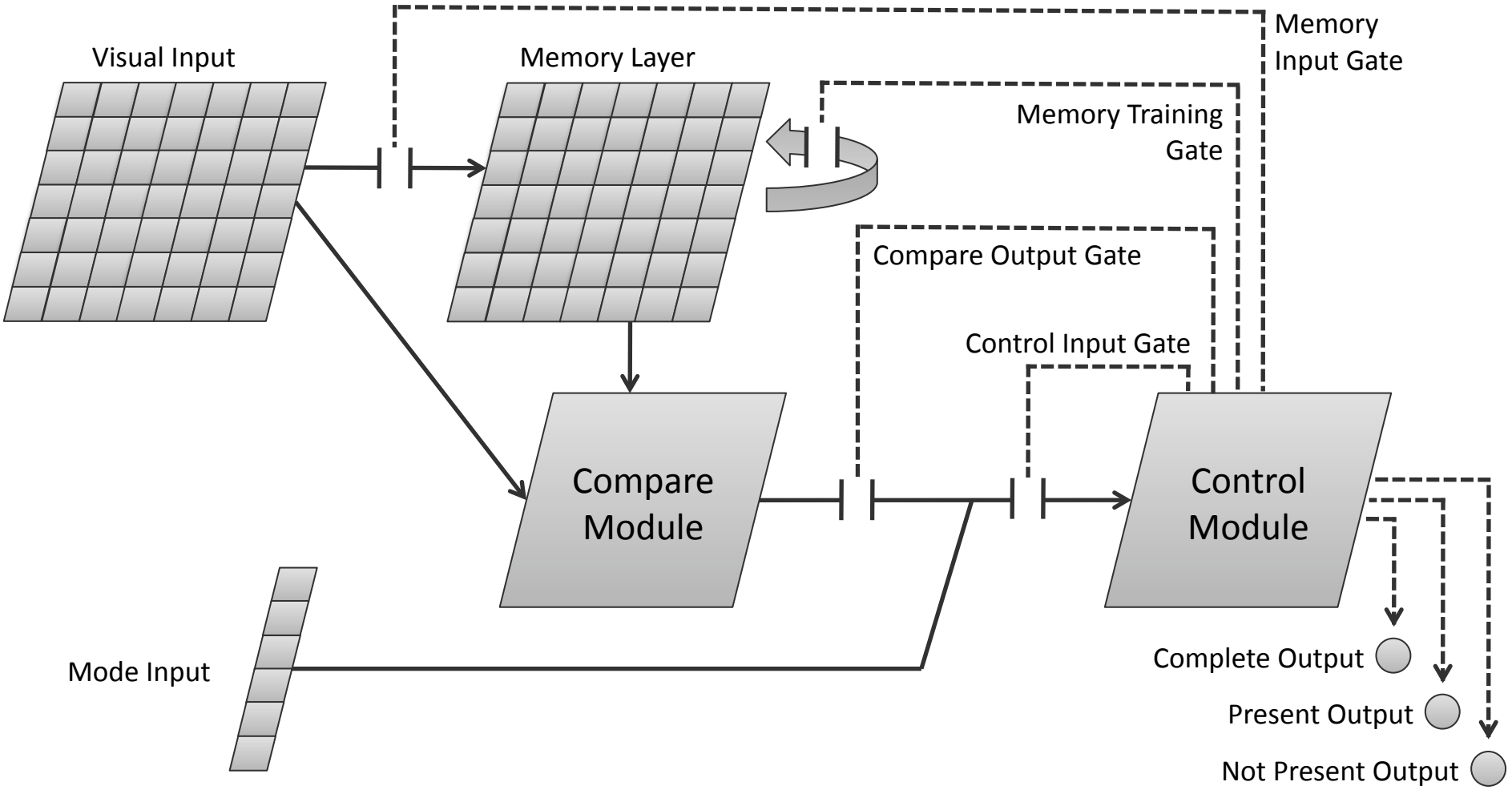

Mode Input Layer (24)
Compare Module (32)
Encoder Input (56)
Encoder Hetero-associative (200)
Encoder Homo-associative (200)
Cue Nodes (200)
Instruction Memory
Response Nodes (120)
Select a winning output
Action Selection (8)
Convert from winning action to associated gate controls
Decoder Output (7)
Gates

Task: "Store/Recognize"

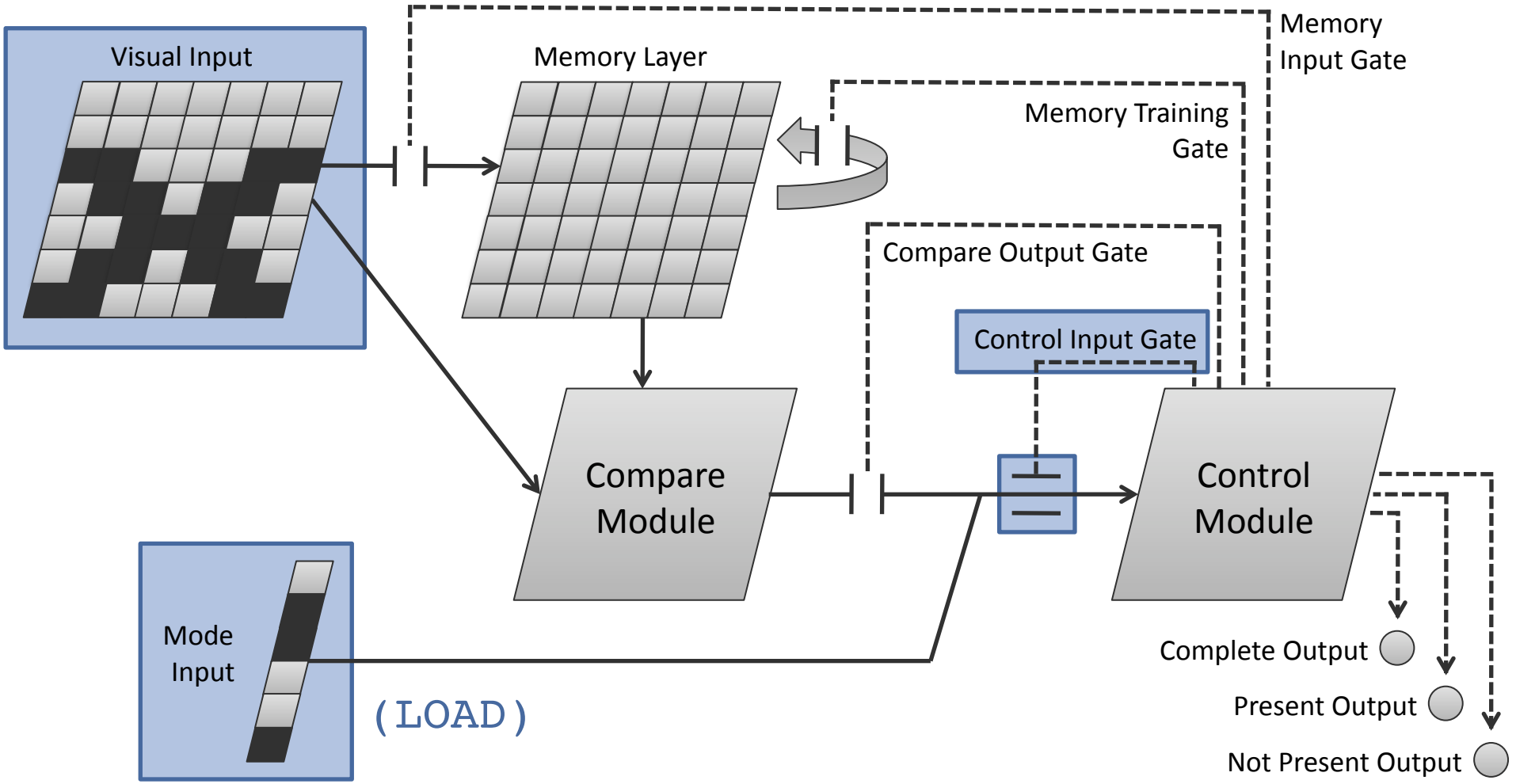
- Visual stimulus 
- Mode input
  - "load" – add visual stimulus to W.M.; output "complete" when done
  - "evaluate" – is visual stimulus in W.M?
    - If so, output "present."
    - If not, add it to W.M. & output "not present"

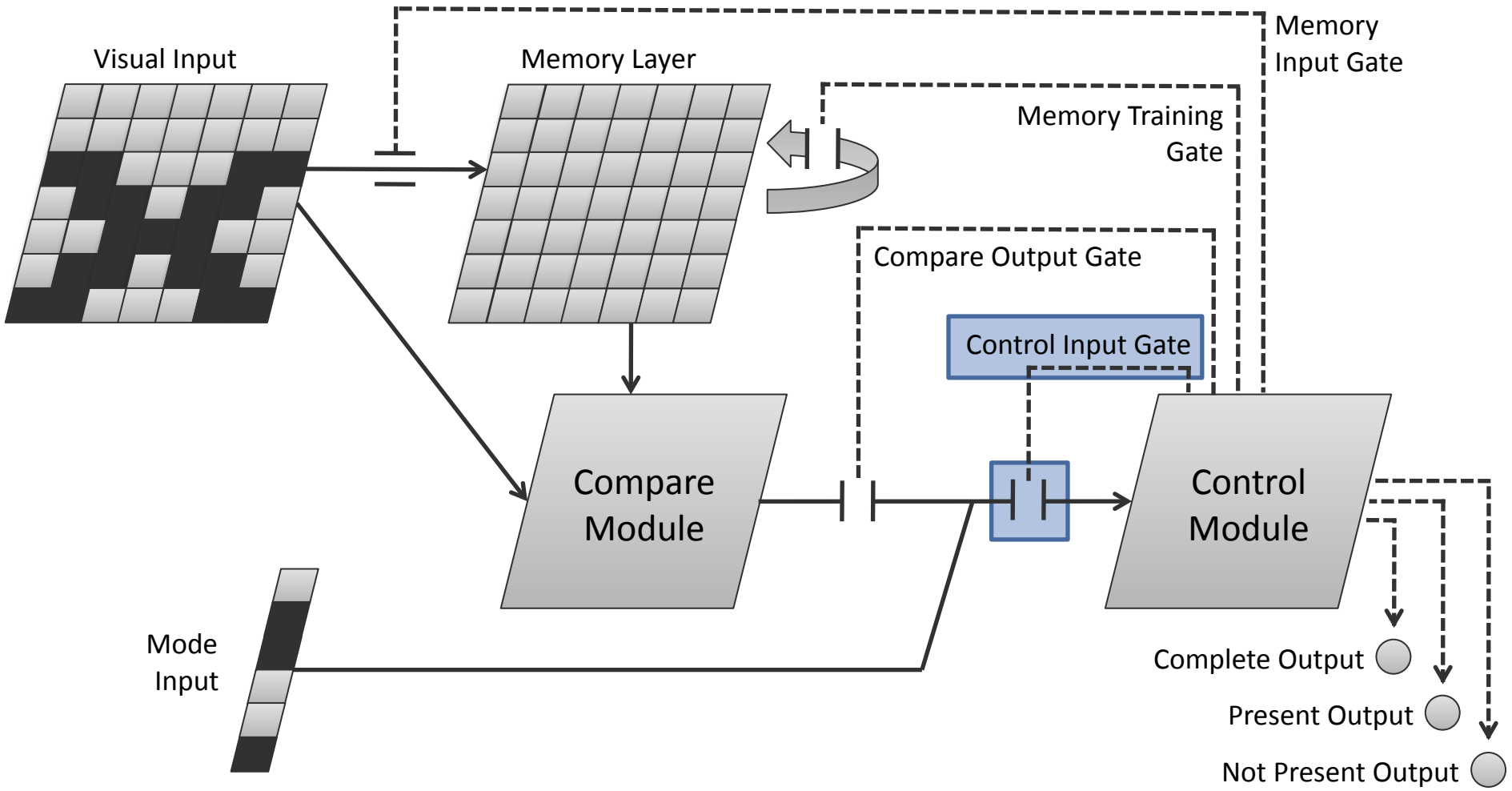
Visual input	A	B	A	X	Y	X
Mode input	load	load	evaluate	evaluate	evaluate	evaluate
Correct response	complete	complete	present	not present	not present	present

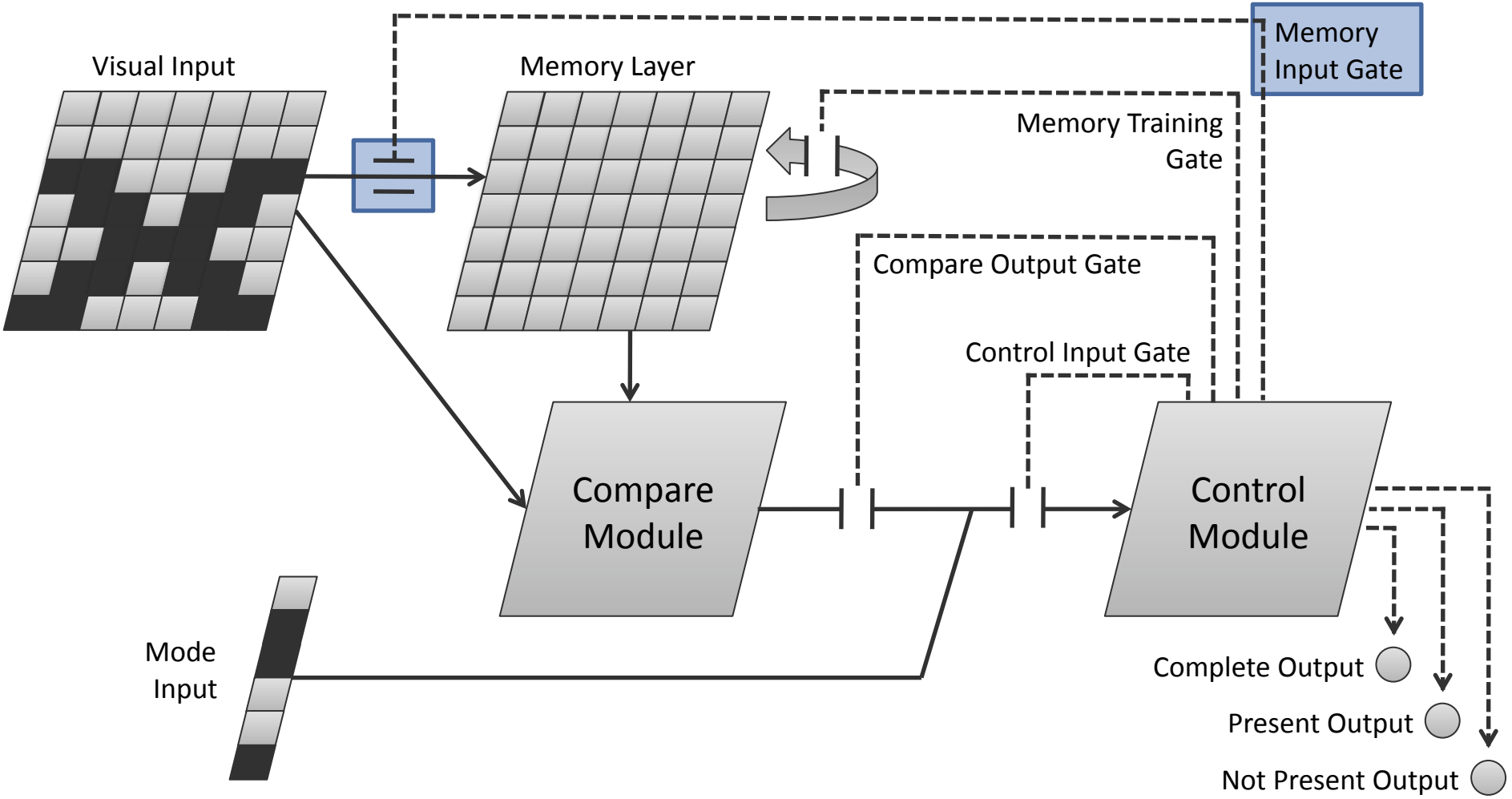

Visual Input
Memory Layer
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode Input
Complete Output
Present Output
Not Present Output

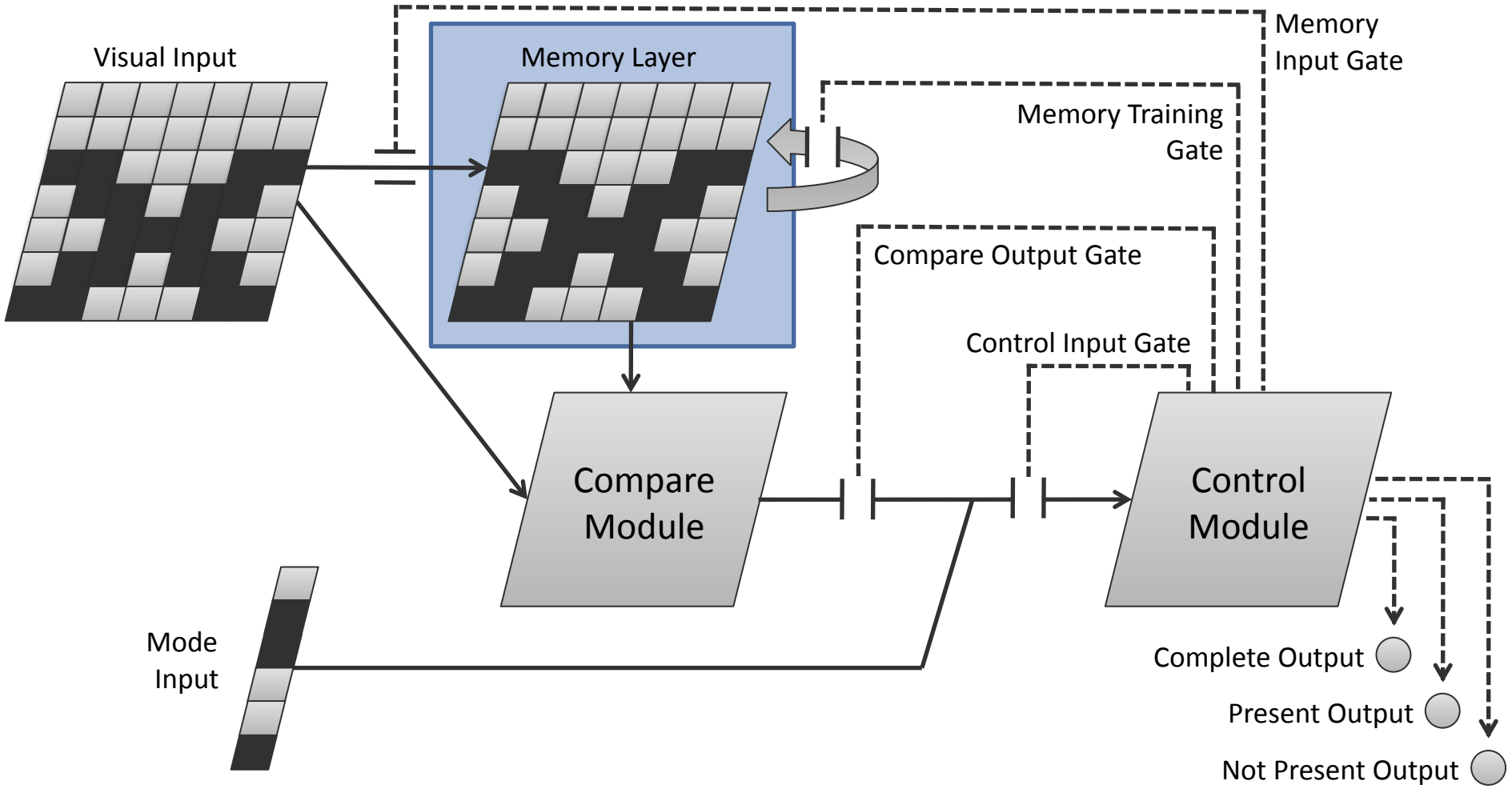


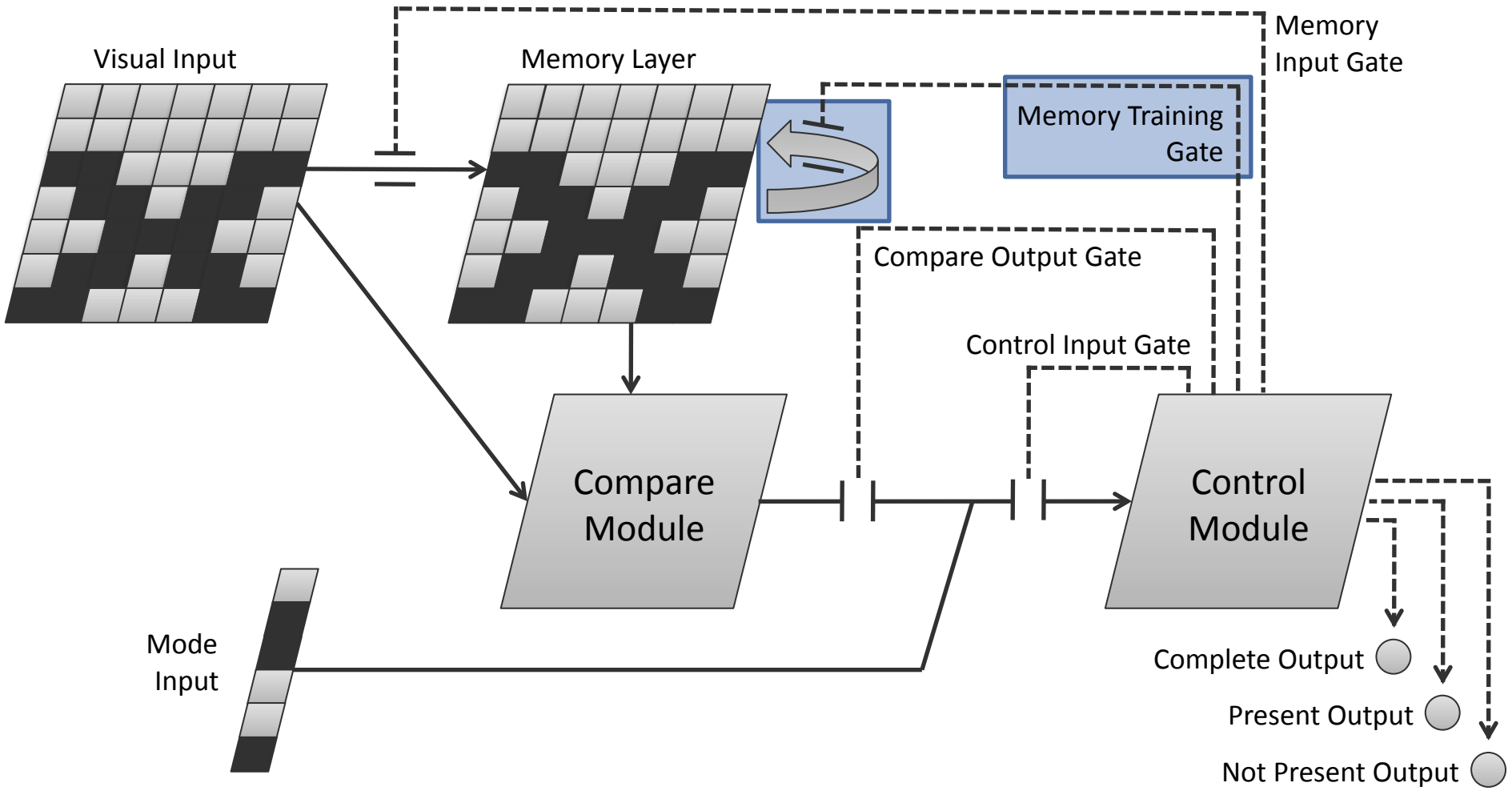
How to add item to working memory

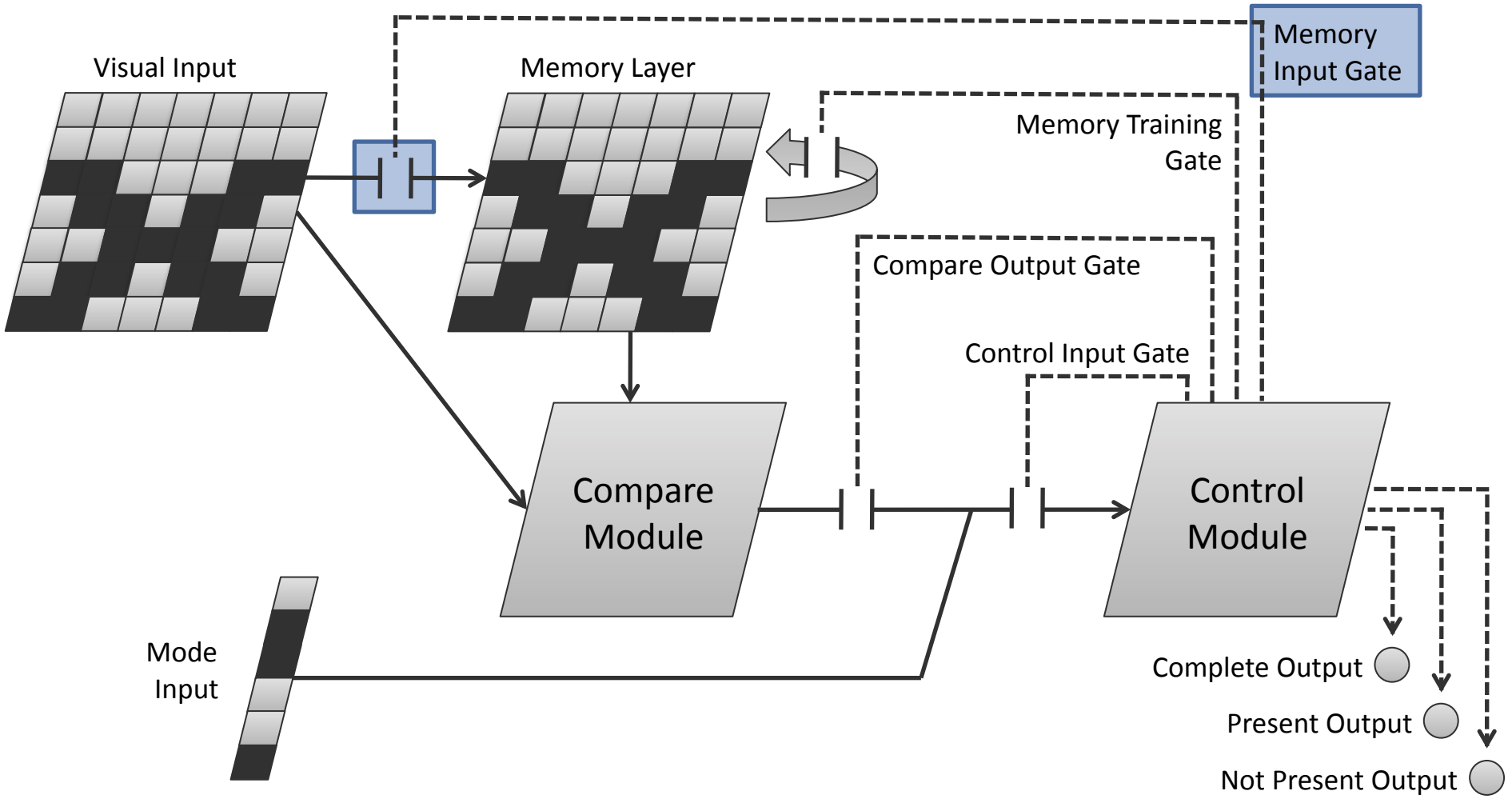

Visual Input
Memory Layer
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
(LOAD)
Complete Output
Present Output
Not Present Output

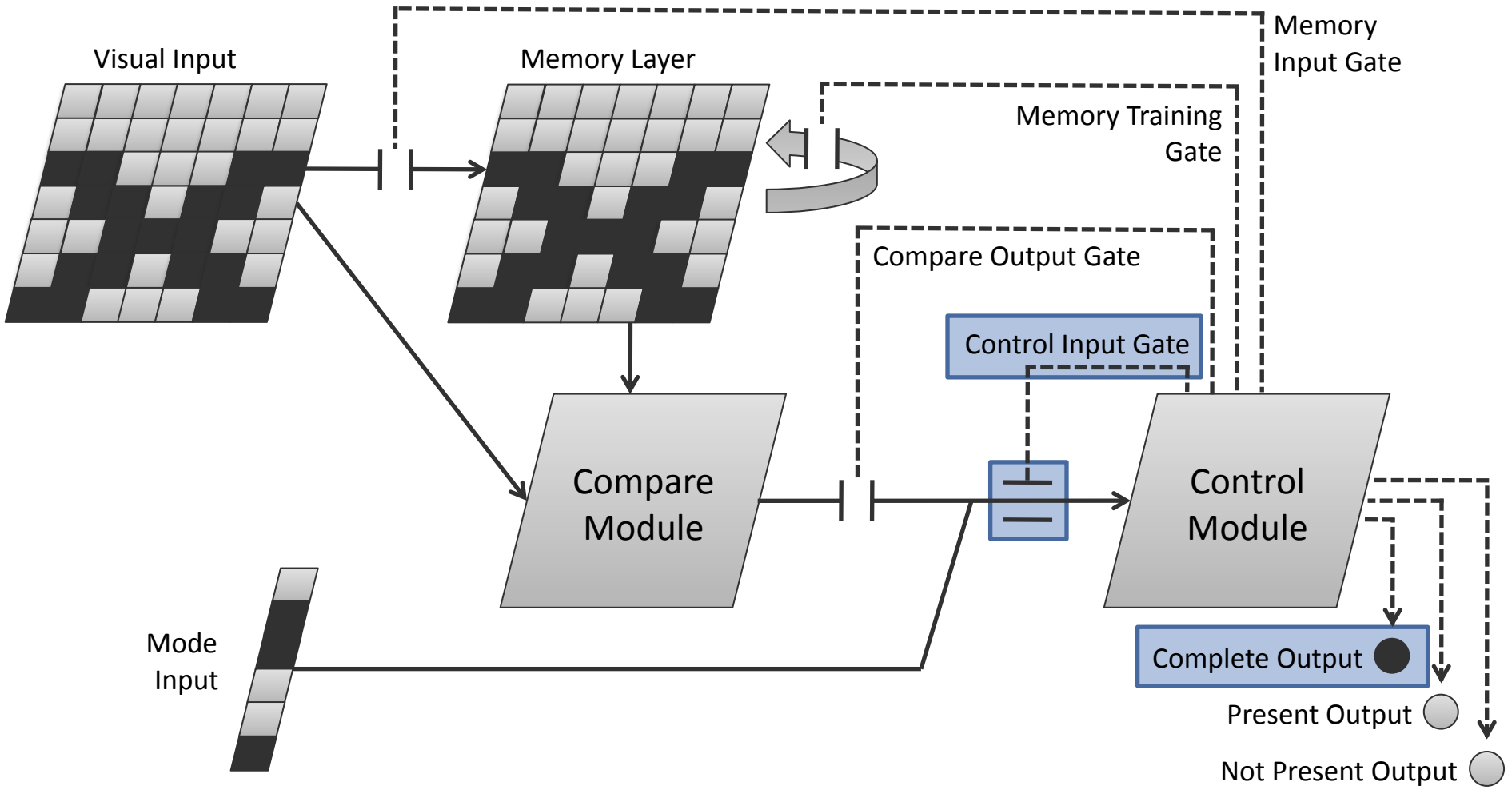

Visual Input
Memory Layer
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
Complete Output
Present Output
Not Present Output


Visual Input
Memory Layer
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
Complete Output
Present Output
Not Present Output

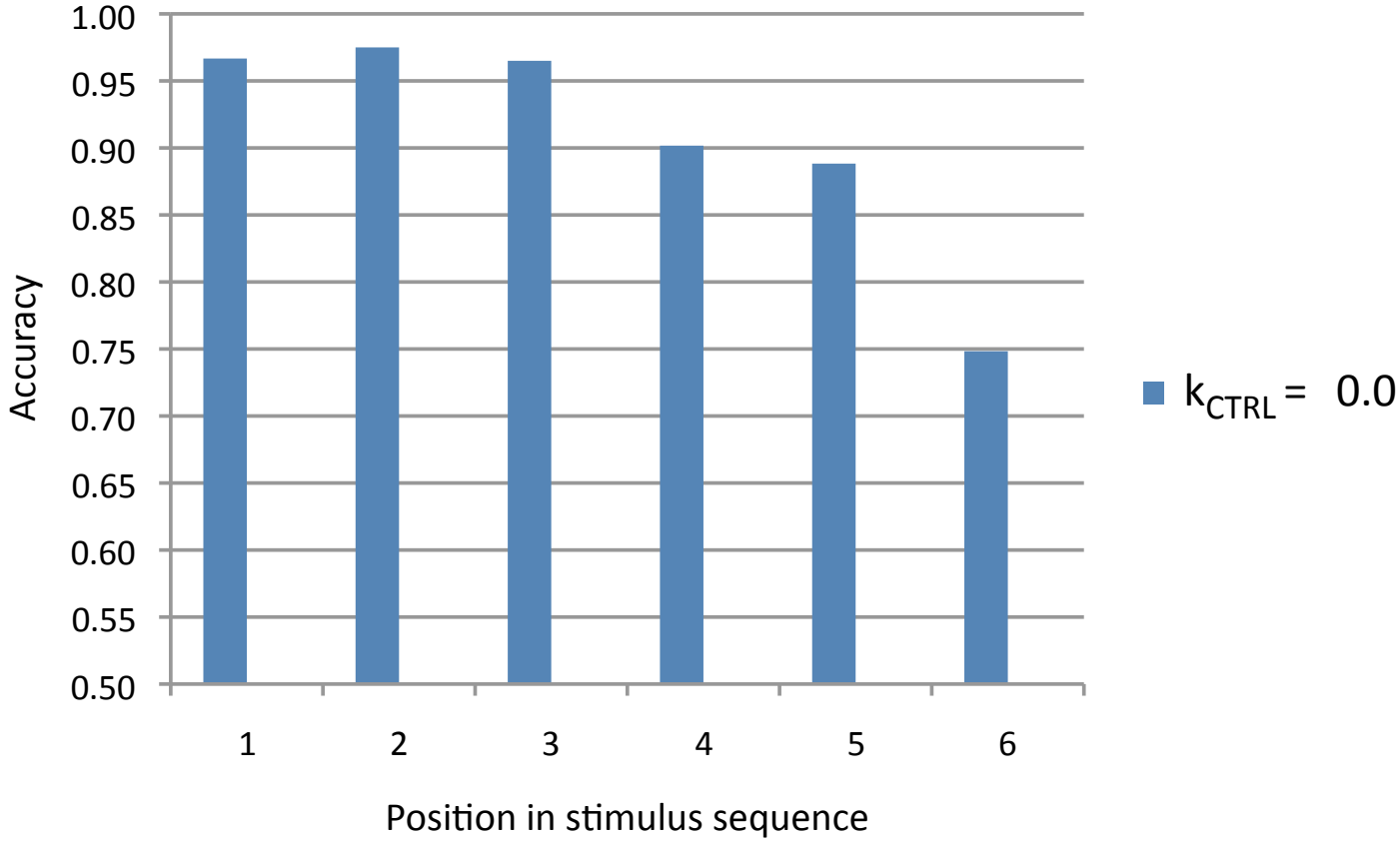

Visual Input
Memory Layer
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
Complete Output
Present Output
Not Present Output


Visual Input
Memory Layer
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
Complete Output
Present Output
Not Present Output


Visual Input
Memory Layer
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
Complete Output
Present Output
Not Present Output


Visual Input
Memory Layer
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
Complete Output
Present Output
Not Present Output

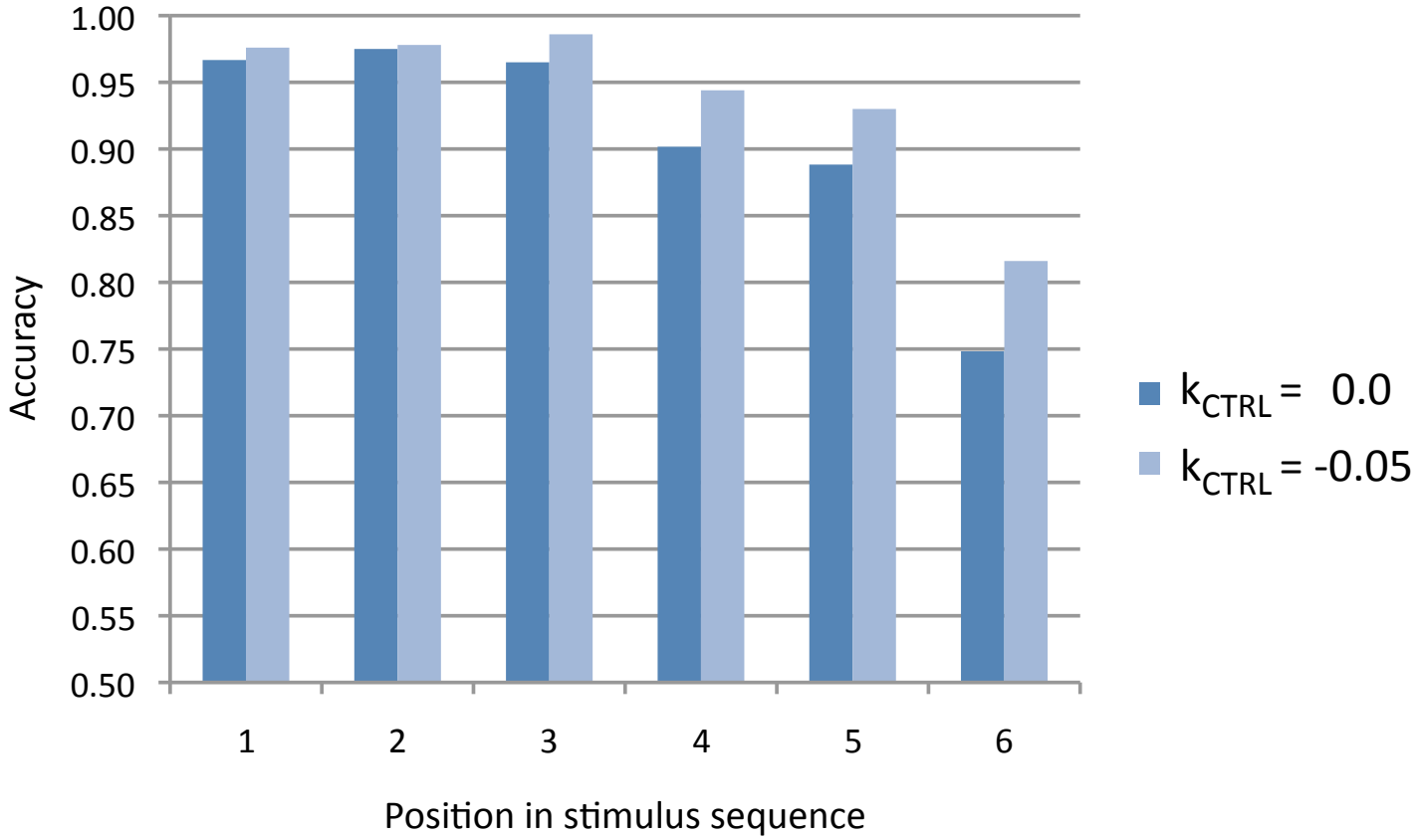
Accuracy varying decay in Instruction Memory



k_{CTRL} is the "decay" rate in the controller's instruction memory layer

Sylvester J, Reggia J, Weems S. Cognitive Control as a Gated Cortical Net, *Proc. 2nd Int'l Conf. on Biologically-Inspired Cognitive Architectures*, IOP Press, 2011, in press.

Accuracy varying decay in Instruction Memory

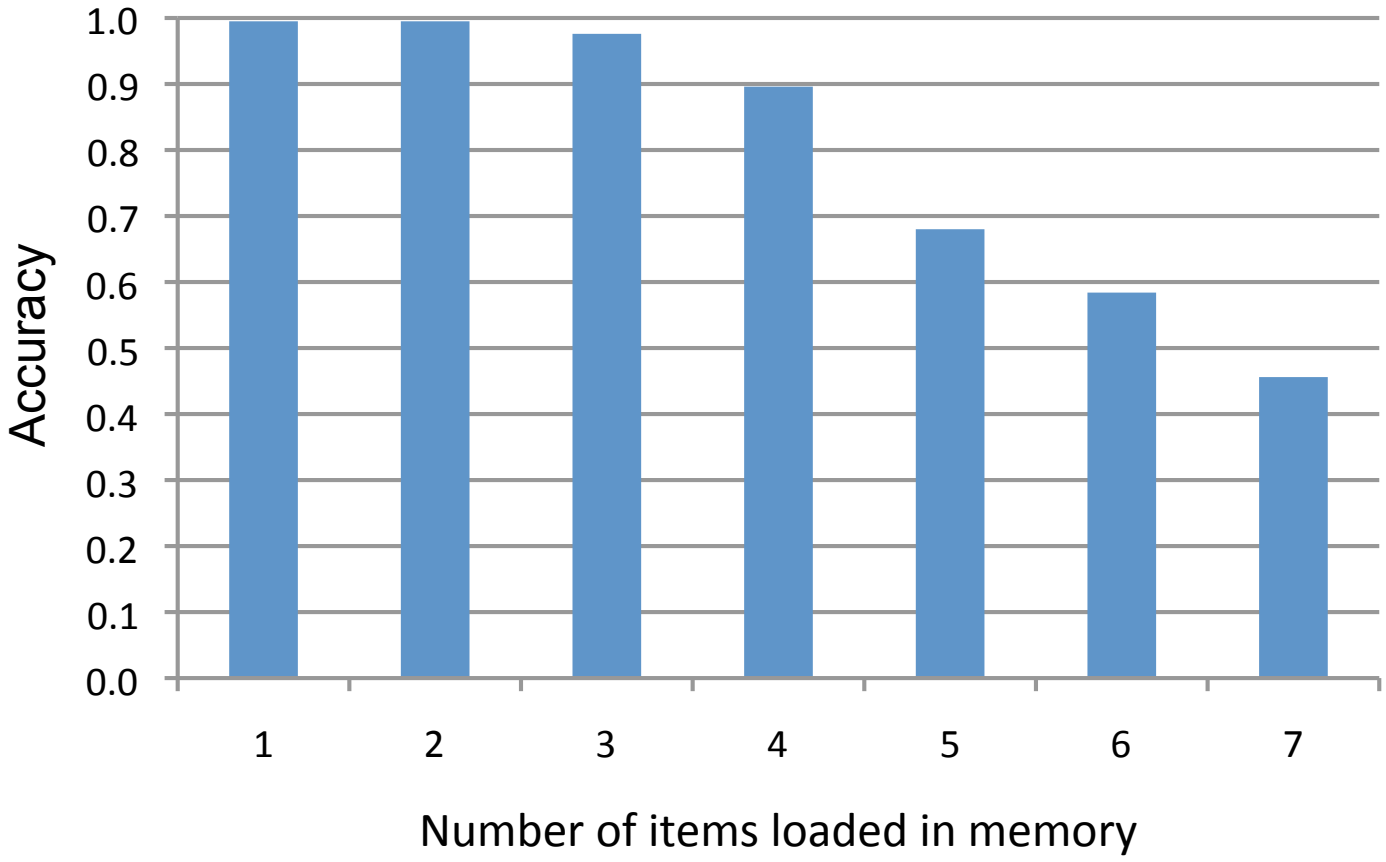


k_{CTRL} is the "decay" rate in the controller's instruction memory layer

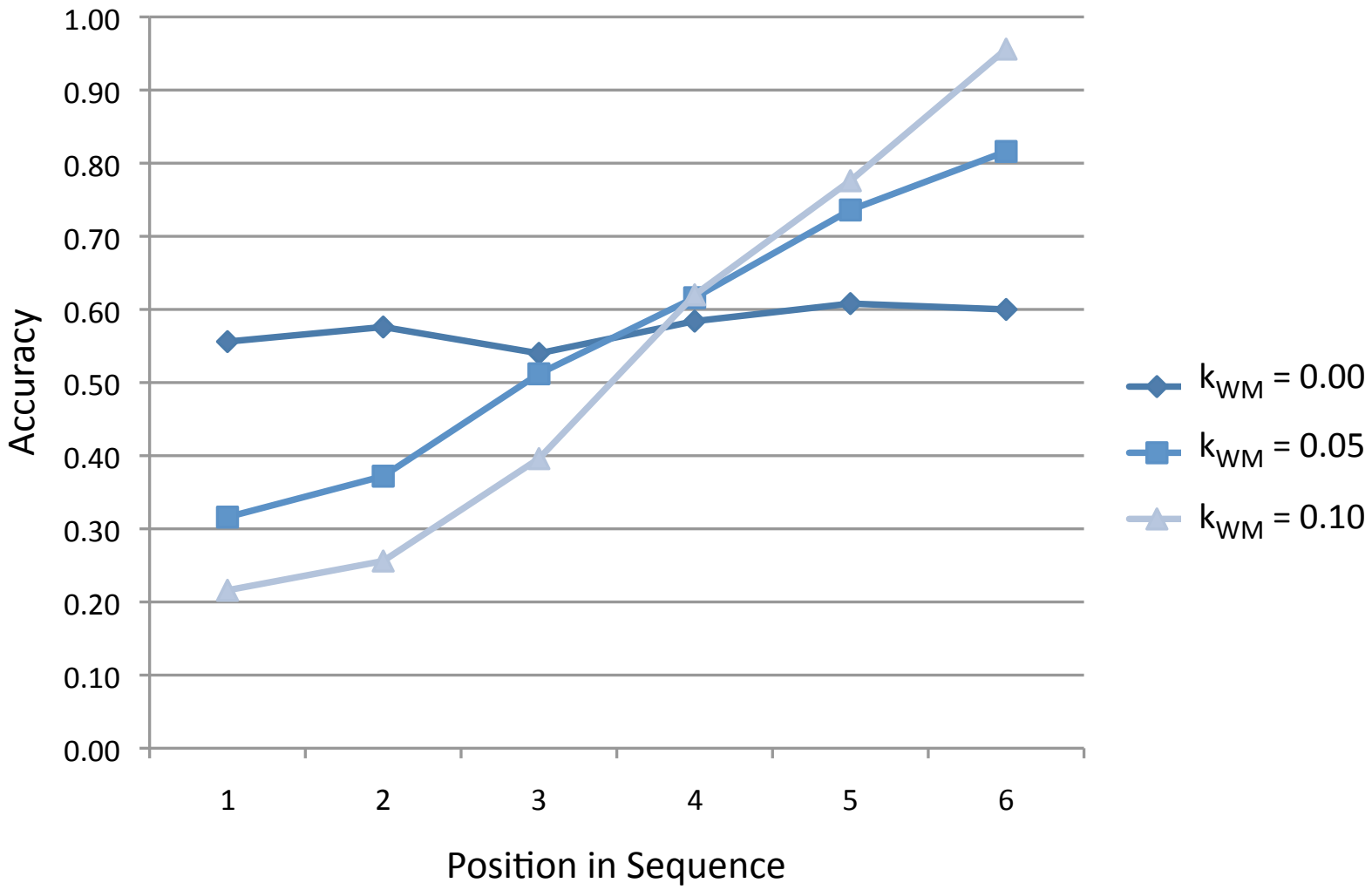
Negative decay (i.e. gain) performs better

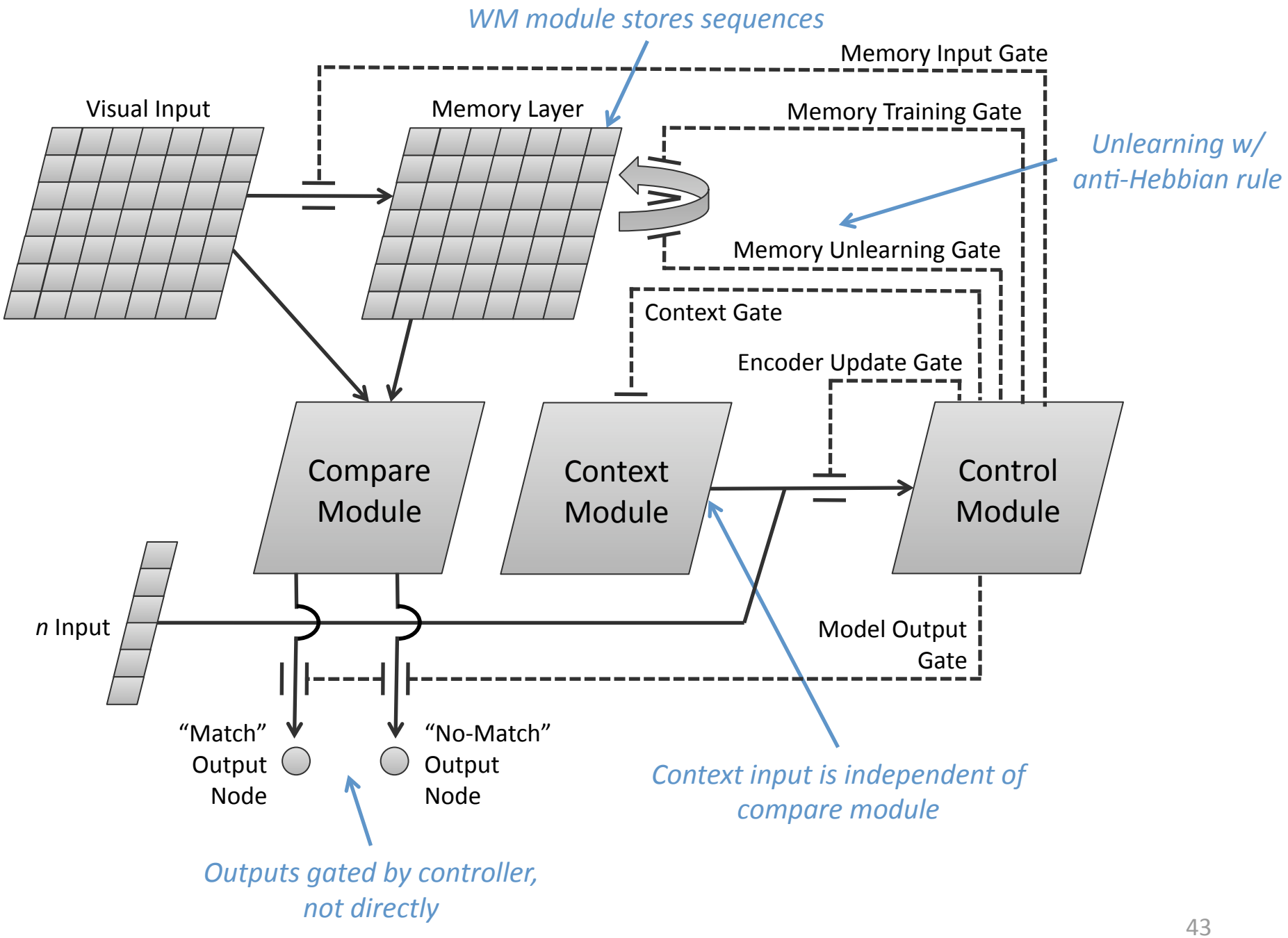
Sylvester J, Reggia J, Weems S. Cognitive Control as a Gated Cortical Net, *Proc. 2nd Int'l Conf. on Biologically-Inspired Cognitive Architectures*, IOP Press, 2011, in press.

Fraction Recalled vs. Number of Items Stored



Effect of W.M. decay on recall




WM module stores sequences
Memory Input Gate
Visual Input
Memory Layer
Memory Training Gate
Unlearning w/
anti-Hebbian rule
Memory Unlearning Gate
Context Gate
Encoder Update Gate
Compare
Module
Context
Module
Control
Module
n Input
Model Output
Gate
"Match"
Output
Node
"No-Match"
Output
Node
Context input is independent of
compare module
Outputs gated by controller,
not directly

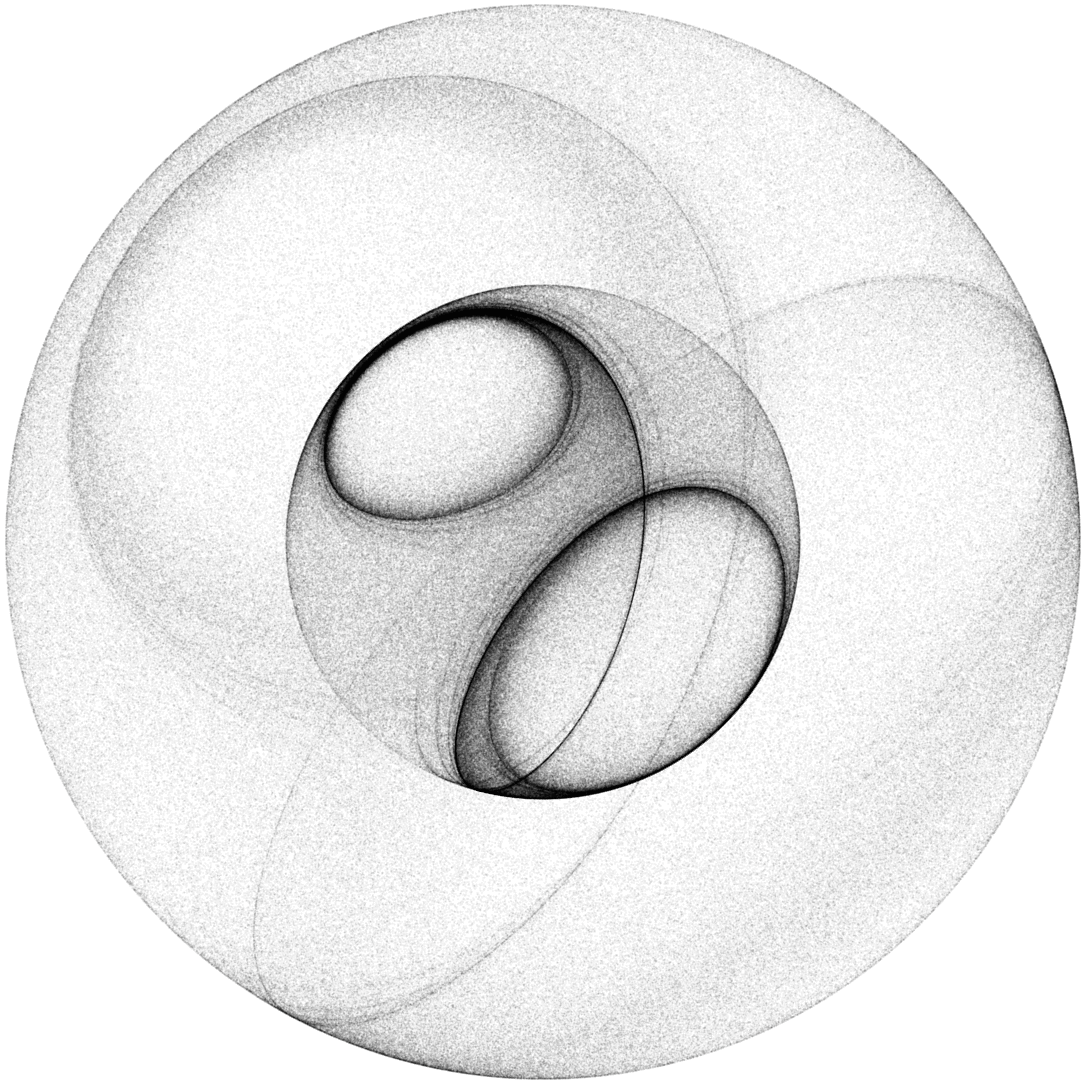
Details redacted pending publication

n-Back Results

Results redacted pending publication

Relating Model & Training Studies

- This is a framework on which to test ideas about training
  - Adaptive controller outputs during task
  - Adjust threshold in compare module
  - Change decay params to handle longer sequences
  - Because relatively few changes to architecture between tasks, is it easier to look at transfer?



"Lormalized" (single frame), Reza Ali, 2010
Attractor-based algorithmic animation