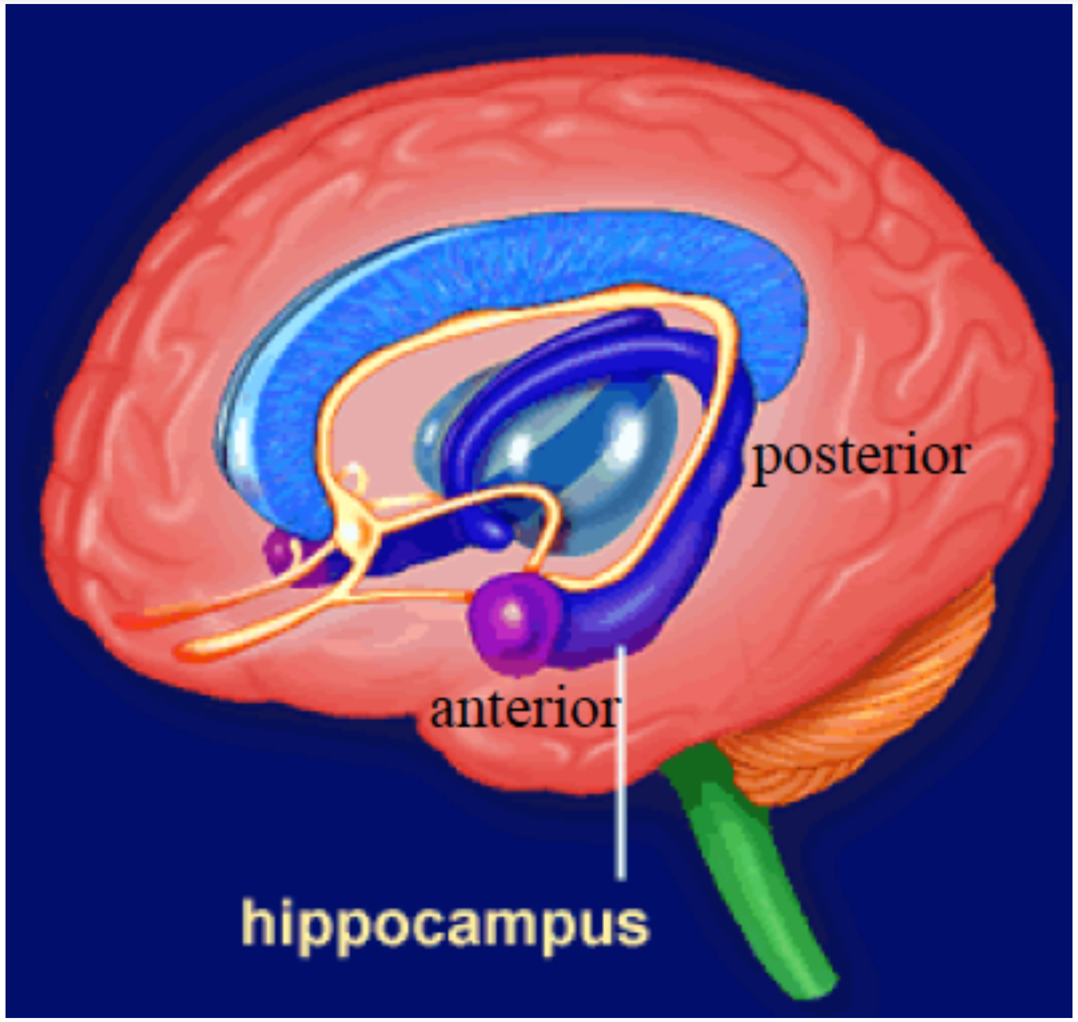
CS375 / Psych 249:
Large-Scale Neural Network Models for Neuroscience

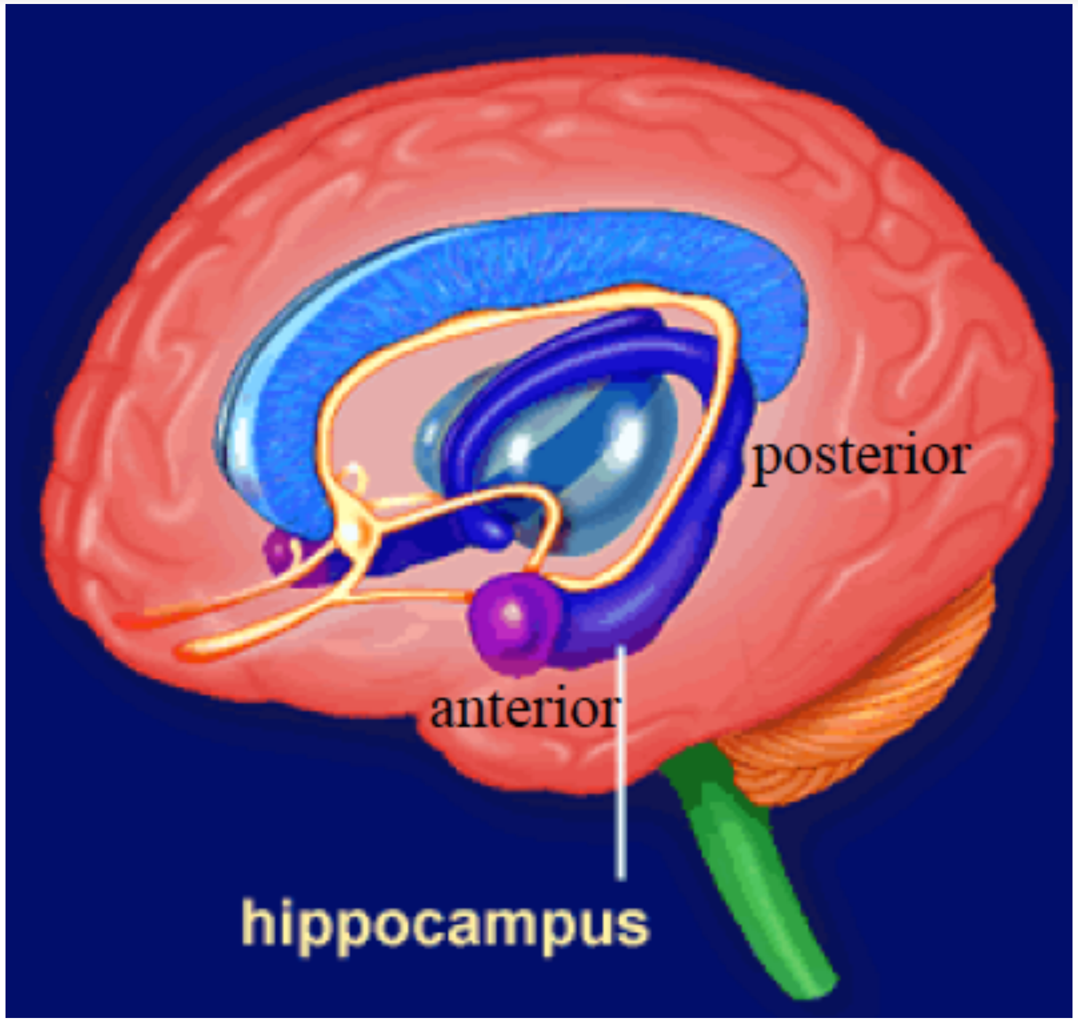
Lecture 9: Models of the Hippocampus (Memory, Navigation)
2026.02.18

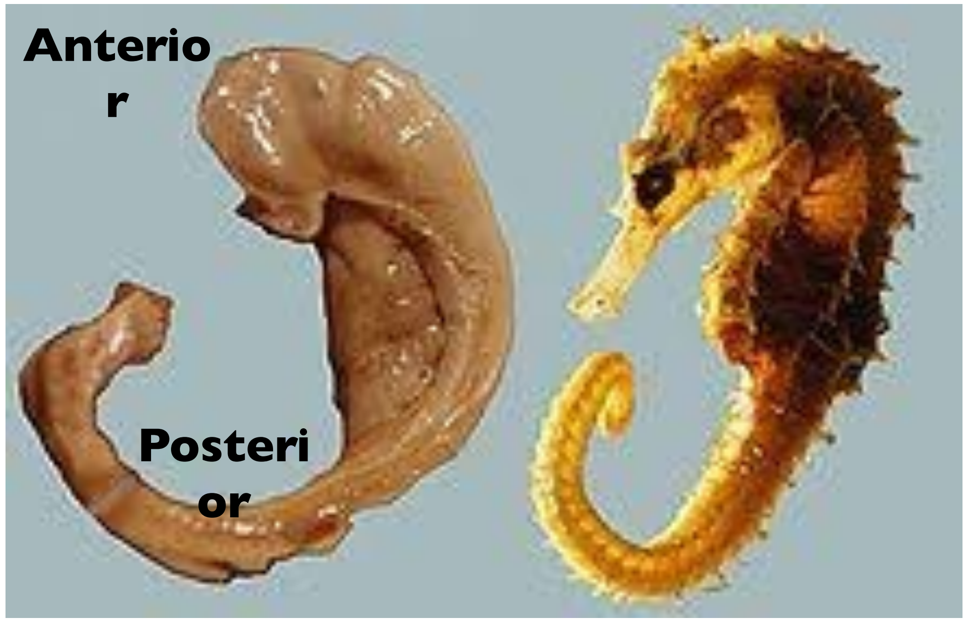
Daniel Yamins
Departments of Computer Science and of Psychology
Stanford Neuroscience and Artificial Intelligence Laboratory
Wu Tsai Neurosciences Institute
Stanford University

The Hippocampus



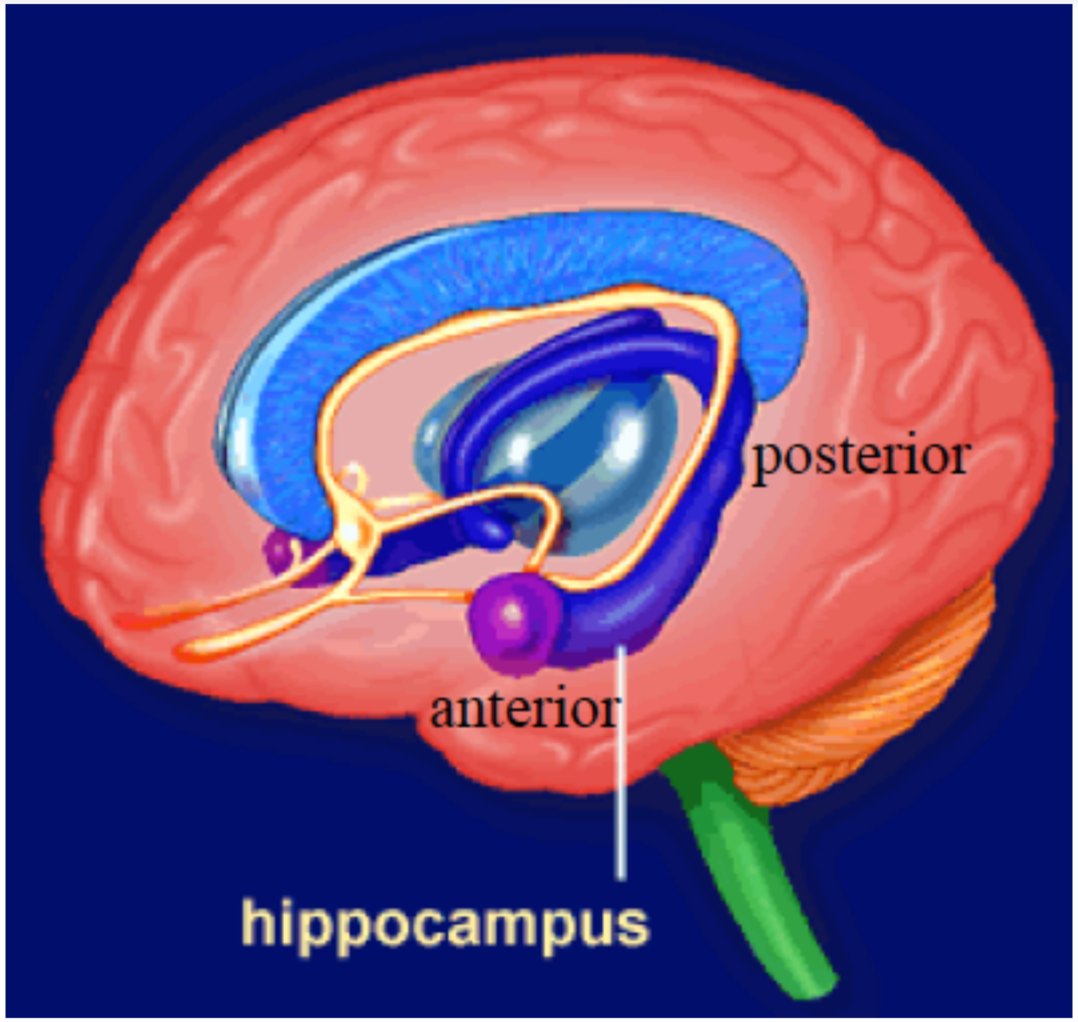
The Hippocampus

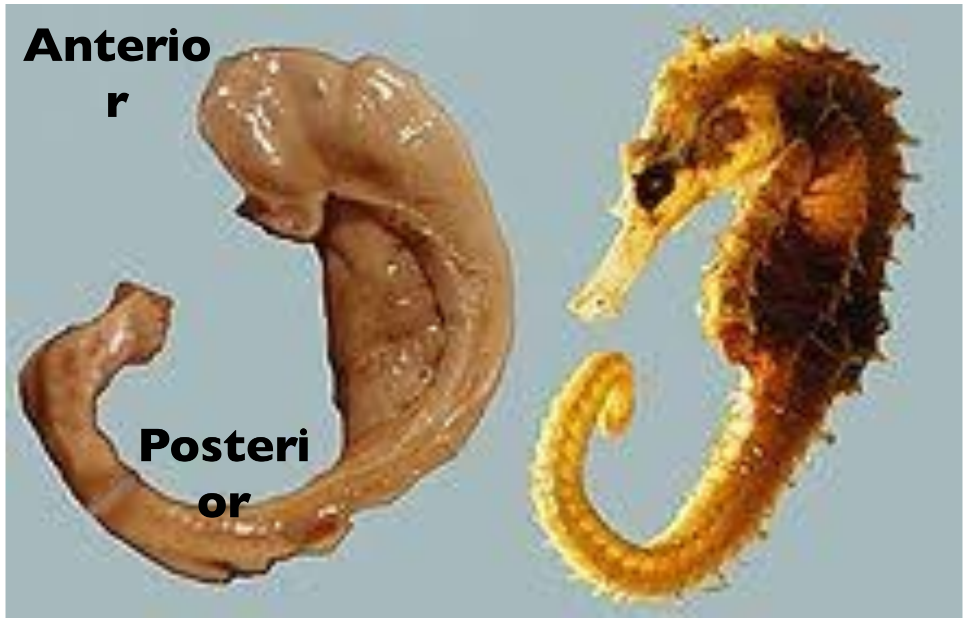




Latin for "seahorse"

The Hippocampus



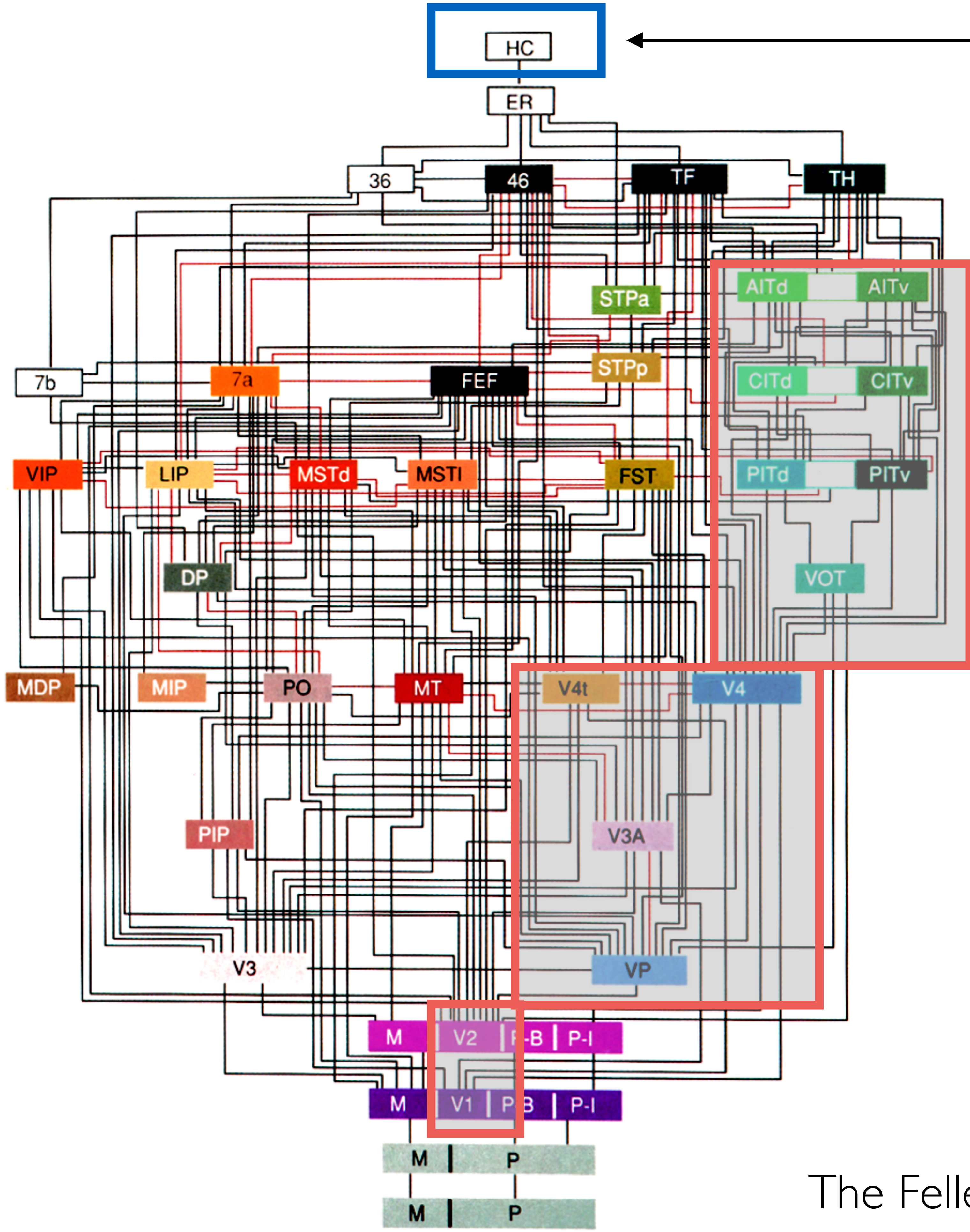


Latin for "seahorse"

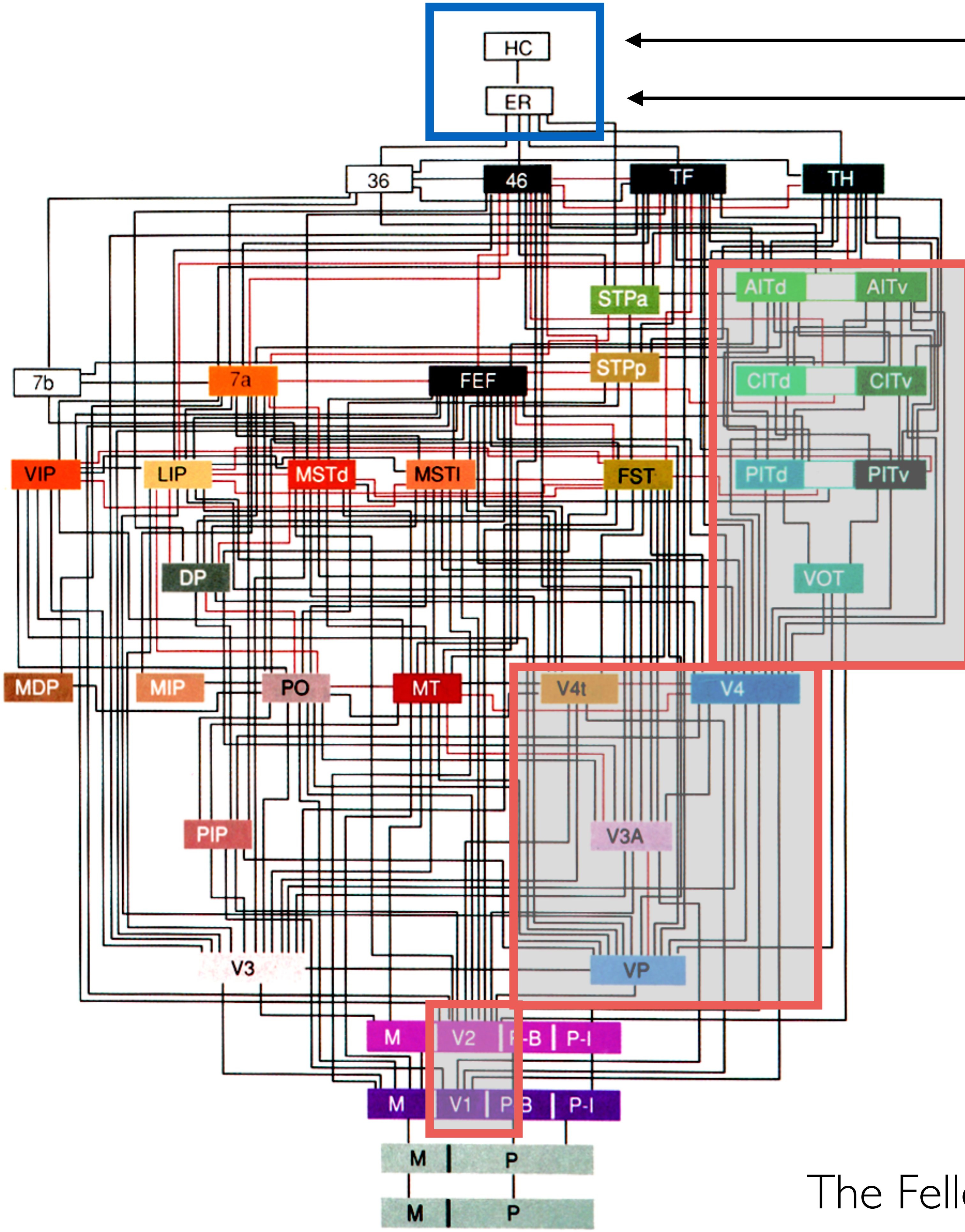
"Cortex" = archicortex (hippocampus) + neocortex (PFC, visual, etc)

archi = ancient, b/c earlier evolutionarily

HPC at the top.



The Felleman-vanEssen Diagram

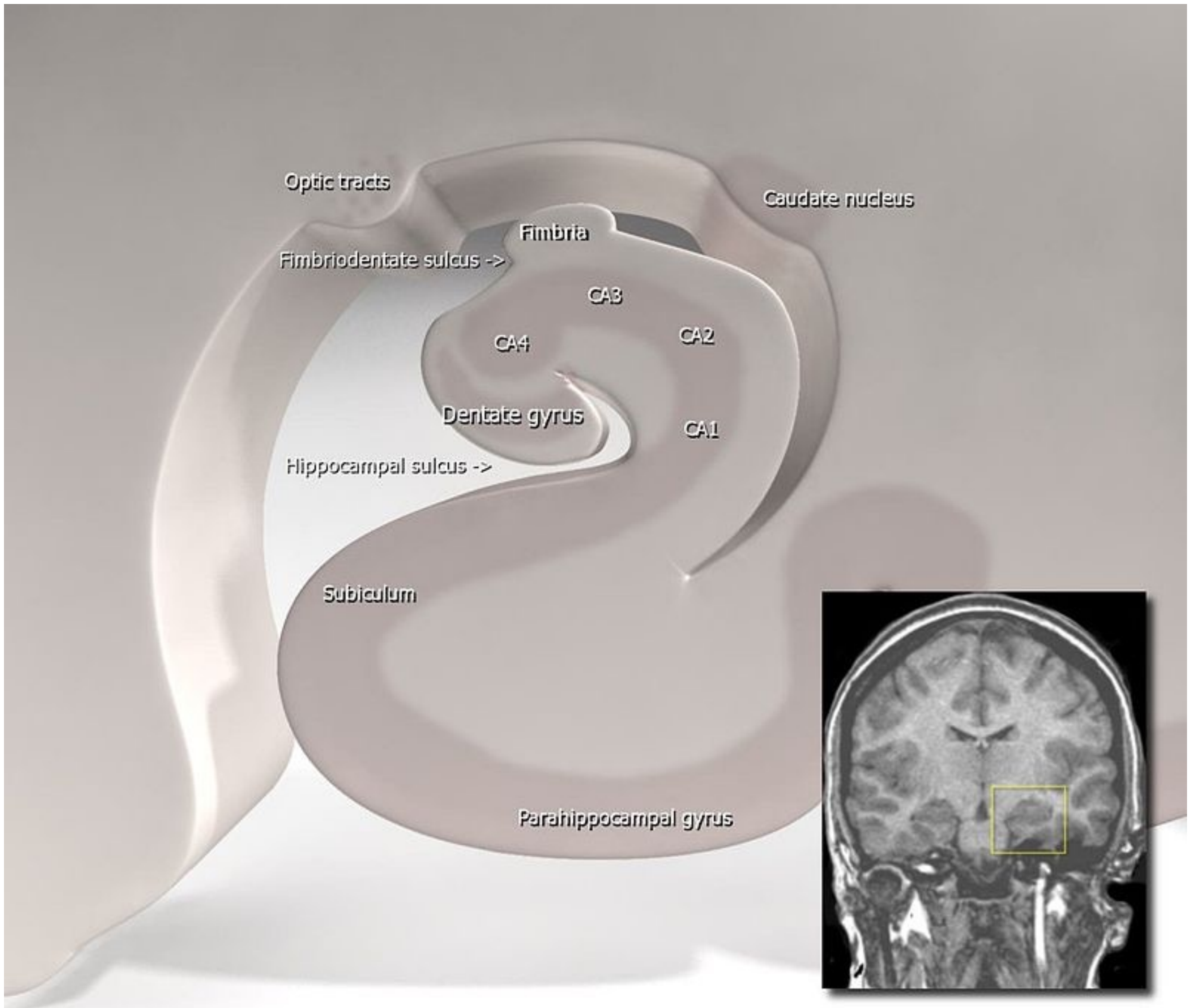


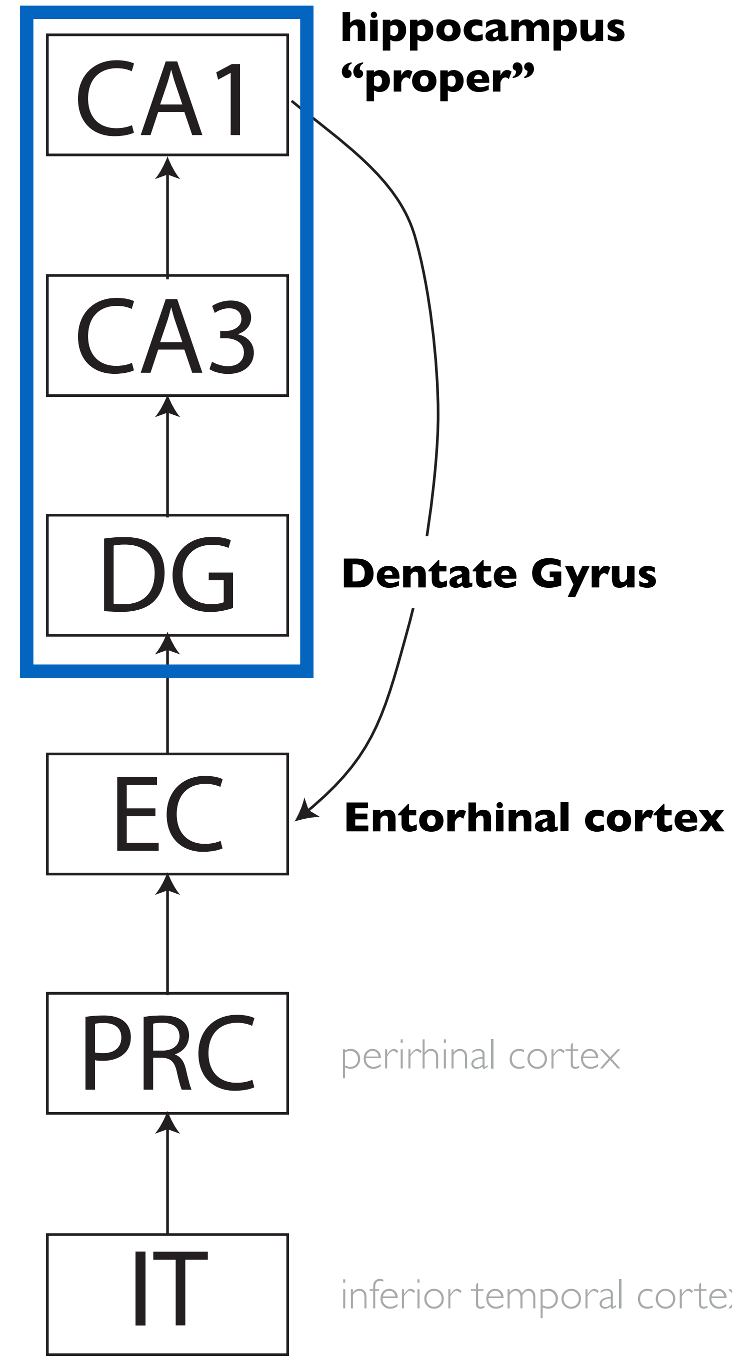
HPC at the top.

Entorhinal cortex just below HPC, and above IT

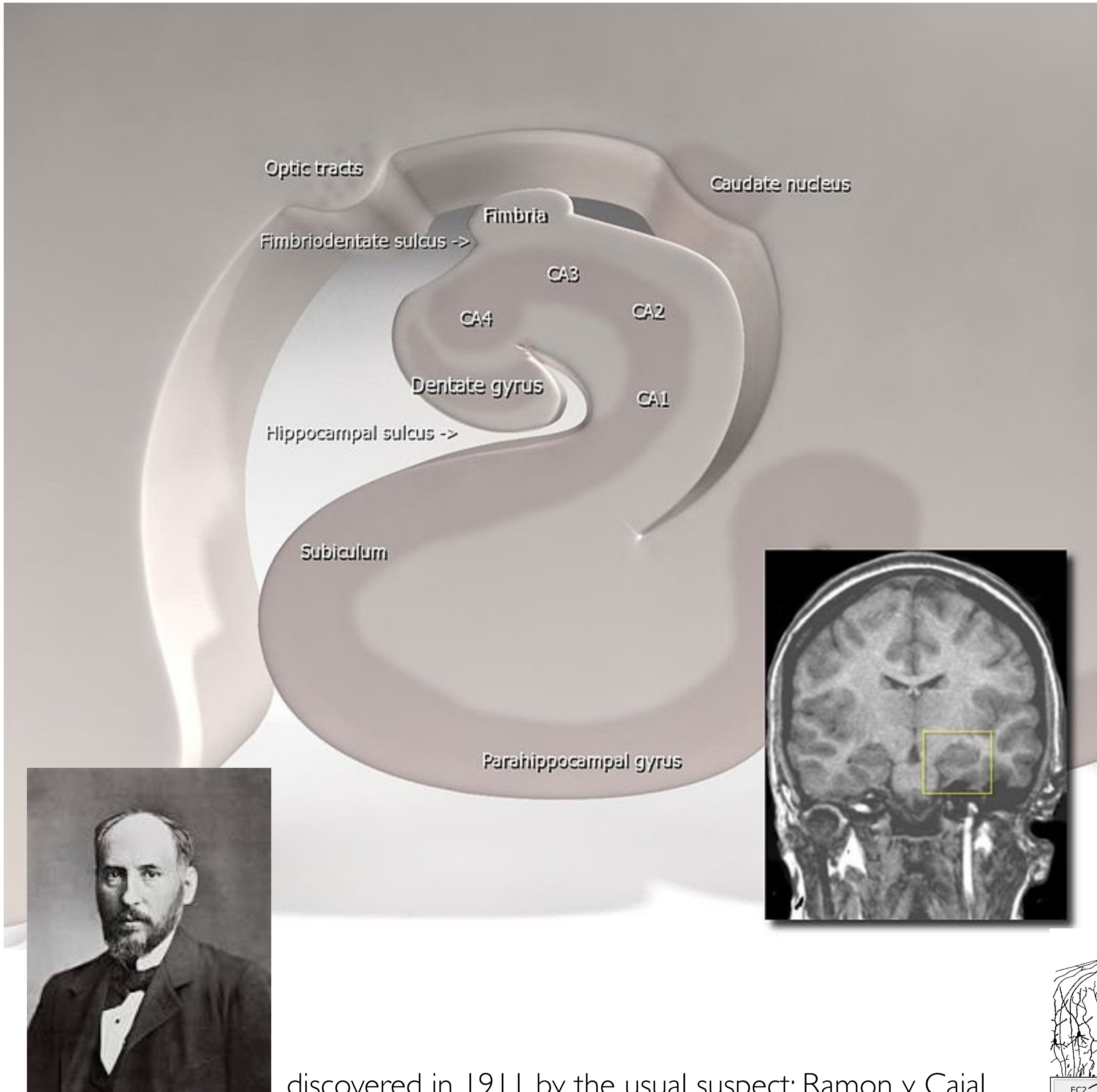
The Felleman-vanEssen Diagram

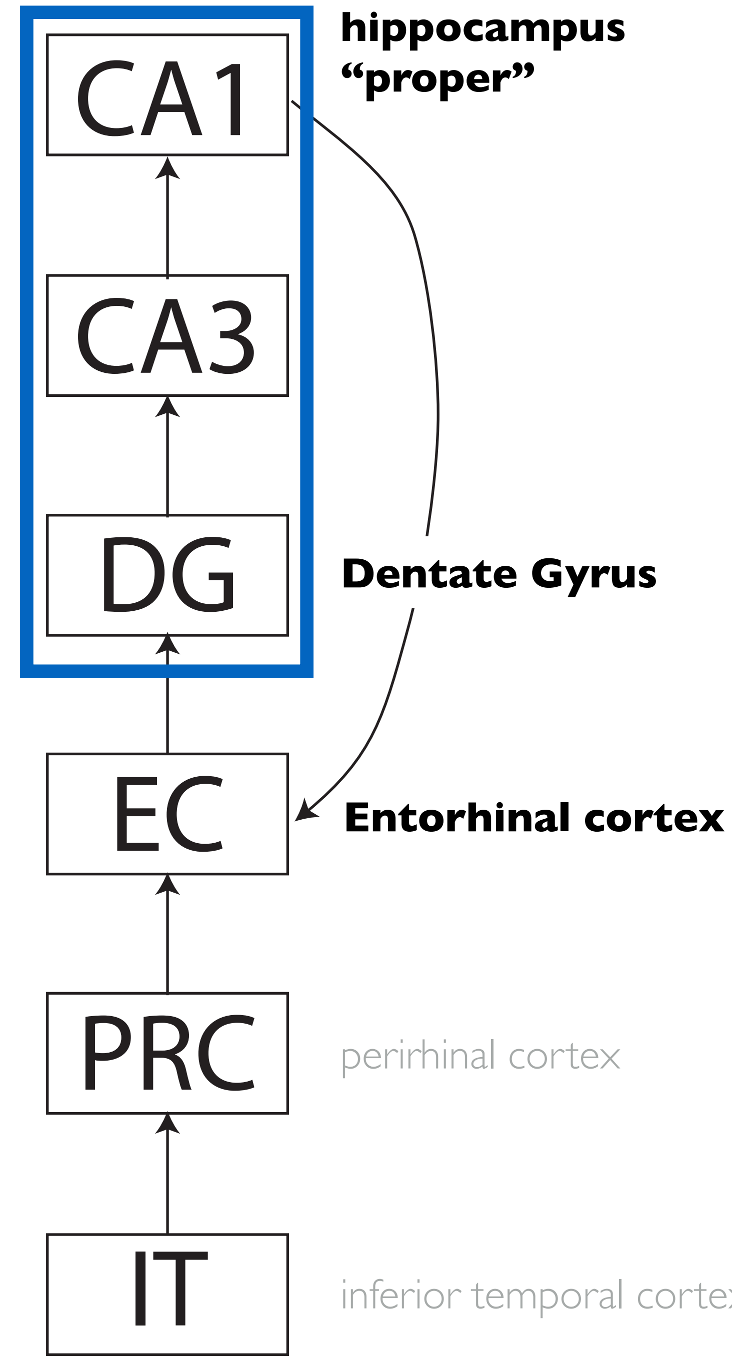
Anatomy of the Tri-synaptic circuit



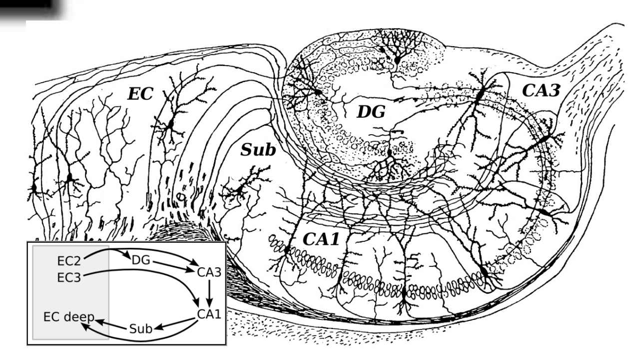


Anatomy of the Tri-synaptic circuit



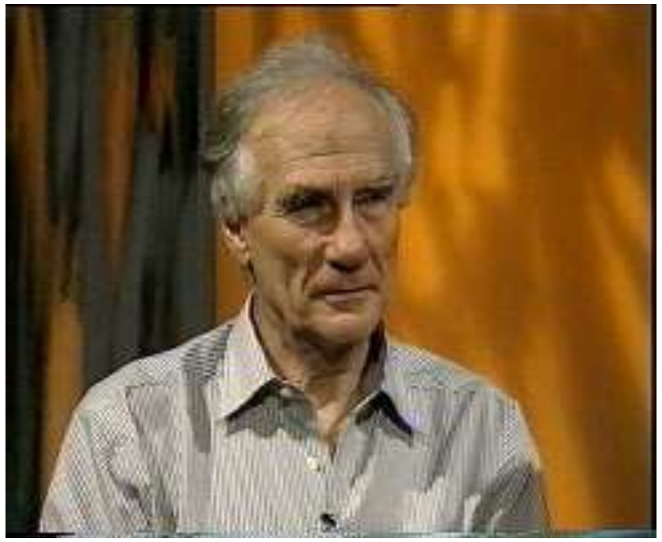


discovered in 1911 by the usual suspect: Ramon y Cajal



Functions of the hippocampus

1. Behavioral inhibition theory ("slam on the breaks")



Jeffrey Gray

Functions of the hippocampus

1. Behavioral inhibition theory ("slam on the breaks")

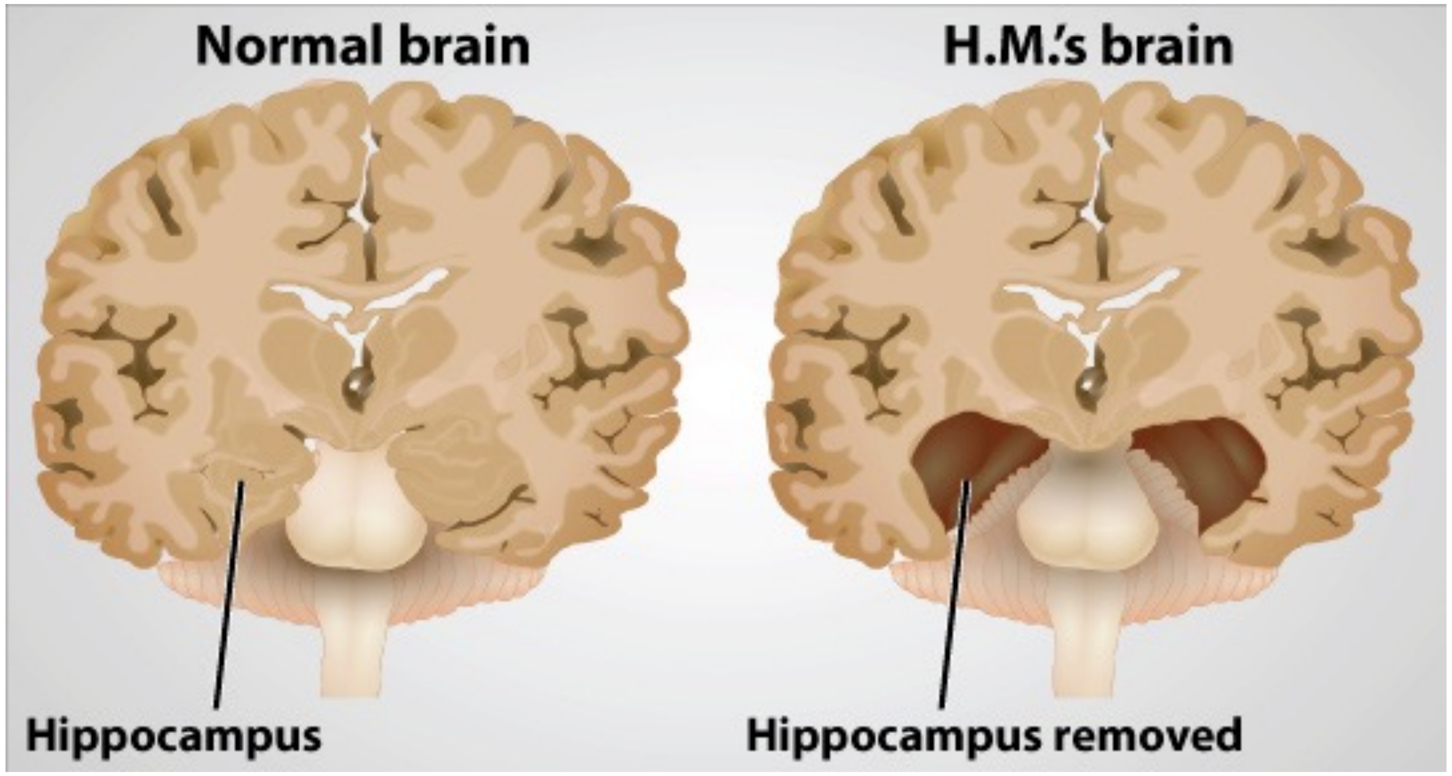
2. Memory

The Hippocampus

patient H.M.

The Hippocampus

patient H.M.

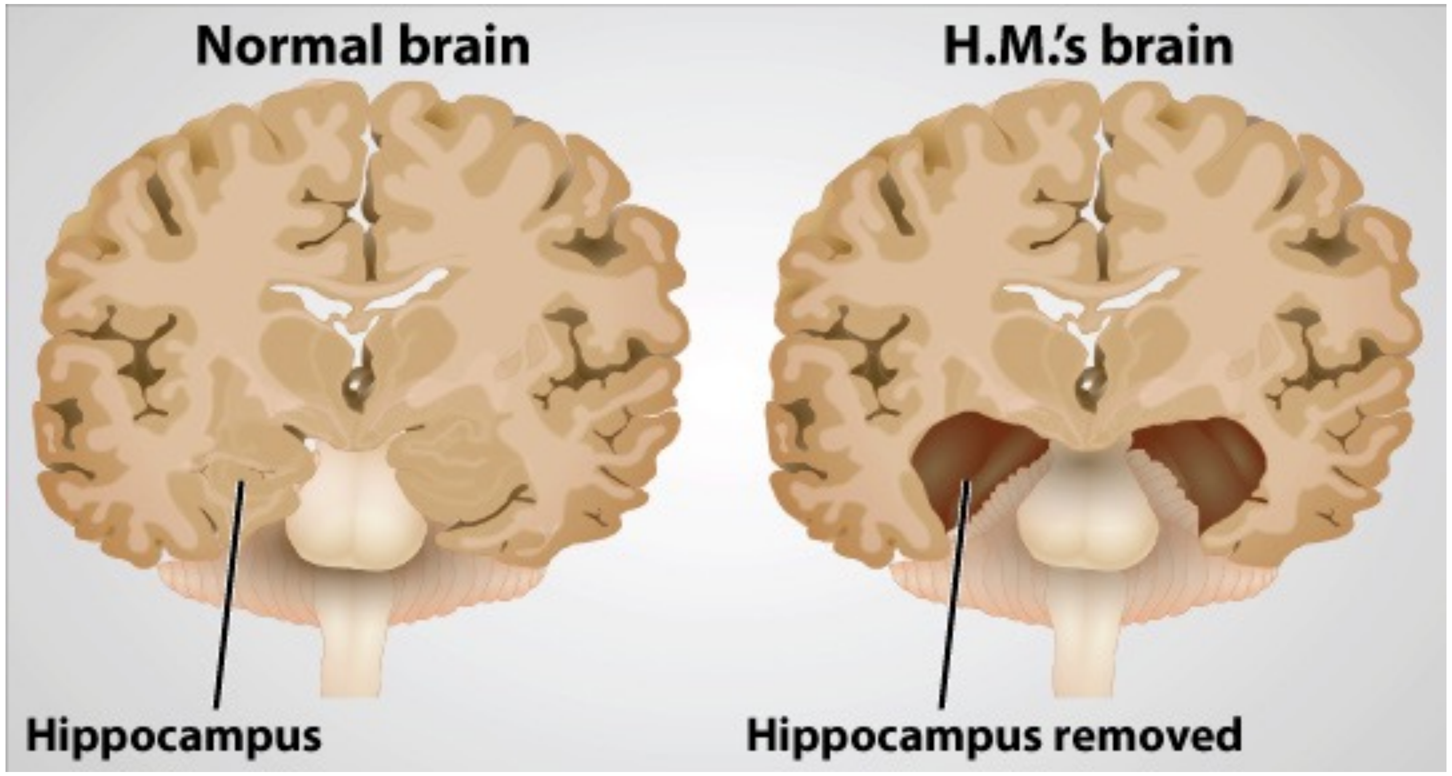


Temporal lobectomy (to treat epilepsy)

- resolved his epilepsy, but….

The Hippocampus

patient H.M.

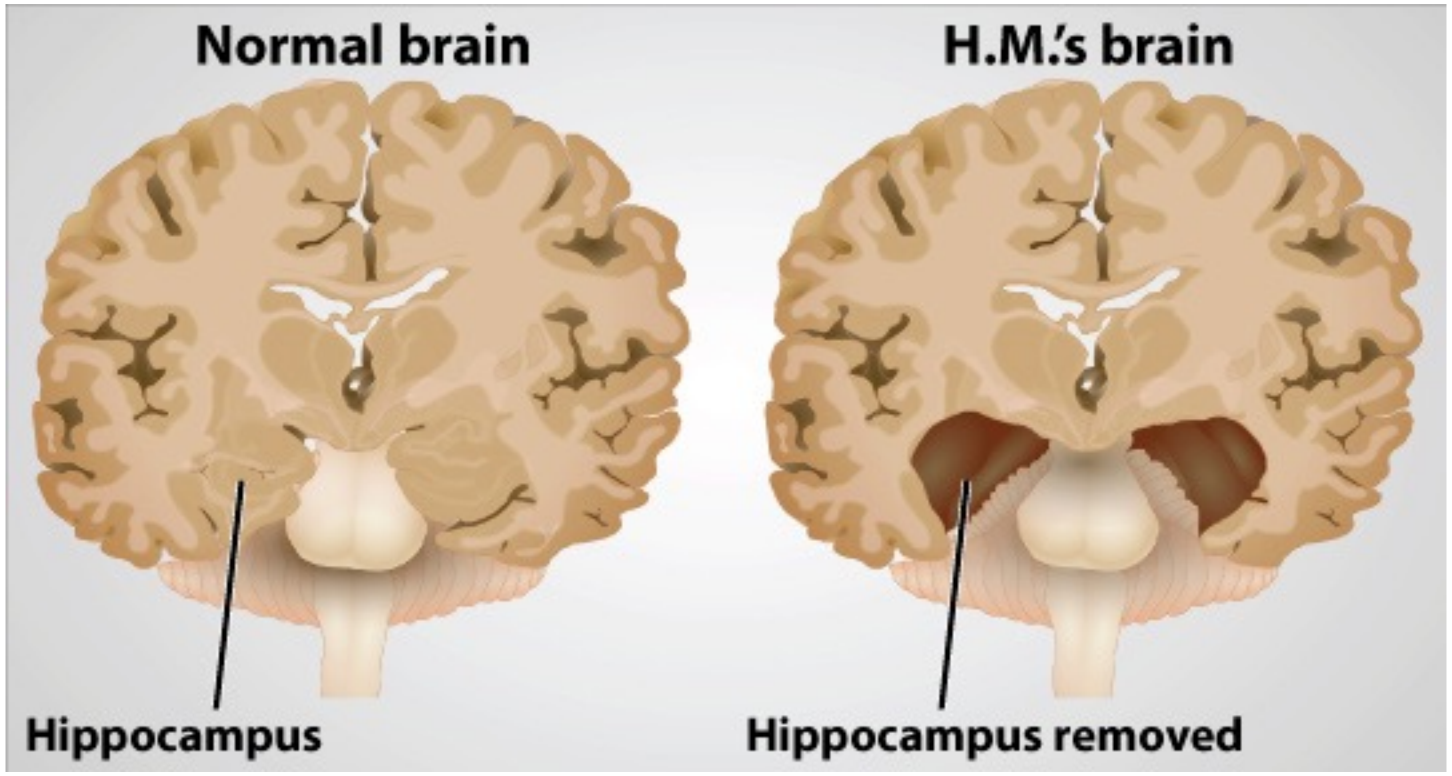


Temporal lobectomy (to treat epilepsy)

- resolved his epilepsy, but….
- could no longer form memories (though cognitive capabilities intact)

The Hippocampus

patient H.M.



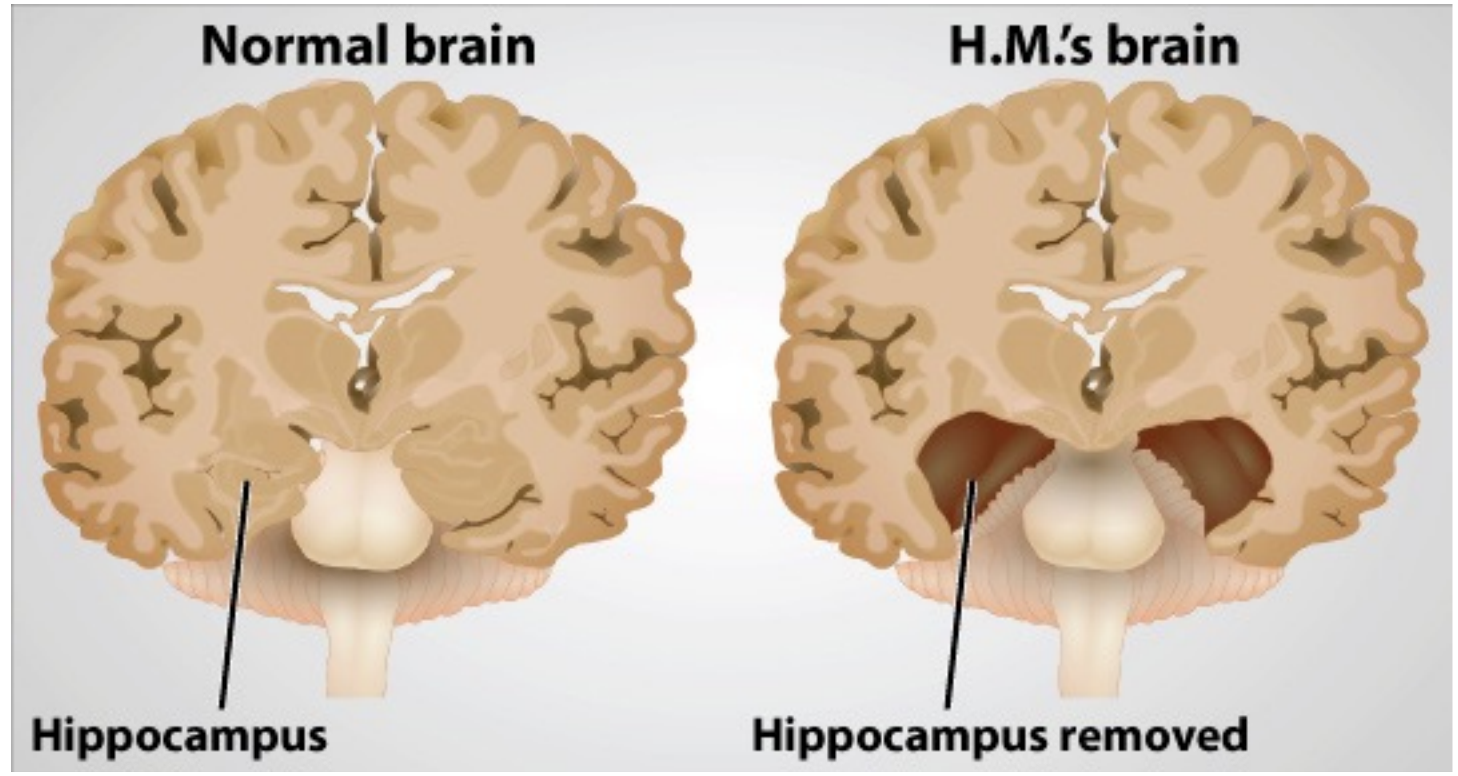
Temporal lobectomy (to treat epilepsy)

- resolved his epilepsy, but….
- could no longer form memories (though cognitive capabilities intact)

Hippocampal dysfunction leaves old/semantic knowledge intact, but disrupts recent memory formation and new learning

The Hippocampus

patient H.M.



Temporal lobectomy (to treat epilepsy)

- resolved his epilepsy, but….
- could no longer form memories (though cognitive capabilities intact)

Hippocampal dysfunction leaves old/semantic knowledge intact, but disrupts recent memory formation and new learning

Consolidated old memory

Not-yet-consolidated

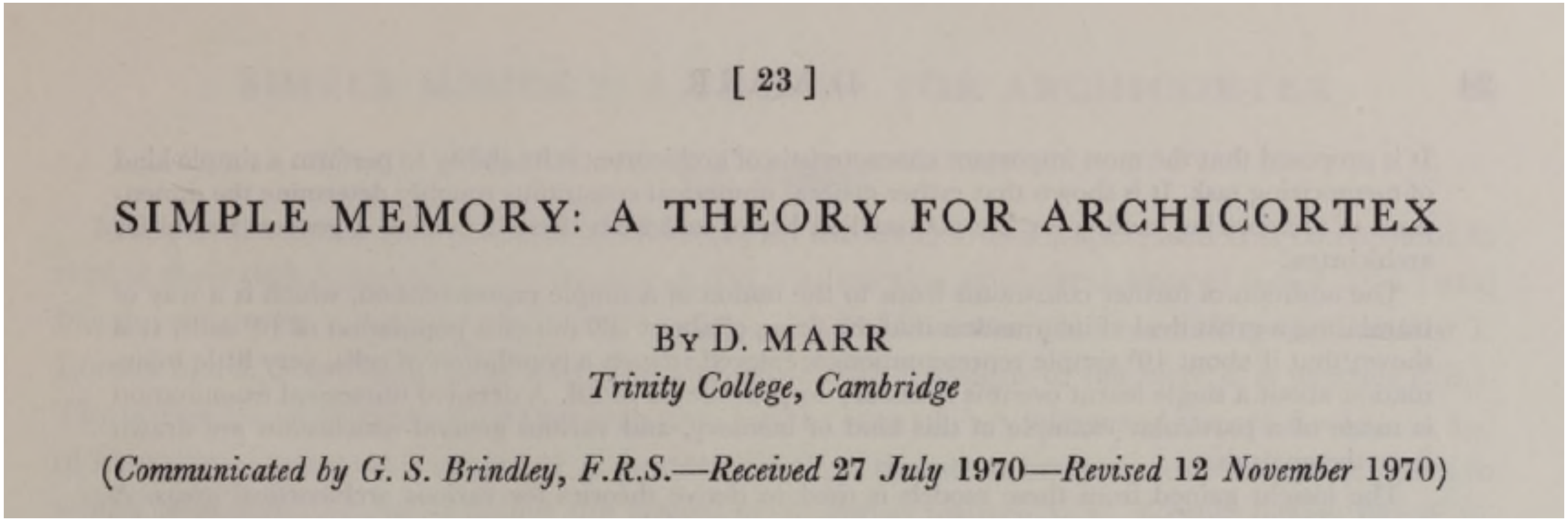
Anterograde amnesia

Functions of the hippocampus

1. Behavioral inhibition theory (“slam on the breaks”)

2. Memory (Milner & Scoville from HM)

Models: Marr



[23]

SIMPLE MEMORY: A THEORY FOR ARCHICORTEX

BY D. MARR
Trinity College, Cambridge

(*Communicated by G. S. Brindley, F.R.S.—Received* 27 *July* 1970—*Revised* 12 *November* 1970)

David Marr

(Tommy Poggio)

Models: Marr

SIMPLE MEMORY

(*Communicated by G. S. Brindley,*

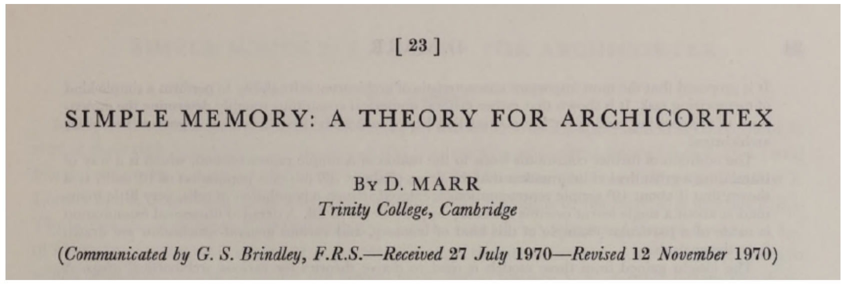
David Marr

(Tommy Poggio)

CONTENTS

Vol. 262. B. 841. (Price £1.55; U.S. $4.00) 3 [Published 1 July 1971

Models: Marr


[23]

SIMPLE MEMORY: A THEORY FOR ARCHICORTEX

By D. MARR
Trinity College, Cambridge

(*Communicated by G. S. Brindley, F.R.S.—Received* 27 *July* 1970—*Revised* 12 *November* 1970)

HPC as "medium-term storage for training" deep cortex.

HPC stores patterns immediately, w/o further analysis

Neocortex later picks out important features, might take a while ('consolidation')

Can support sensory -> HPC construction of "codons" then, recovery of pattern from partial input

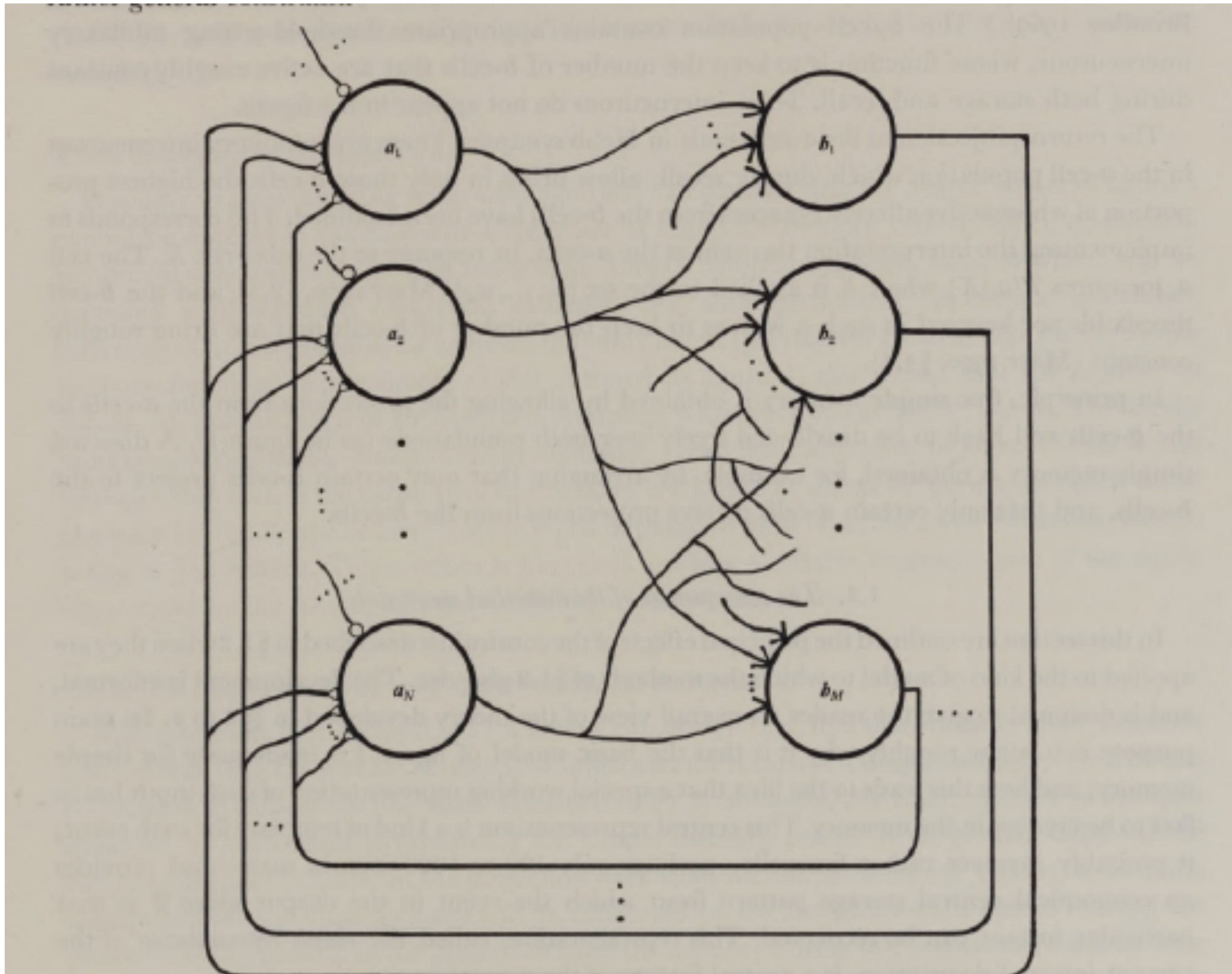


FIGURE 1. A primitive associative memory. The current internal description is an event on the cells $a_1, \ldots, a_N$: this is given a codon representation in the cells $b_1, \ldots, b_M$ (which have Brindley afferent synapses), and the return to the a_i-cells is through Hebb modifiable synapses. The various inhibitory interneurons necessary for the correct operation of the system have been omitted. This class of model provides an efficient associative memory for events on the a_i as long as their number and size are suitably restricted.

Two-layer recurrent model

Models: Marr

[23]

SIMPLE MEMORY: A THEORY FOR ARCHICORTEX

By D. MARR
Trinity College, Cambridge

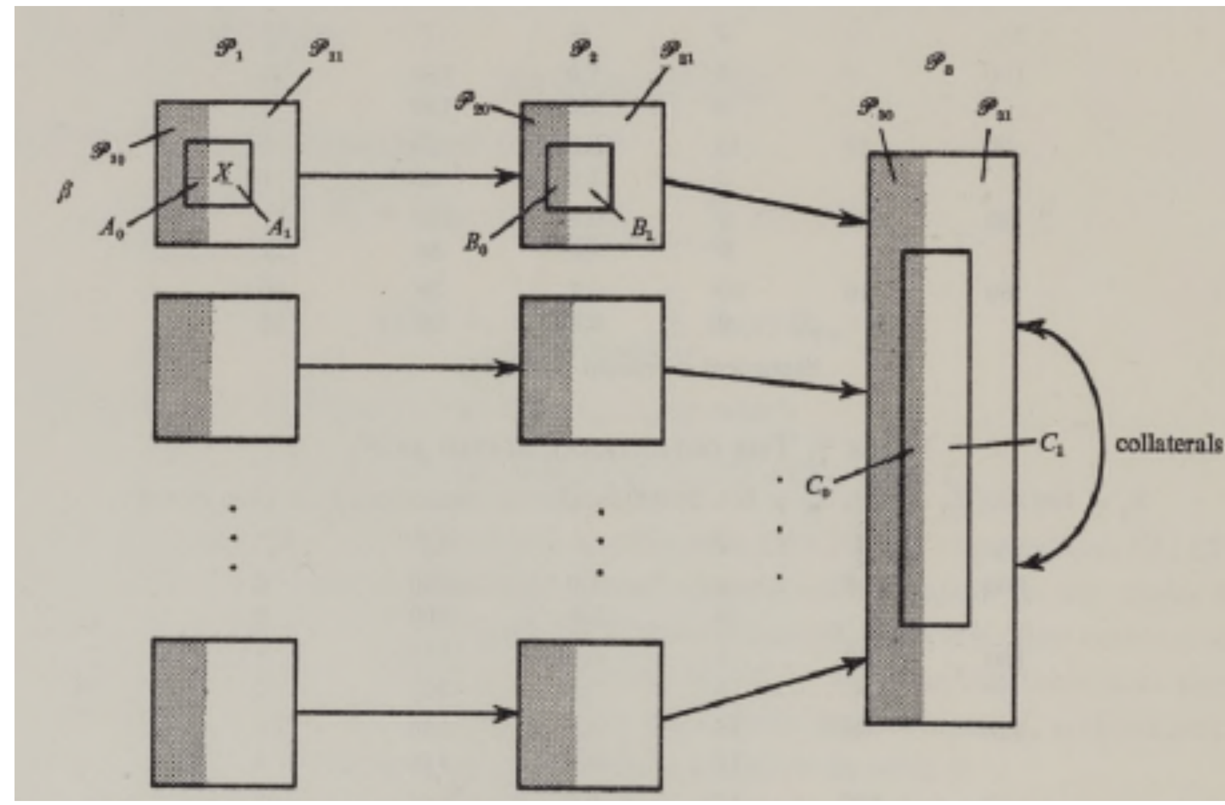
(*Communicated by G. S. Brindley, F.R.S.—Received* 27 *July* 1970—*Revised* 12 *November* 1970)

HPC as "medium-term storage for training" deep cortex.

HPC stores patterns immediately, w/o further analysis

Neocortex later picks out important features, might take a while ('consolidation')

Can support sensory -> HPC construction of "codons" then, recovery of pattern from partial input



Three-layer recurrent model

FIGURE 6. The recall problem. $\mathscr{P}_1$, $\mathscr{P}_2$ and $\mathscr{P}_3$ are the populations of cells defined in table 1. Shading represents the parts of these populations involved in the storage of an event E_0. A new subevent X is presented to one block of $\mathscr{P}_1$, A_0 of whose cells were involved in E_0, and A_1 of which were not. This produces activity in one block of $\mathscr{P}_2$, and in $\mathscr{P}_3$. B_0 of the active cells in $\mathscr{P}_2$ were active in E_0, and B_1 were not: C_0 of the active cells in $\mathscr{P}$ were also active in E_0, and C_1 were not. The numbers A_i, B_i, C_i, $(i = 1, 2)$ are computed in the text.

Models: Marr

[23]

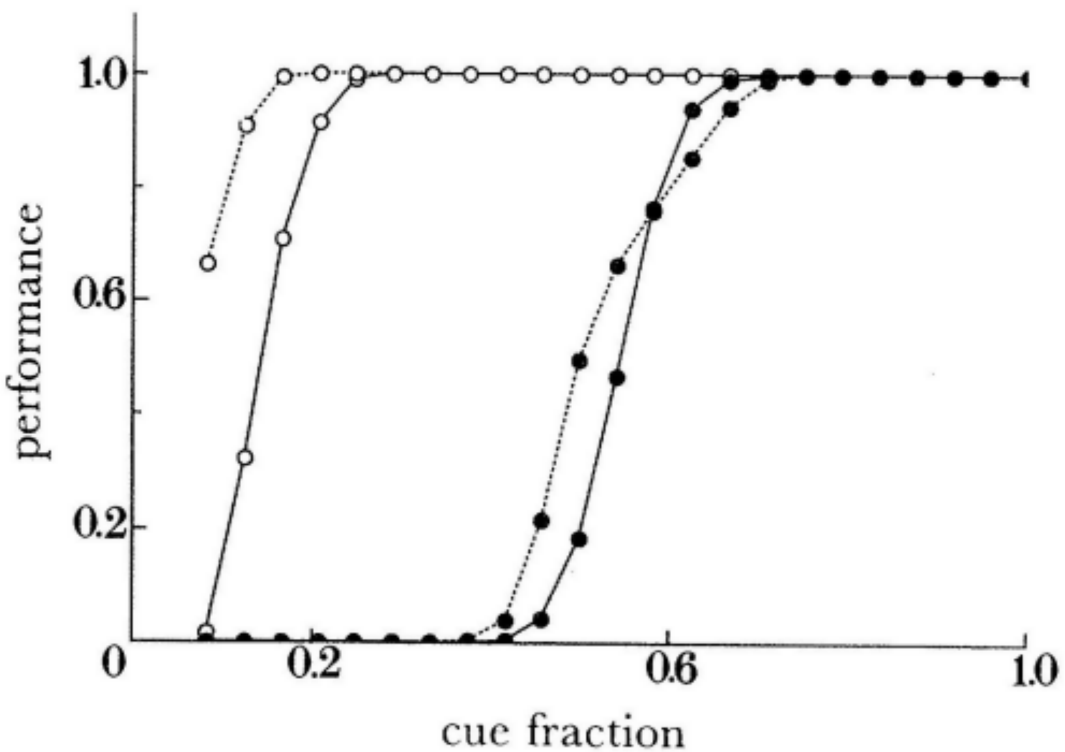
SIMPLE MEMORY: A THEORY FOR ARCHICORTEX

BY D. MARR
Trinity College, Cambridge

(*Communicated by G. S. Brindley, F.R.S.—Received* 27 *July* 1970—*Revised* 12 *November* 1970)

HPC as "medium-term storage for training" deep cortex.

HPC stores patterns immediately, w/o further analysis



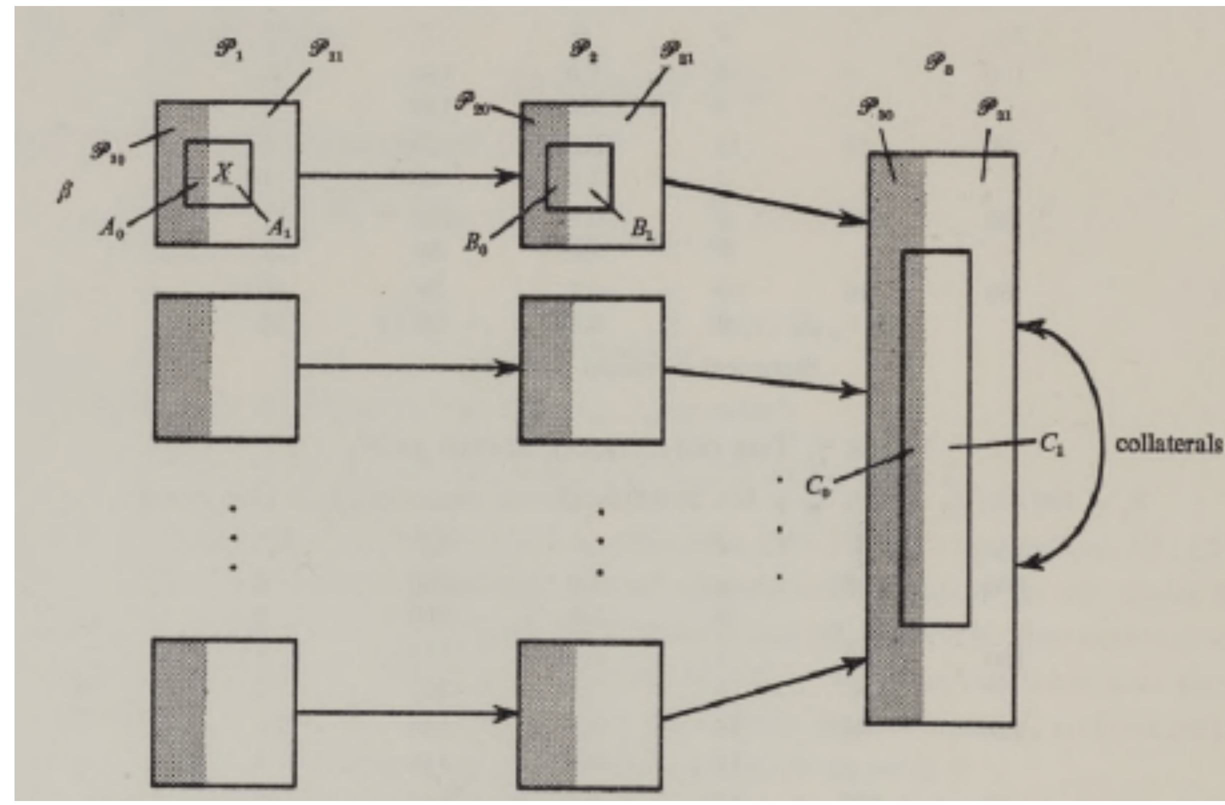


FIGURE 6. The recall problem. $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$ are the populations of cells defined in table 1. Shading represents the parts of these populations involved in the storage of an event E_0. A new subevent X is presented to one block of $\mathcal{P}_1$, A_0 of whose cells were involved in E_0, and A_1 of which were not. This produces activity in one block of $\mathcal{P}_2$, and in $\mathcal{P}_3$. B_0 of the active cells in $\mathcal{P}_2$ were active in E_0, and B_1 were not: C_0 of the active cells in $\mathcal{P}$ were also active in E_0, and C_1 were not. The numbers A_i, B_i, C_i, $(i = 1, 2)$ are computed in the text.

○ = partial cue

● = noisy cue

------ = two-layer

—— = three-layer

Third layer basically irrelevant

Models: Complementary Learning Systems ("CLS")

Why There Are Complementary Learning Systems in the Hippocampus and Neocortex: Insights From the Successes and Failures of Connectionist Models of Learning and Memory

James L. McClelland
Carnegie Mellon University
and the Center for the Neural Basis of Cognition

Bruce L. McNaughton
University of Arizona

Randall C. O'Reilly
Carnegie Mellon University
and the Center for the Neural Basis of Cognition

HPC as "medium-term storage for training" deep cortex. …

Q: What happens if you don't randomize ImageNet before training?

Models: Complementary Learning Systems ("CLS")

Why There Are Complementary Learning Systems in the Hippocampus and Neocortex: Insights From the Successes and Failures of Connectionist Models of Learning and Memory

James L. McClelland
Carnegie Mellon University
and the Center for the Neural Basis of Cognition

Bruce L. McNaughton
University of Arizona

Randall C. O'Reilly
Carnegie Mellon University
and the Center for the Neural Basis of Cognition

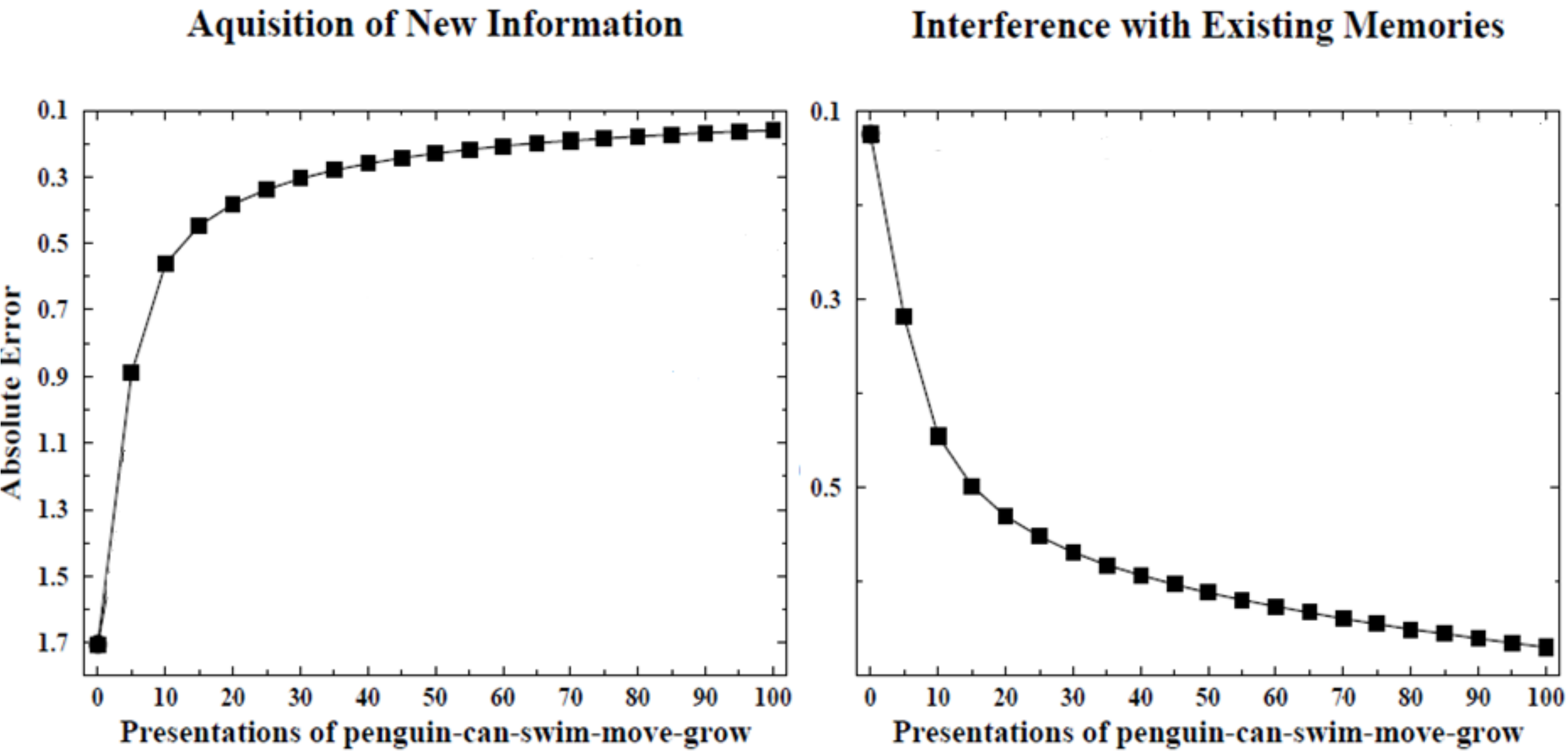
HPC as "medium-term storage for training" deep cortex. . . .

Q: What happens if you don't randomize ImageNet before training?

A: *catastrophic forgetting.*

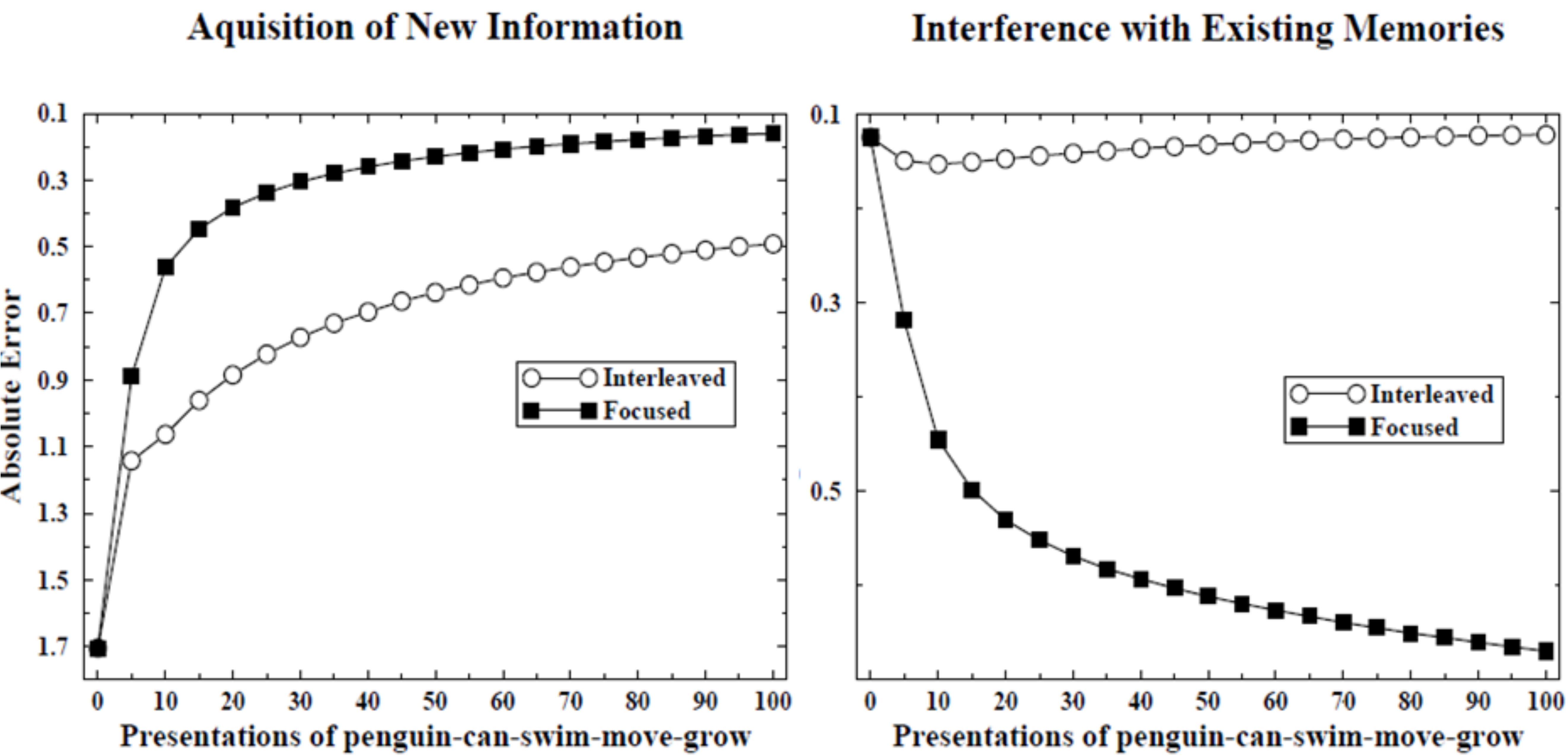
. . . because you want to avoid *catastrophic forgetting.*

Models: Complementary Learning Systems ("CLS")



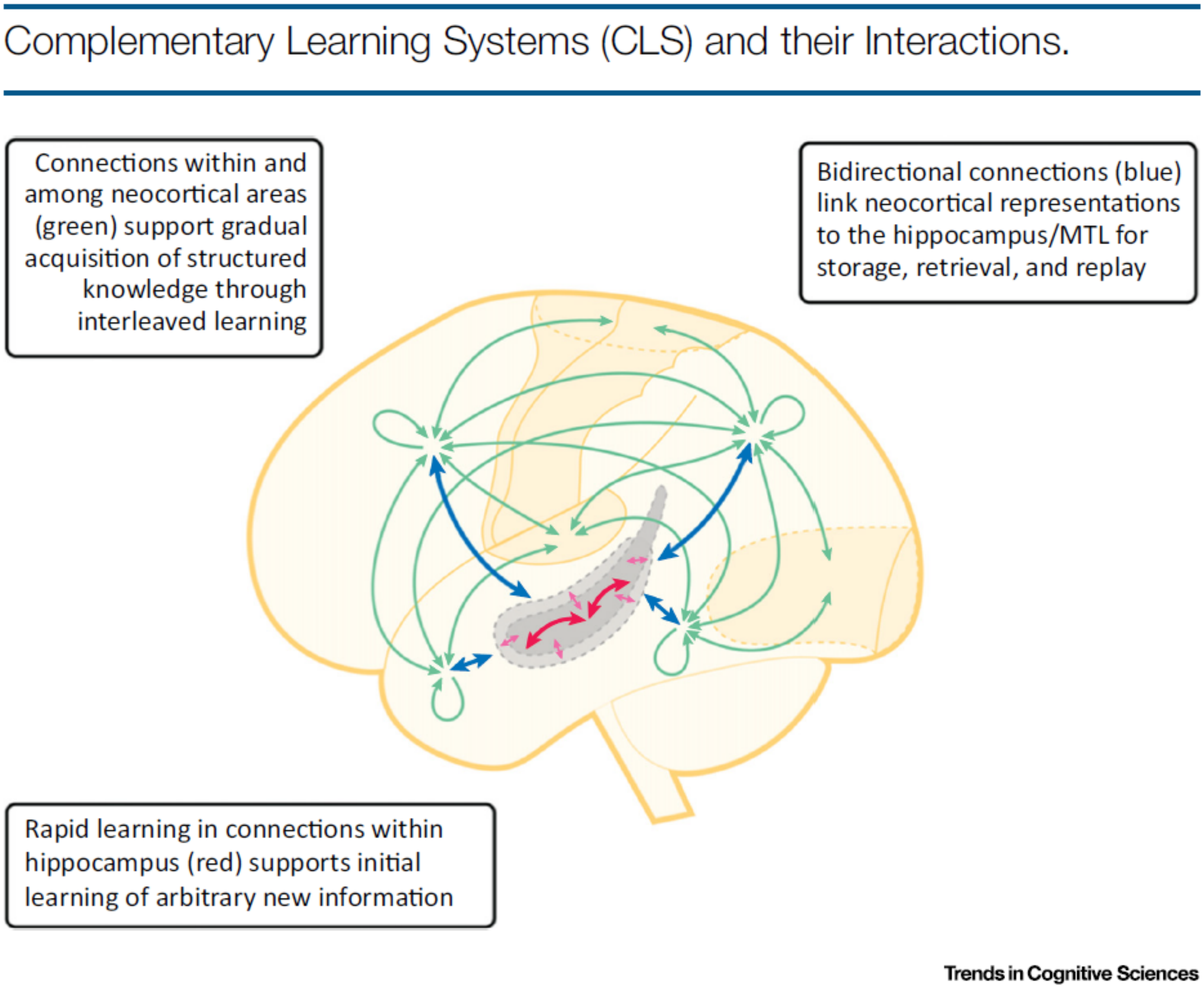
from McClelland 2013

Models: Complementary Learning Systems ("CLS")

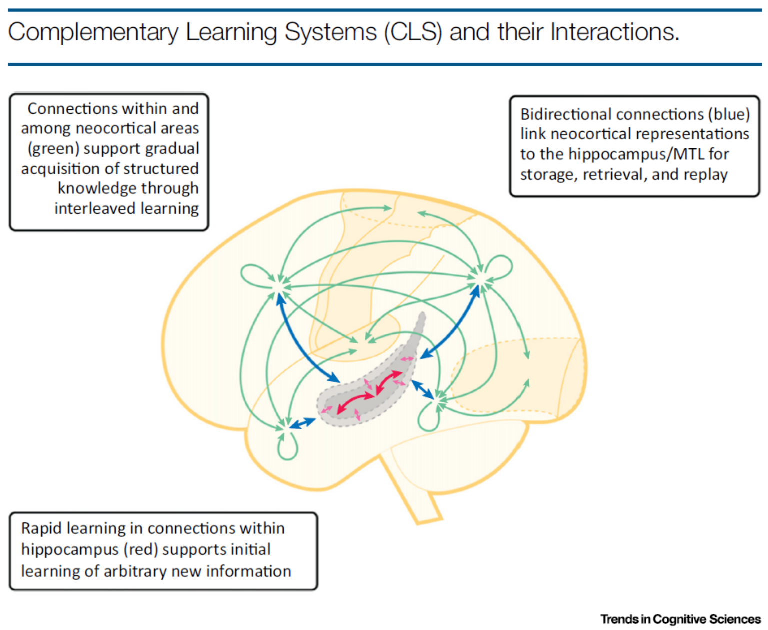


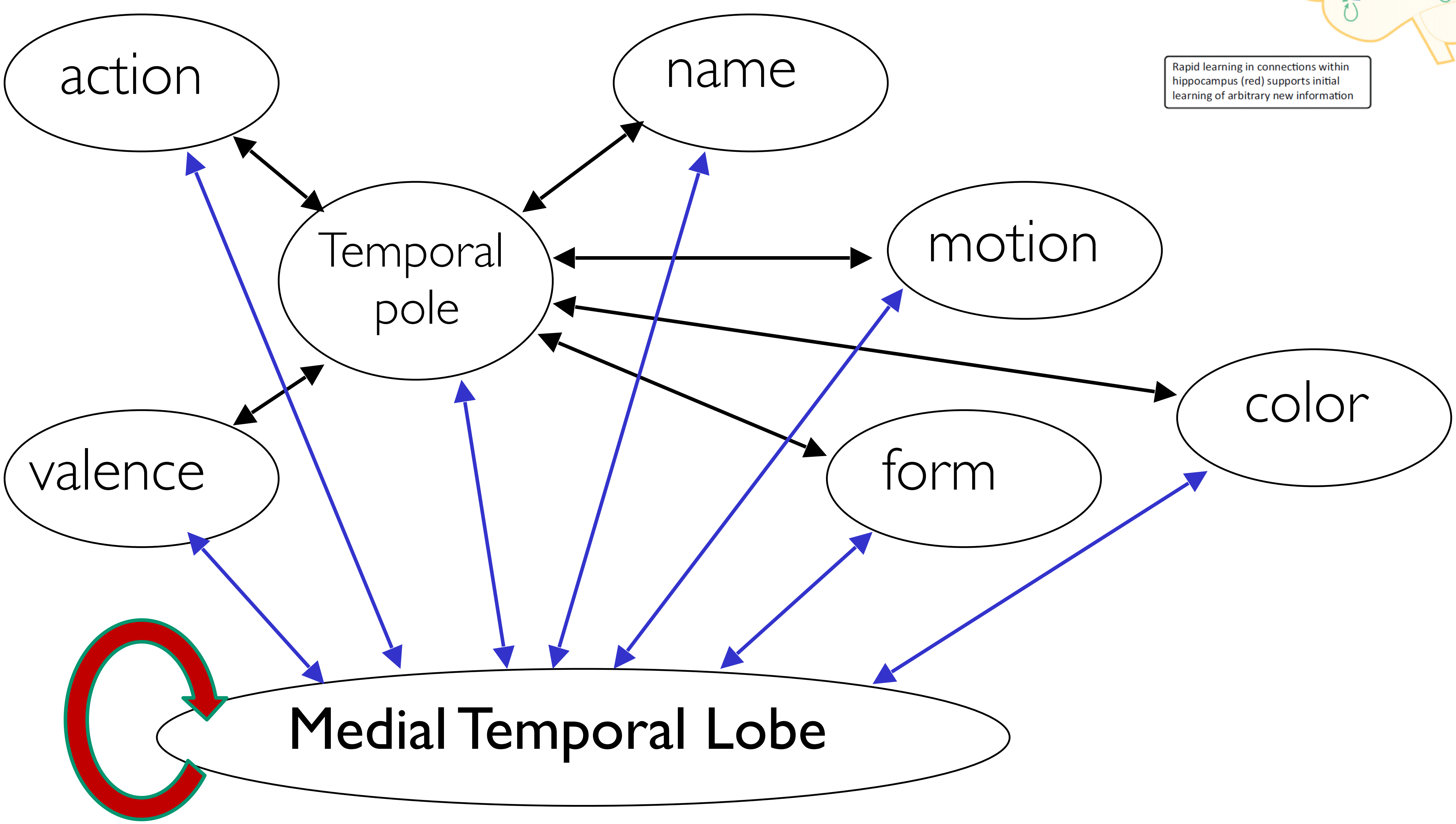
from McClelland 2013

Models: Complementary Learning Systems ("CLS")



Models: Complementary Learning Systems ("CLS")





Models: Complementary Learning Systems ("CLS")

Why There Are Complementary Learning Systems in the Hippocampus and Neocortex: Insights From the Successes and Failures of Connectionist Models of Learning and Memory

James L. McClelland
Carnegie Mellon University
and the Center for the Neural Basis of Cognition

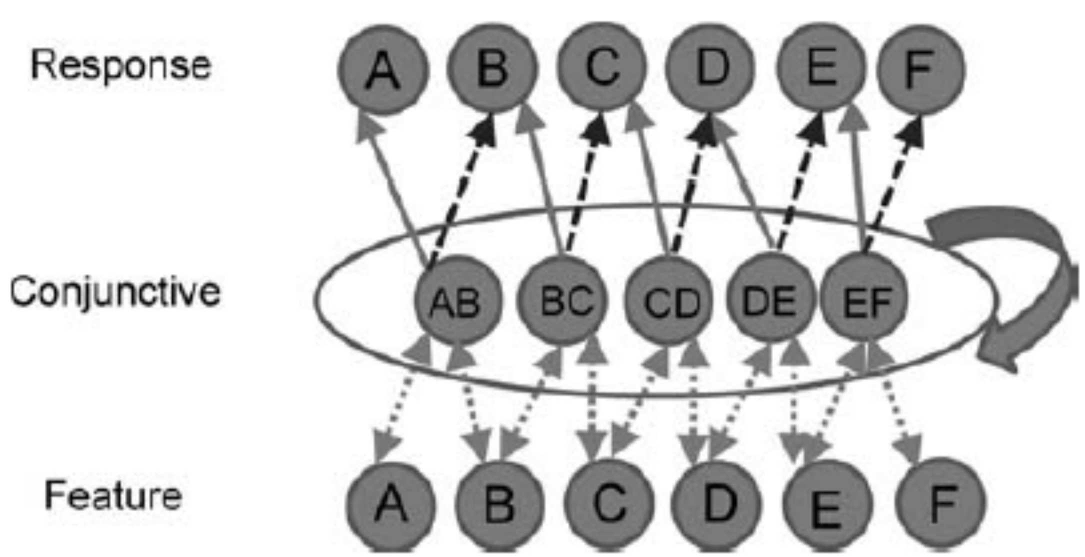
Bruce L. McNaughton
University of Arizona

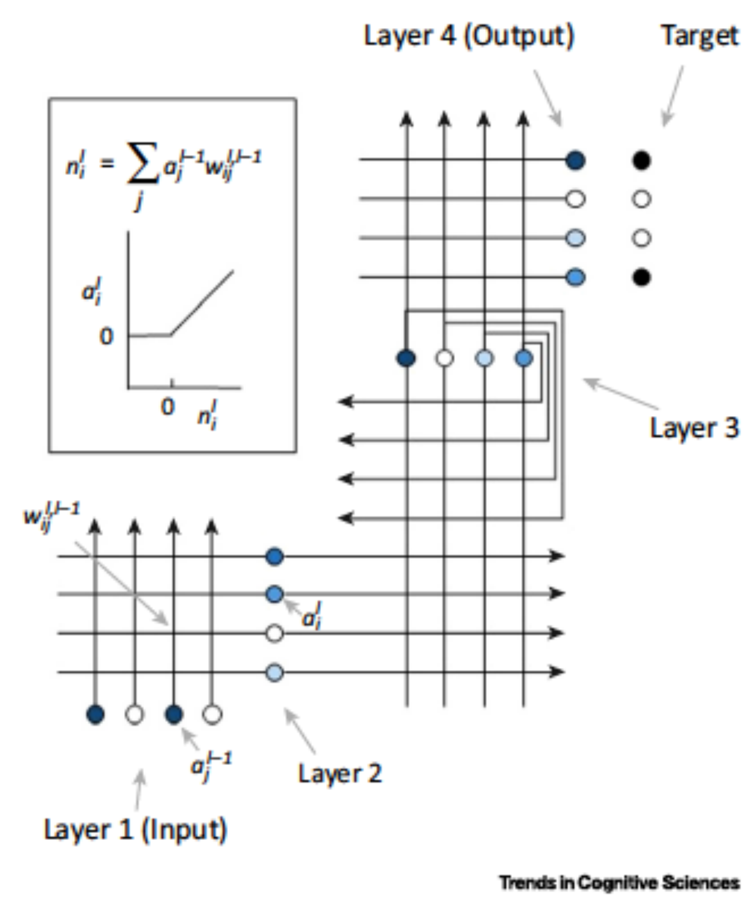
Randall C. O'Reilly
Carnegie Mellon University
and the Center for the Neural Basis of Cognition

HPC as "medium-term storage for training" deep cortex. …

… because you want to avoid *catastrophic forgetting.*

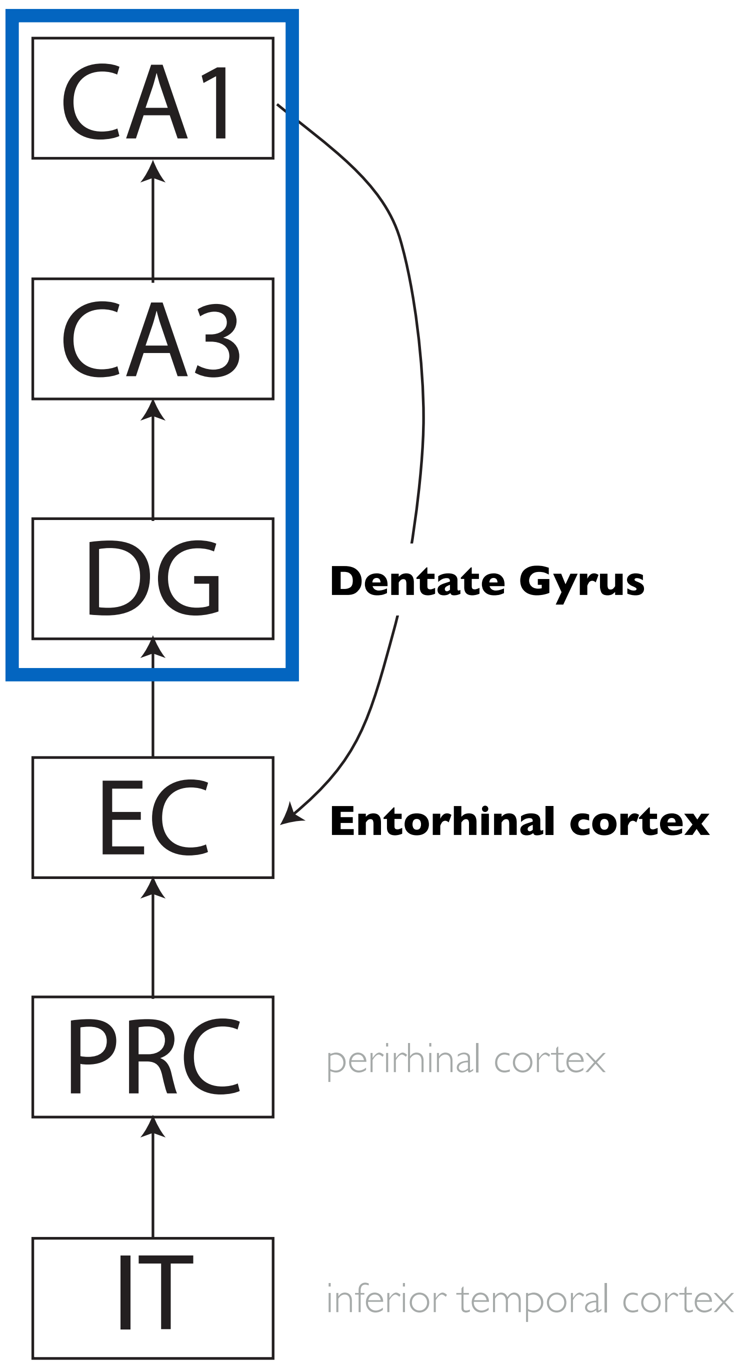
via experience replay and interleaving.



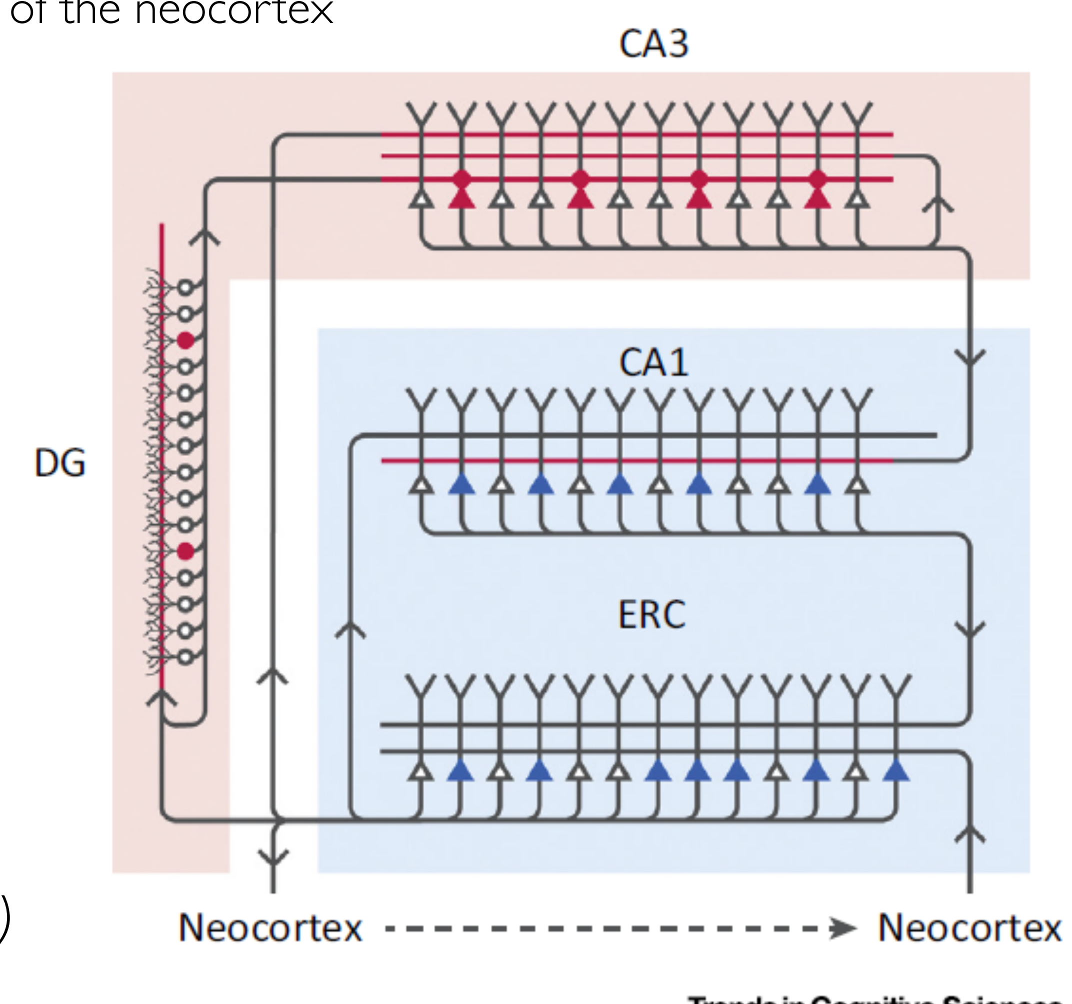


from Kumaran & McClelland (2012)

Models: Complementary Learning Systems ("CLS")

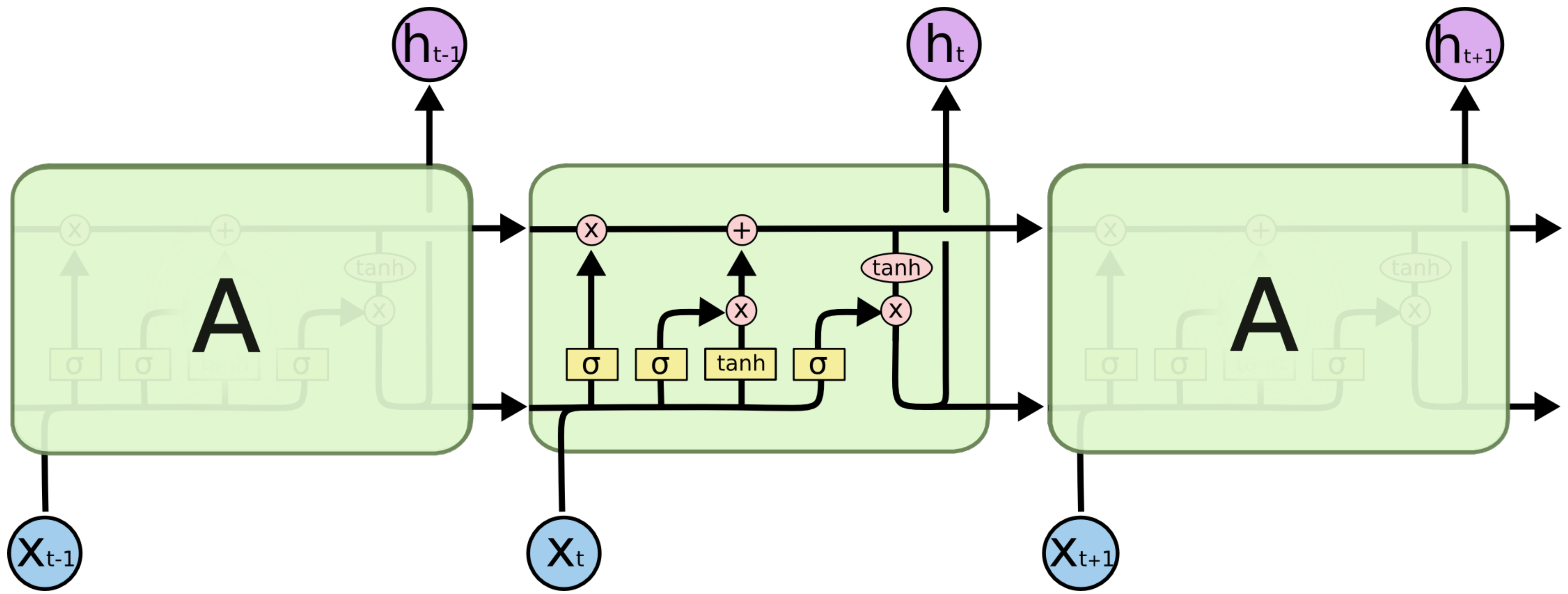


- Input from neocortex comes into EC; EC projects to DG, CA3, and CA1
- Drastic pattern separation occurs in DG
- Downsampling in CA3 assigns an arbitrary code
- Invertable somewhat sparsified representation in CA1
- Fewish-shot learning in DG, CA3, CA3->CA1 allows reconstruction of ERC pattern from partial input.
- Other connections shown in black are part of the slow-learning neocortical network.
- Recurrence within CA3, through the hippocampal circuit shown, and through the outer loop also involving the rest of the neocortex

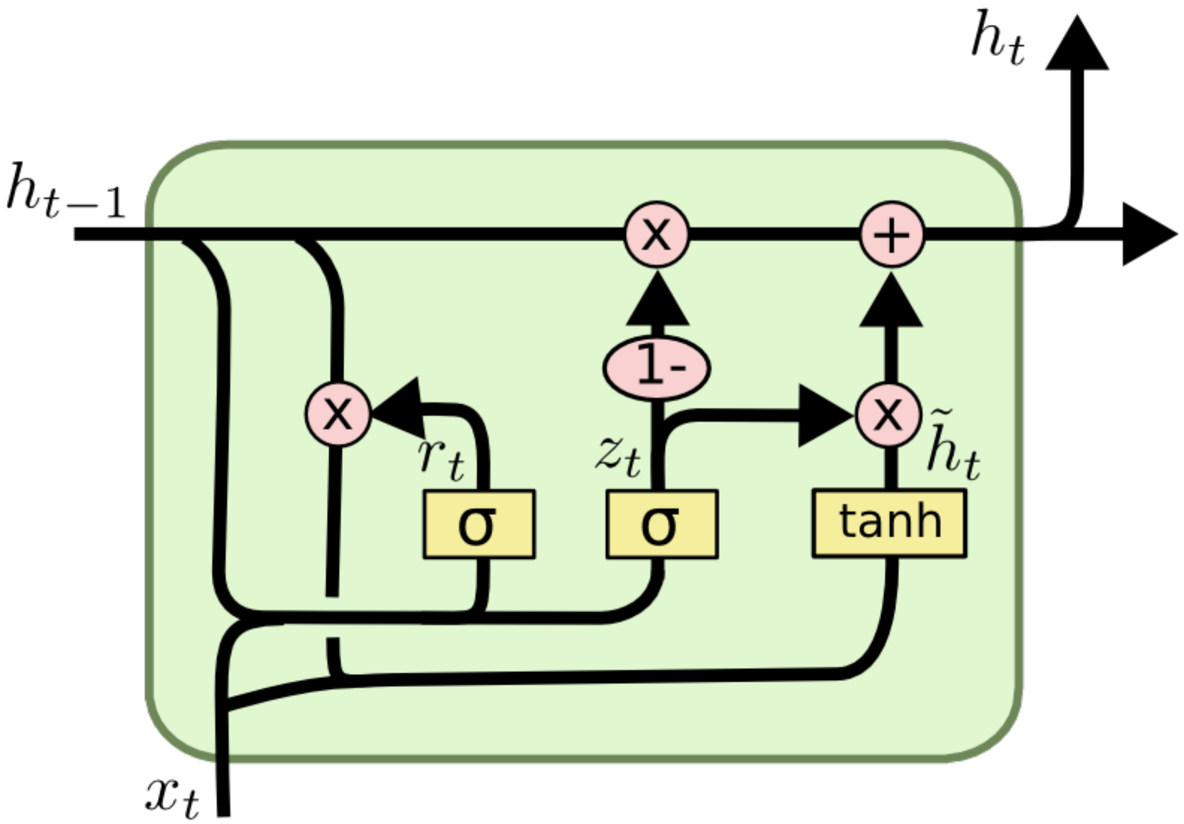


from Kumaran, Hassabis & McClelland (2016)

Long-Short Term Memory (LSTMs)



Gated Recurrent Unit (GRU) combines forget and input gate:

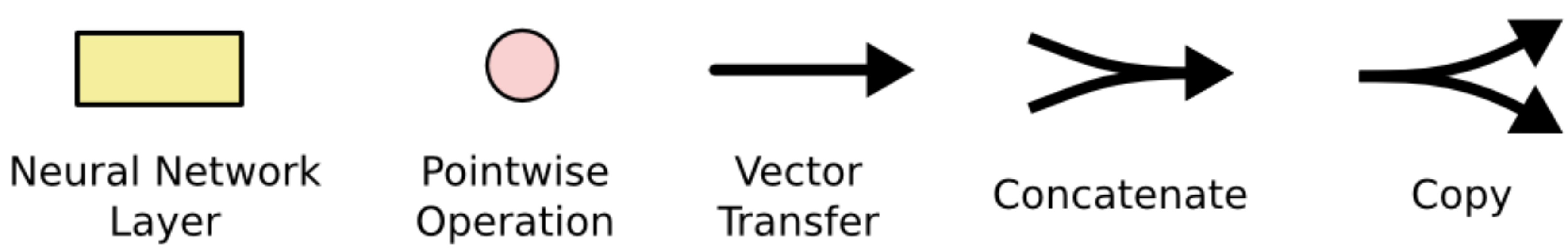


$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

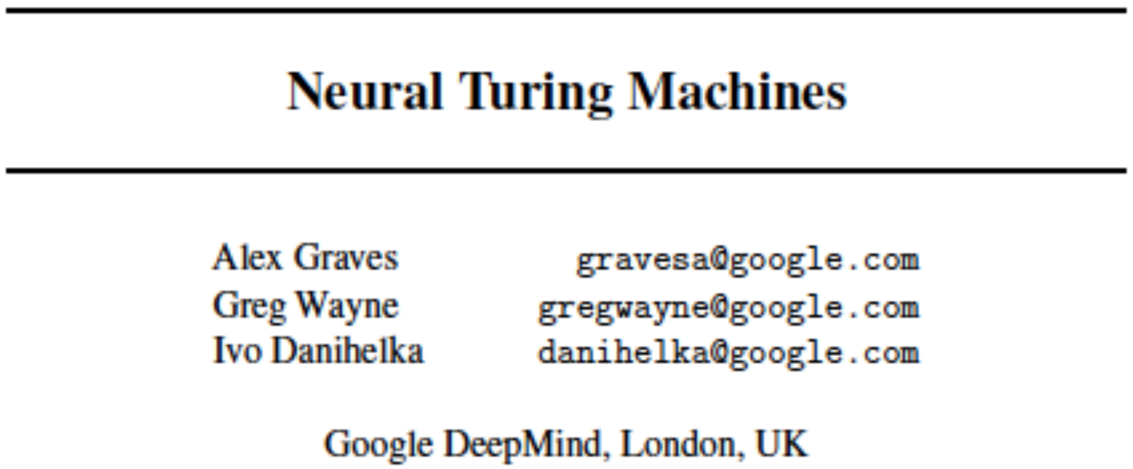


Models: RNNs

LSTMs and GRUs very useful in speech recognition and a variety of other problems with long time-scale dependencies.

. . . but have some severe limitations.

Models: Deep Networks with External Memory Stores

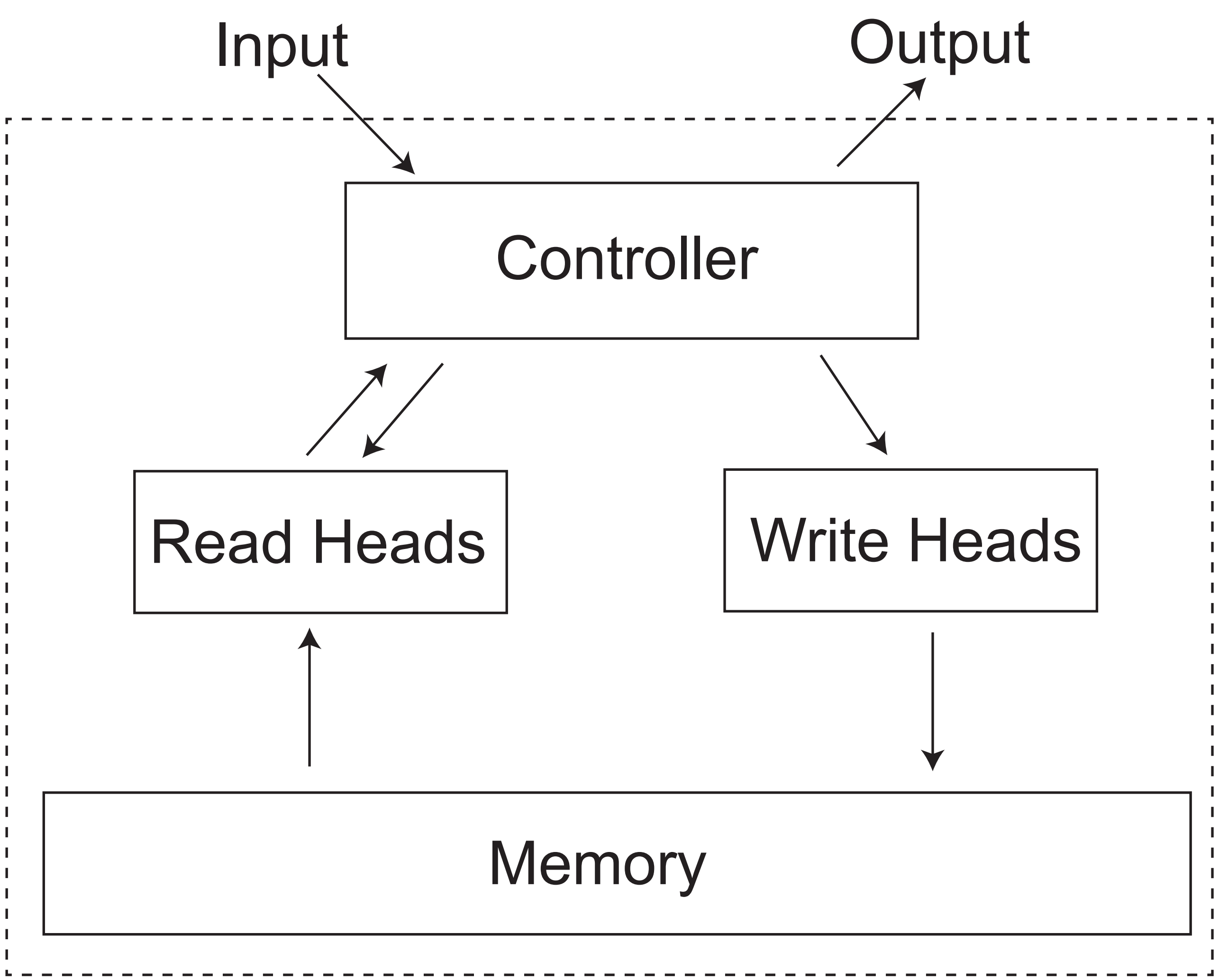


Neural Turing Machines

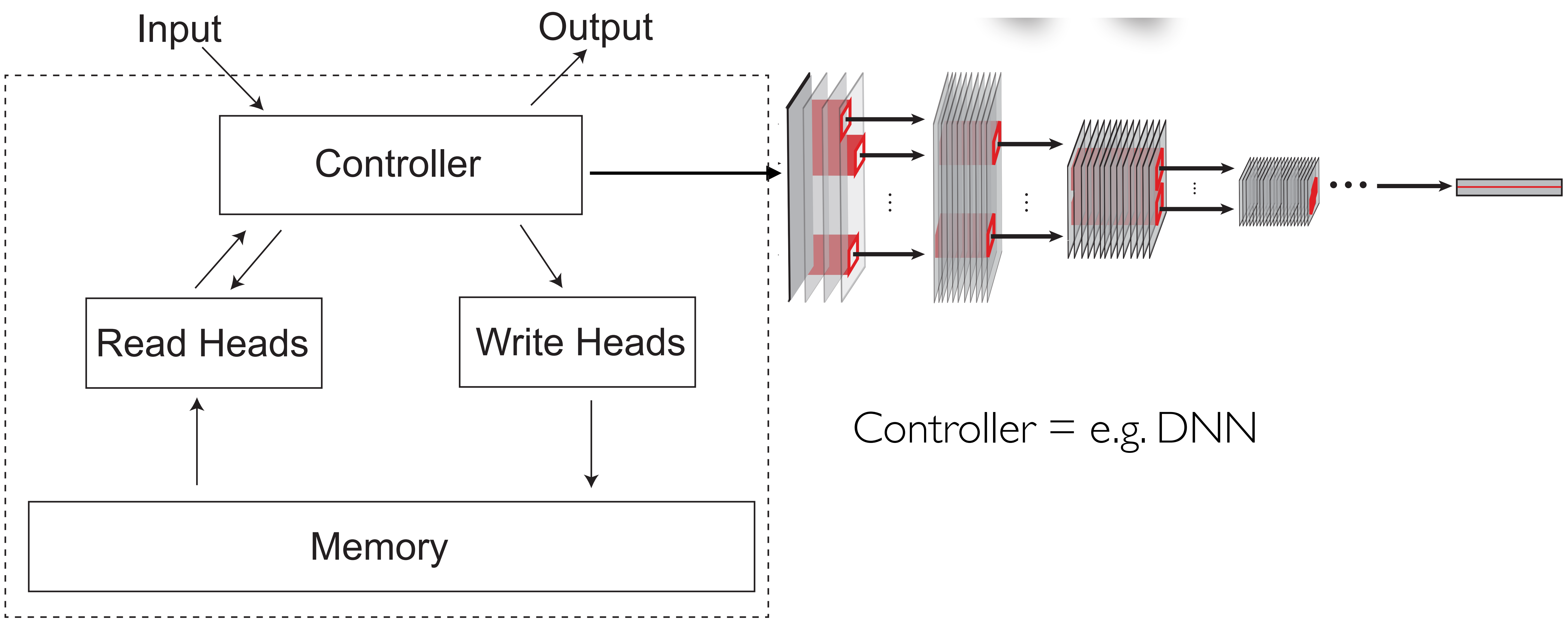
Alex Graves gravesa@google.com
Greg Wayne gregwayne@google.com
Ivo Danihelka danihelka@google.com

Google DeepMind, London, UK

Approach: Learn parameters (weights on read / write heads) via gradient descent

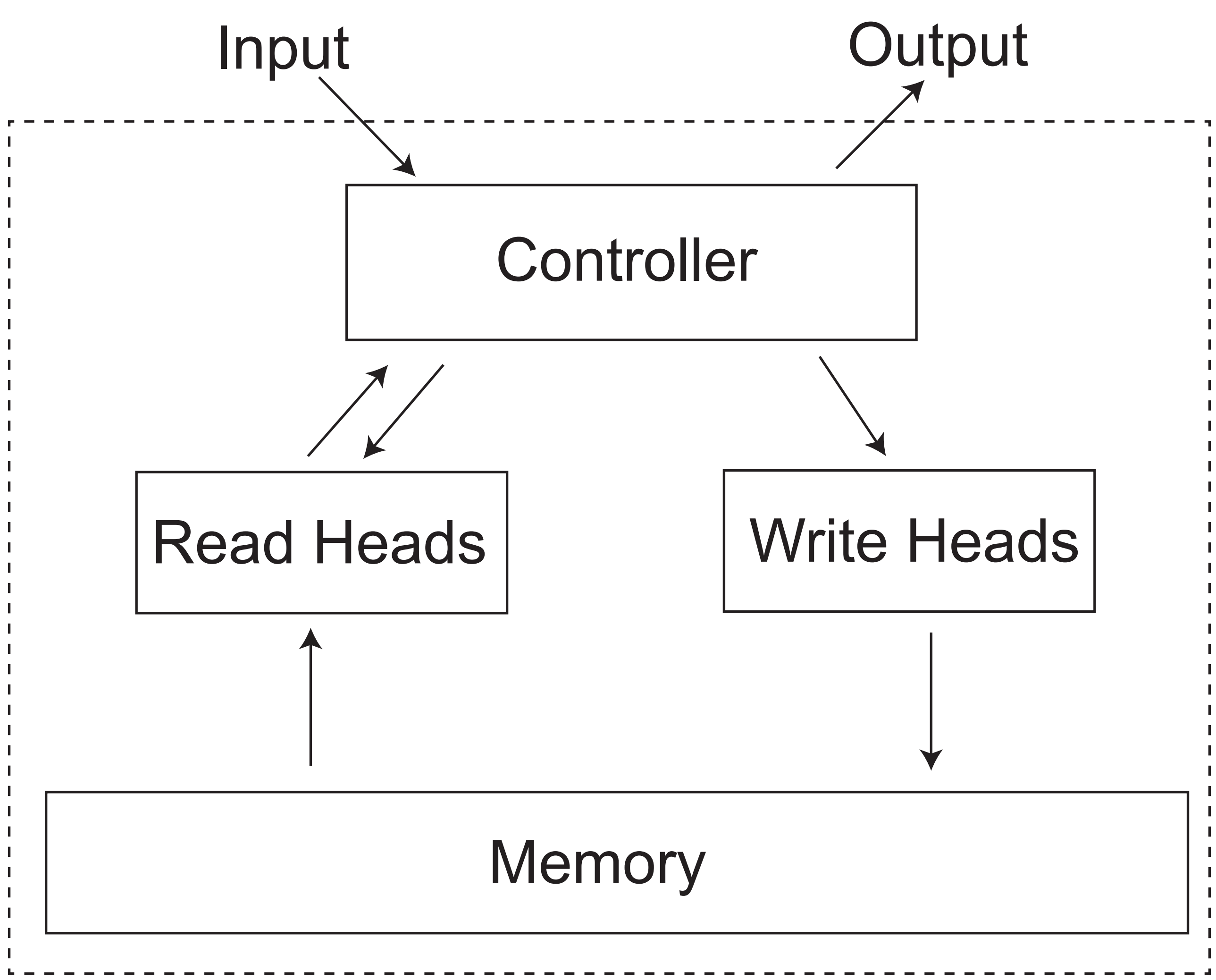


Models: Deep Networks with External Memory Stores



Controller = e.g. DNN

Models: Deep Networks with External Memory Stores



Reading: getting / interpretation memories from memory bank (RAM)

Writing: changing the contents of the memory bank as a function of what's in the active focus

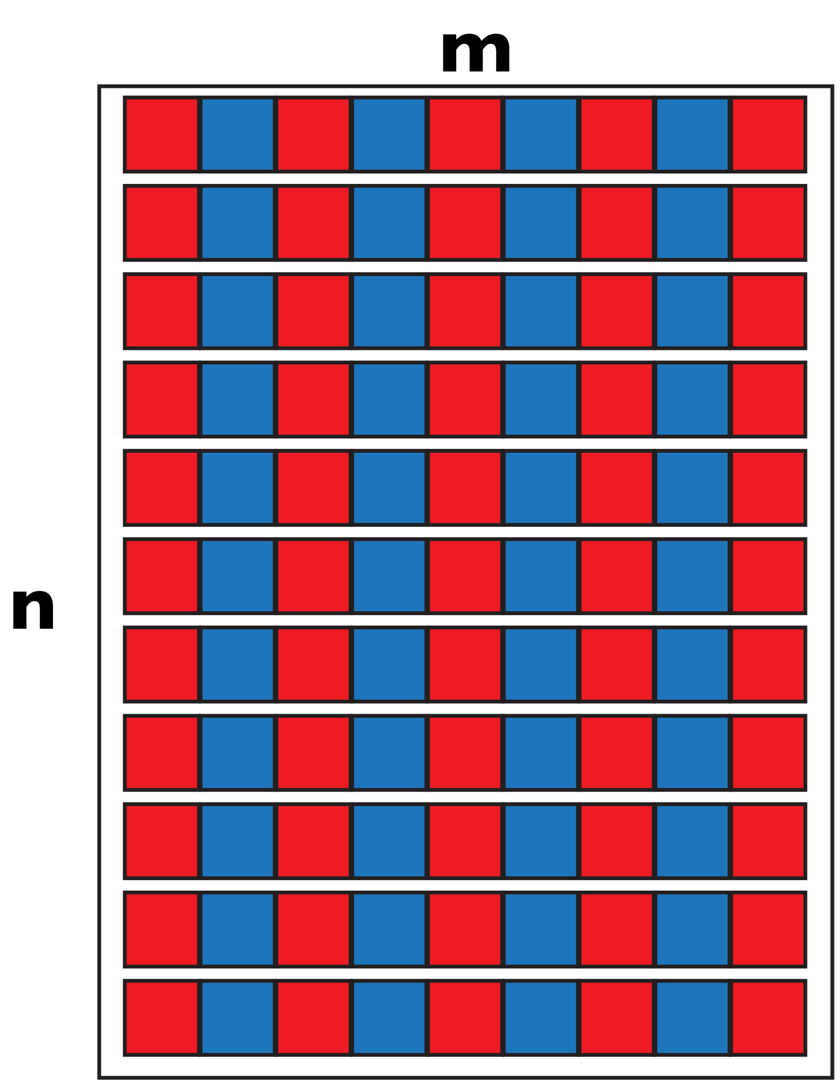
Addressing: changing the connection between the active focus and the memory bank

Models: Deep Networks with External Memory Stores

1. Reading

n = number of memory vectors

m = memory vector length



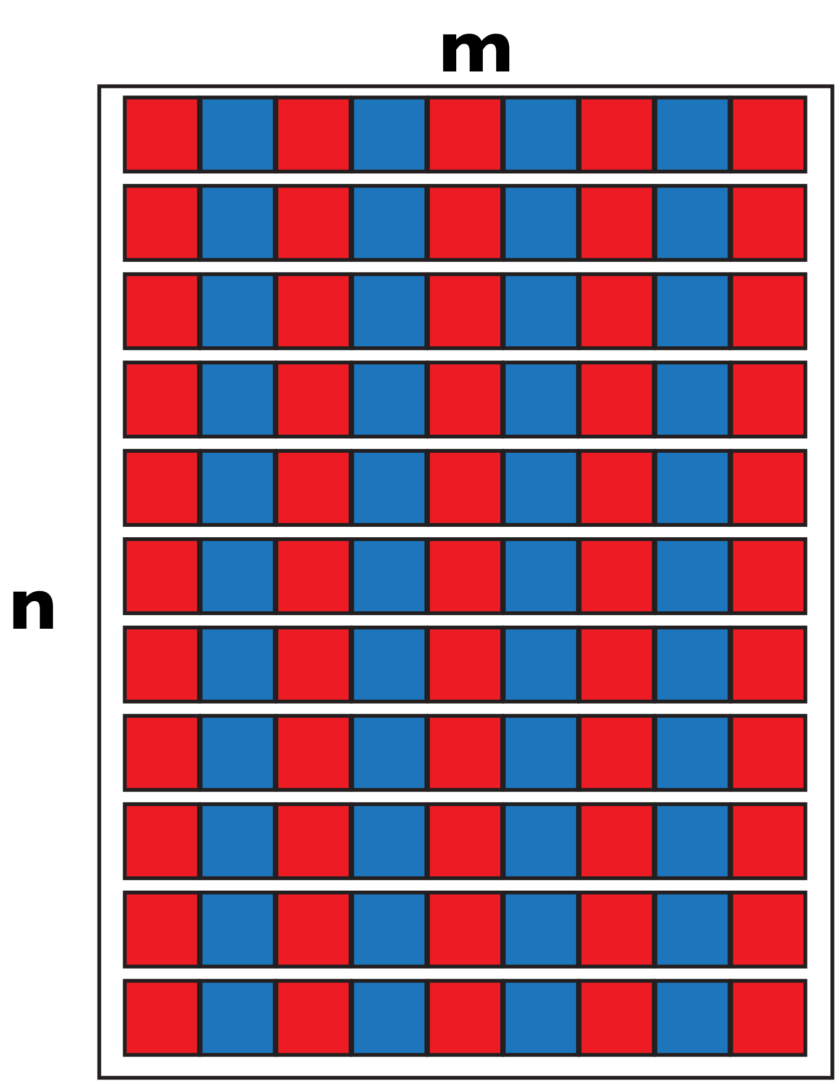
M_t = ***n*** × ***m*** memory matrix
at time ***t***

Models: Deep Networks with External Memory Stores

I. Reading

n = number of memory vectors

m = memory vector length



M_t = ***n*** × ***m*** memory matrix at time ***t***

w_t **=** <u>read weight vector</u> of length **n**

$$w_t(i) \geq 0 \qquad \sum_i w_t(i) = 1$$

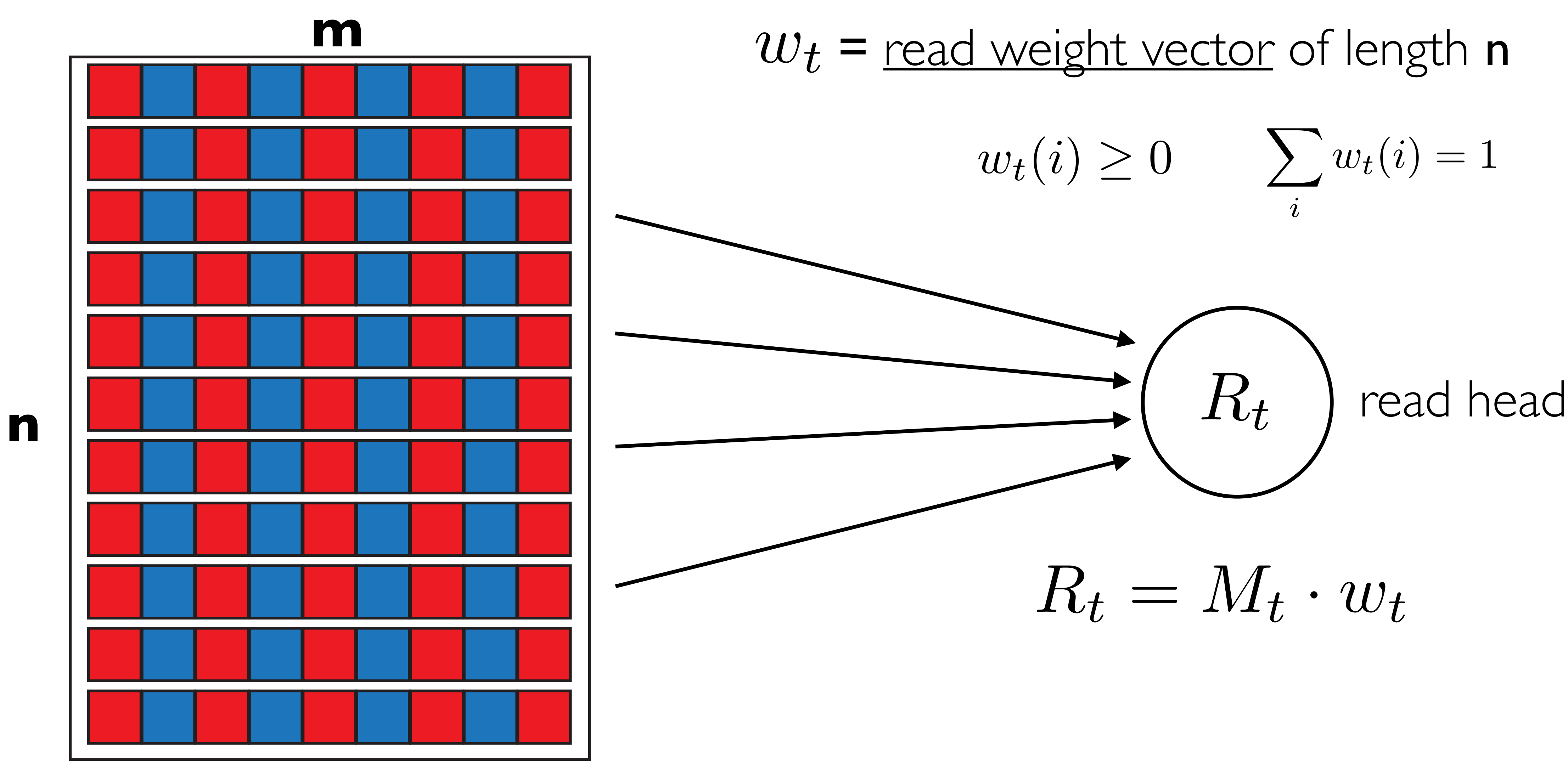
I. Reading

n = number of memory vectors

m = memory vector length

w_t = <u>read weight vector</u> of length **n**

$w_t(i) \geq 0 \qquad \sum_i w_t(i) = 1$



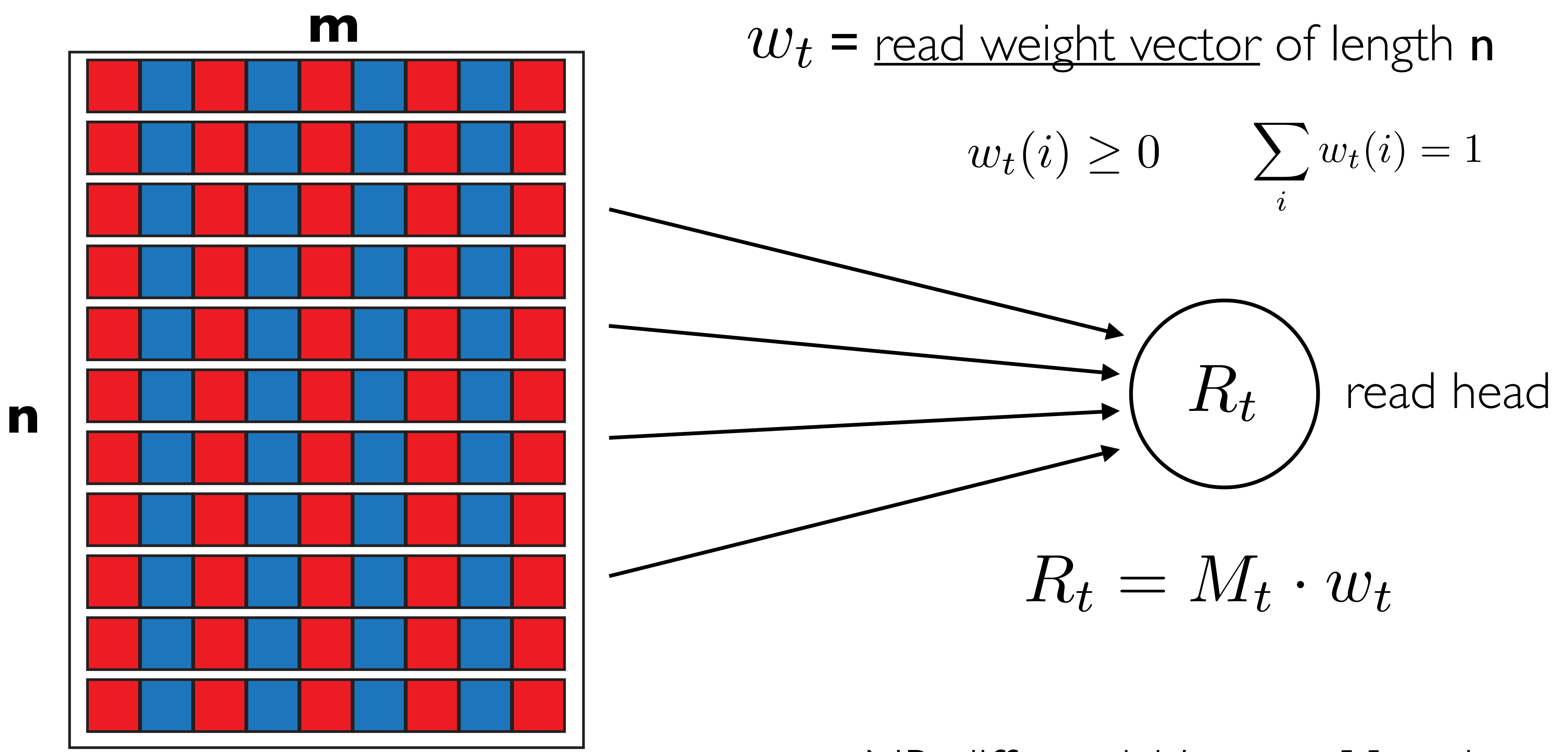
$$R_t = M_t \cdot w_t$$

M_t = ***n*** × ***m*** memory matrix at time ***t***

1. Reading

n = number of memory vectors

m = memory vector length



w_t = read weight vector of length **n**

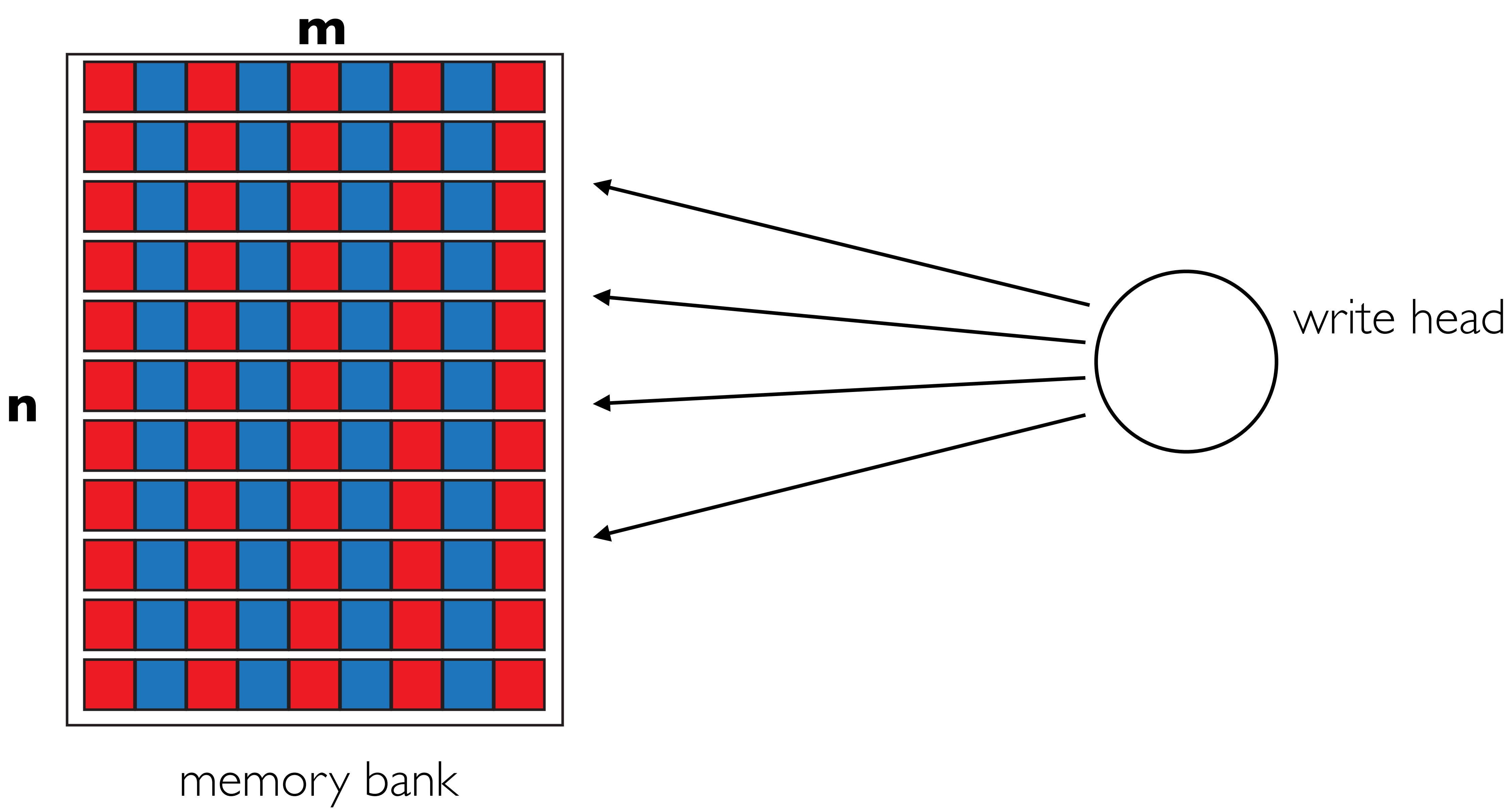
$$w_t(i) \geq 0 \qquad \sum_i w_t(i) = 1$$

$$R_t = M_t \cdot w_t$$

M_t = ***n*** × ***m*** memory matrix at time ***t***

NB: differentiable w.r.t. ***M*** and ***w***

2. Writing



Models: Deep Networks with External Memory Stores

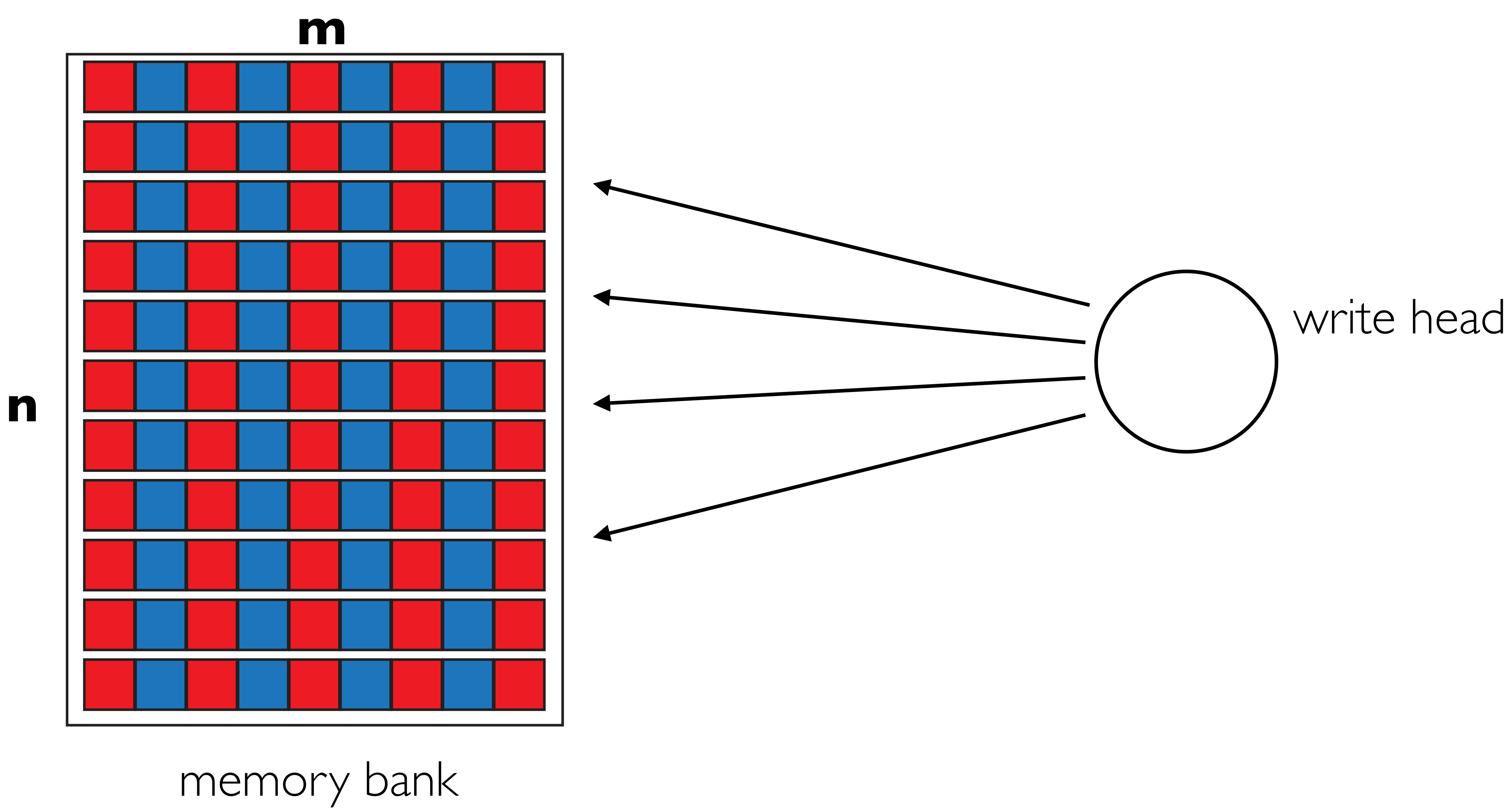
2. Writing

w_t = write weights of length n

e_t = erase vector of length m

a_t = add vector of length m

$$e_t(j), a_j(t) \in [0, 1], 0 \le j < m - 1$$



Models: Deep Networks with External Memory Stores

2. Writing

$\boldsymbol{w_t}$ = write weights of length $\boldsymbol{n}$

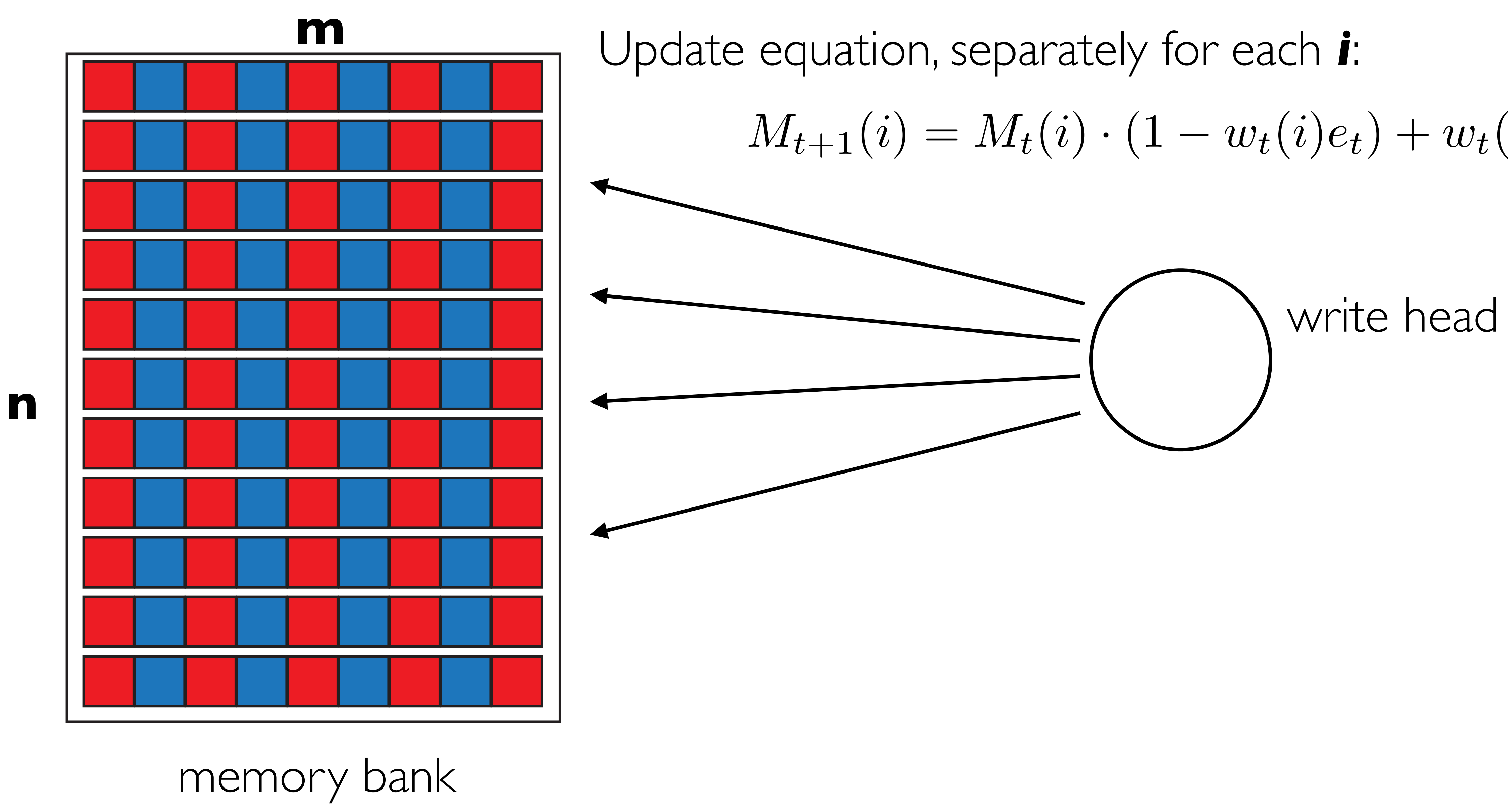
$\mathbf{e_t}$ = erase vector of length $\boldsymbol{m}$

$\boldsymbol{a_t}$ = add vector of length $\boldsymbol{m}$

$$e_t(j), a_j(t) \in [0,1], 0 \leq j < m-1$$

Update equation, separately for each $\boldsymbol{i}$:

$$M_{t+1}(i) = M_t(i) \cdot (1 - w_t(i)e_t) + w_t(i)a_t$$



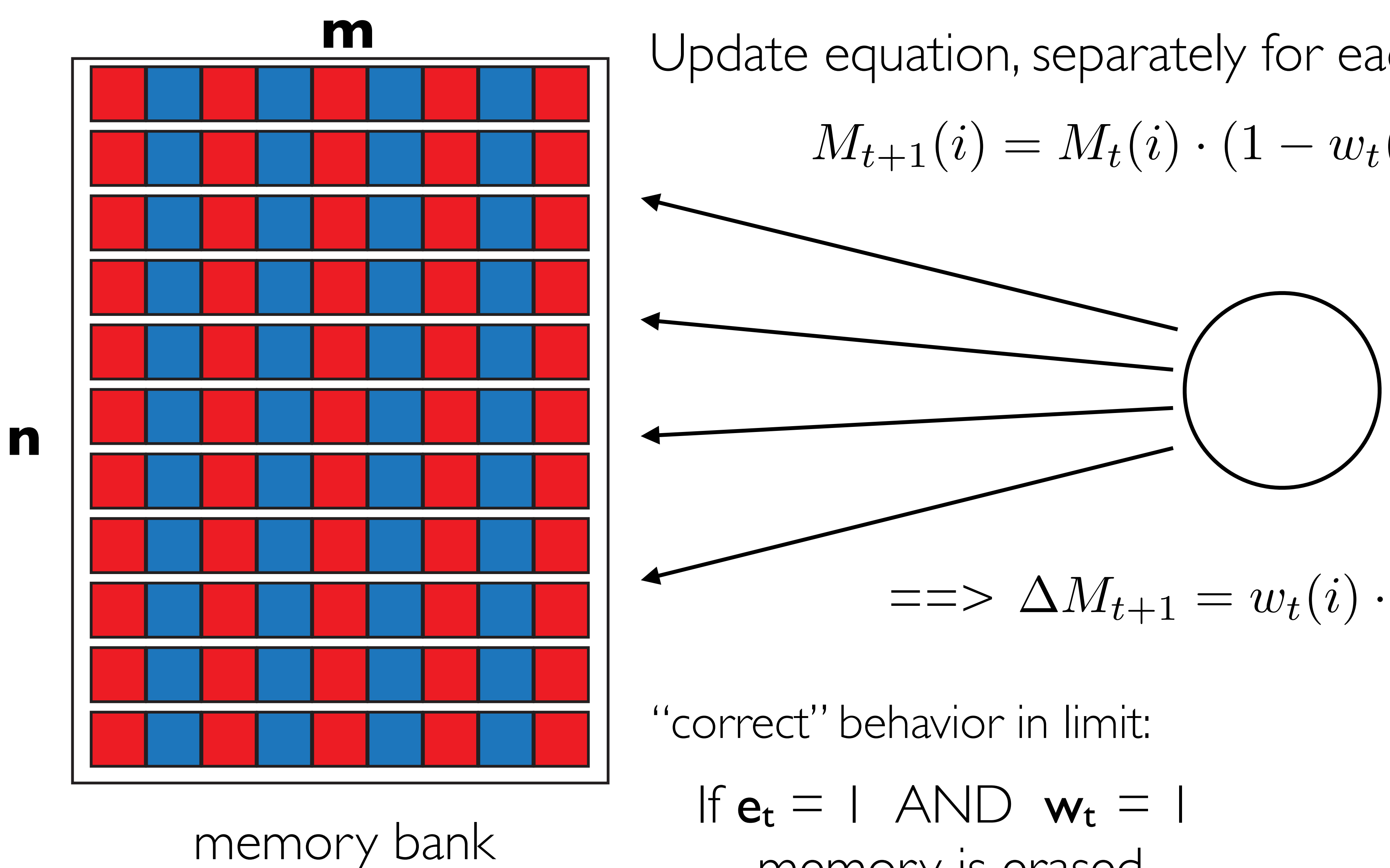
2. Writing

w_t = write weights of length n

e_t = erase vector of length m

a_t = add vector of length m

$$e_t(j), a_j(t) \in [0,1], 0 \leq j < m-1$$



Update equation, separately for each ***i***:

$$M_{t+1}(i) = M_t(i) \cdot (1 - w_t(i)e_t) + w_t(i)a_t$$

$$==> \Delta M_{t+1} = w_t(i) \cdot (a_t - e_t M_t)$$

"correct" behavior in limit:

If $\mathbf{e_t}$ = 1 AND $\mathbf{w_t}$ = 1
memory is erased

Models: Deep Networks with External Memory Stores

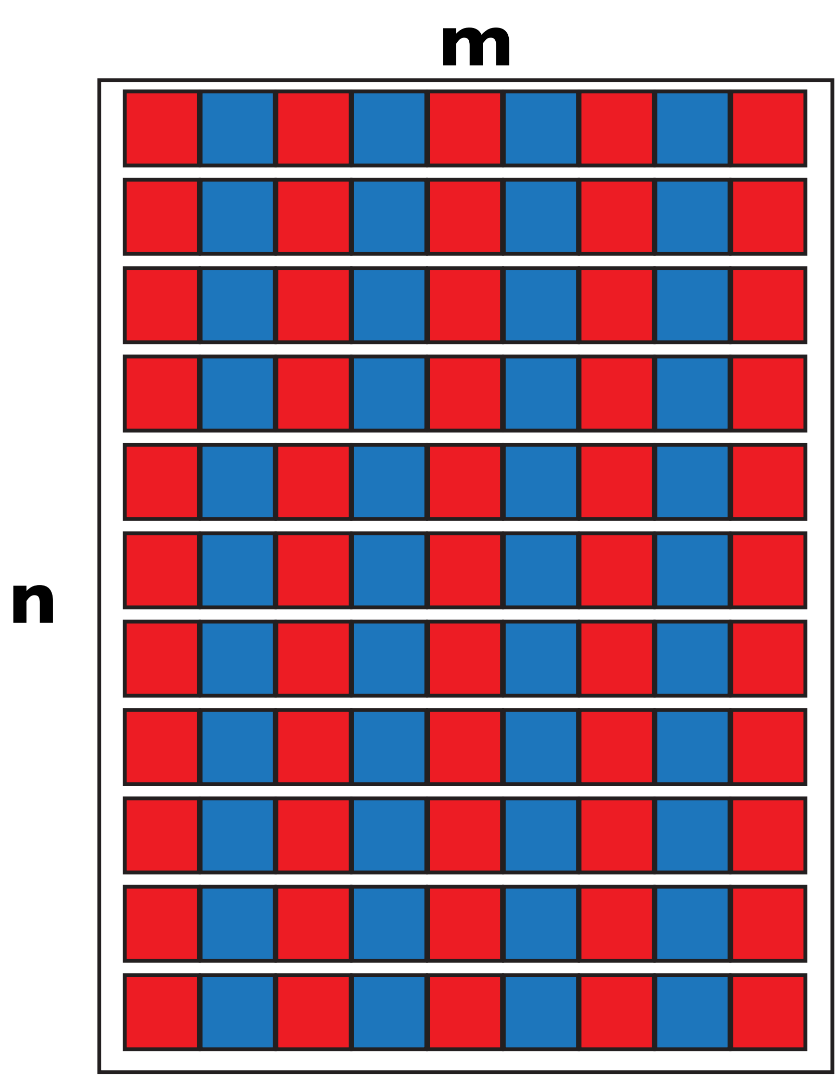
2. Writing

$\boldsymbol{w_t}$ = write weights of length $\boldsymbol{n}$

$\mathbf{e_t}$ = erase vector of length $\boldsymbol{m}$

$\boldsymbol{a_t}$ = add vector of length $\boldsymbol{m}$

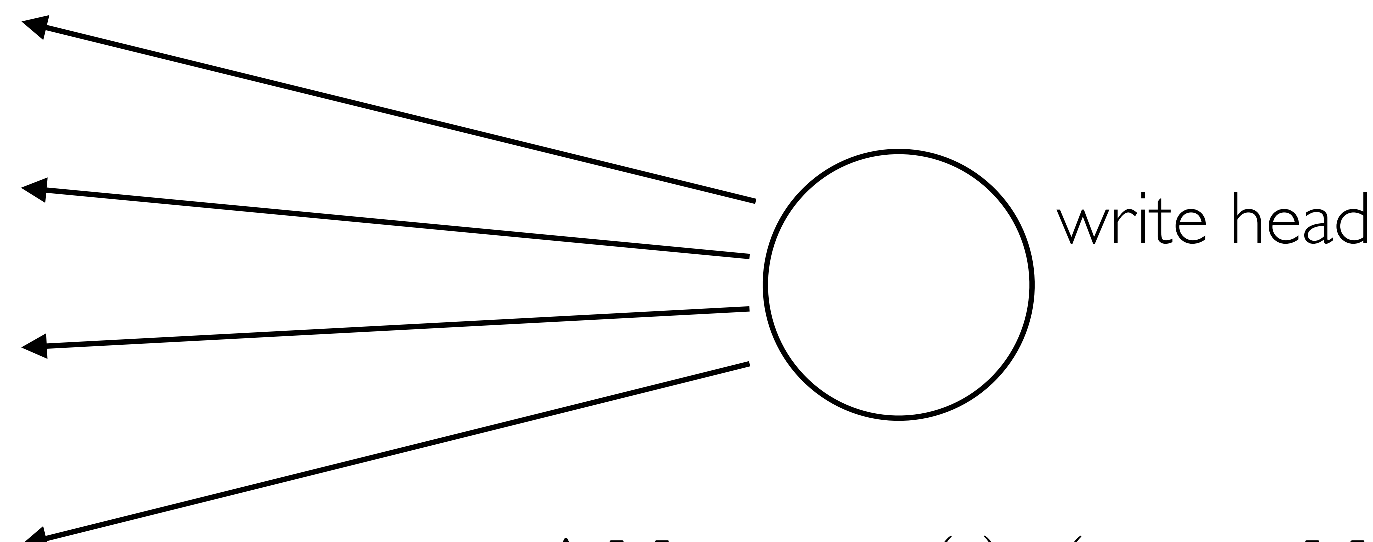
$e_t(j), a_j(t) \in [0,1], 0 \leq j < m-1$



memory bank

Update equation, separately for each $\boldsymbol{i}$:

$$M_{t+1}(i) = M_t(i) \cdot (1 - w_t(i)e_t) + w_t(i)a_t$$



$$==> \Delta M_{t+1} = w_t(i) \cdot (a_t - e_t M_t)$$

"correct" behavior in limit:

If $\mathbf{e_t}$ = 1 AND $\mathbf{w_t}$ = 1
memory is erased

If $\mathbf{e_t}, \mathbf{a_t}$ = 0 OR $\mathbf{w_t}$ = 0
memory is unchanged

Models: Deep Networks with External Memory Stores

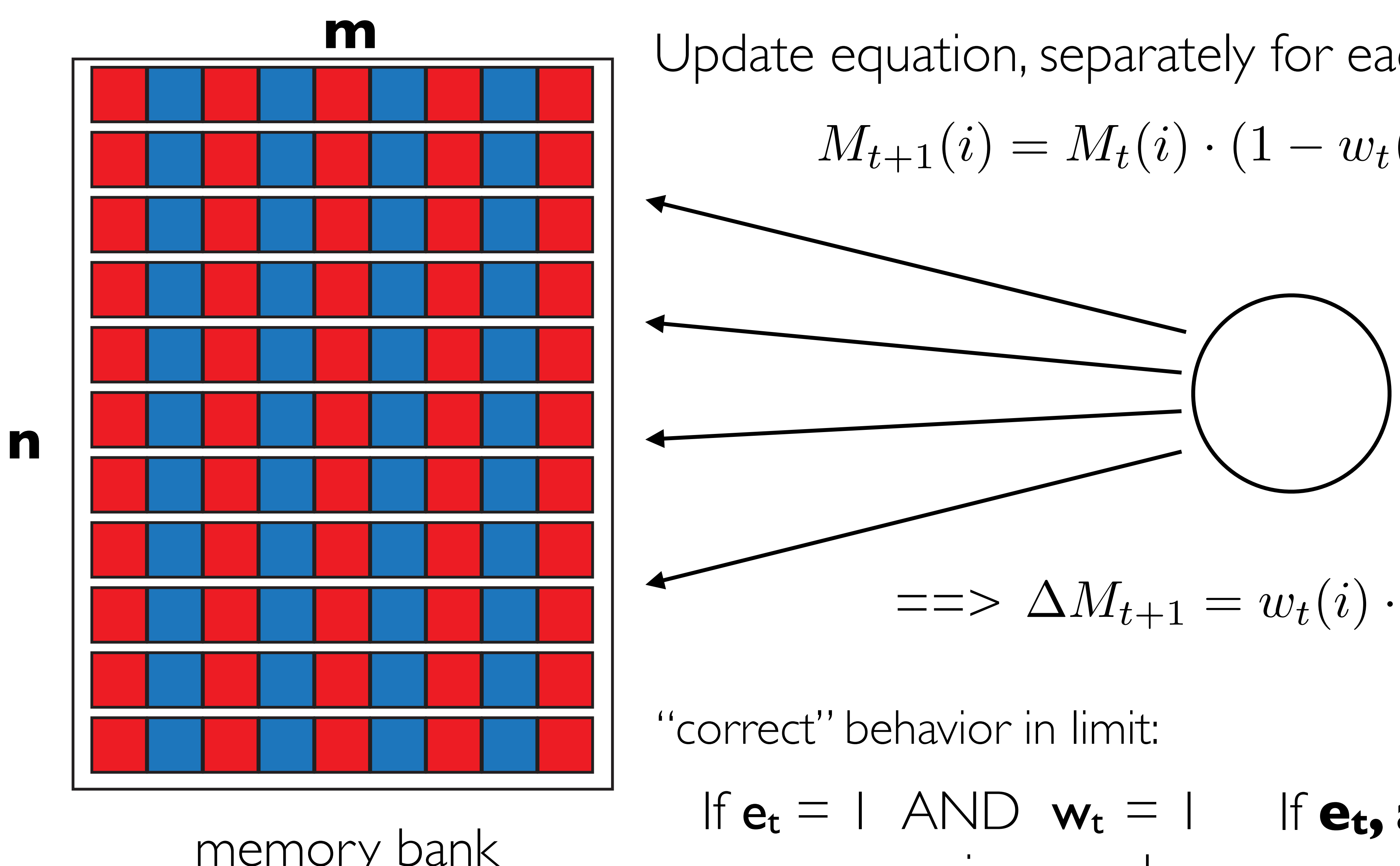
2. Writing

w_t = write weights of length n

e_t = erase vector of length m

a_t = add vector of length m

$e_t(j), a_j(t) \in [0,1], 0 \leq j < m-1$



Update equation, separately for each i:

$$M_{t+1}(i) = M_t(i) \cdot (1 - w_t(i)e_t) + w_t(i)a_t$$

$$\Longrightarrow \Delta M_{t+1} = w_t(i) \cdot (a_t - e_t M_t)$$

"correct" behavior in limit:

If $\mathbf{e_t} = 1$ AND $\mathbf{w_t} = 1$
memory is erased

If $\mathbf{e_t}, \mathbf{a_t} = 0$ OR $\mathbf{w_t} = 0$
memory is unchanged

formula smoothly interpolates two cases above

Models: Deep Networks with External Memory Stores

But where do the $\boldsymbol{w_t}$ come from? **3. Addressing**

Models: Deep Networks with External Memory Stores

But where do the $\boldsymbol{w_t}$ come from? **3. Addressing**

John Hopfield

1. Content Addressing: Hopfield Networks (Hopfield, 1982)
Shiffrin, Hintzman (Minerva II, 1984)

address == similarity between
controller (DNN)
and
memory

Models: Deep Networks with External Memory Stores

But where do the $\boldsymbol{w_t}$ come from? **3. Addressing**

John Hopfield

1. Content Addressing: Hopfield Networks (Hopfield, 1982)
Shiffrin, Hintzman (Minerva II, 1984)

address == similarity between
controller (DNN)
and
memory

$$w_t^c(i) = \beta_t \cdot \langle k_t, M_t(i) \rangle$$

β_t = coupling strength

k_t = output from the controller (DNN)

$\langle \cdot, \cdot \rangle$ = similarity measure e.g. cosine distance

Models: Deep Networks with External Memory Stores

But where do the $\boldsymbol{w_t}$ come from? **3. Addressing**

John Hopfield

1. Content Addressing: Hopfield Networks (Hopfield, 1982)
Shiffrin, Hintzman (Minerva II, 1984)

address == similarity between
controller (DNN)
and
memory

$$w_t^c(i) = \textbf{Softmax}(\beta_t \cdot \langle k_t, M_t(i) \rangle)$$

β_t = coupling strength

k_t = output from the controller (DNN)

$\langle \cdot, \cdot \rangle$ = similarity measure e.g. cosine distance

Models: Deep Networks with External Memory Stores

But where do the $\boldsymbol{w_t}$ come from? **3. Addressing**

1. Content Addressing: Hopfield Networks (Hopfield, 1982)
Shiffrin, Hintzman (Minerva II, 1984)

John Hopfield

address == similarity between
controller (DNN)
and
memory

$$w_t^c(i) = \frac{exp(\beta_t \cdot \langle k_t, M_t(i) \rangle)}{\sum_{\boxed{j}} exp(\beta_t \cdot \langle k_t, M_t(j) \rangle)}$$

competition between memories

β_t = coupling strength

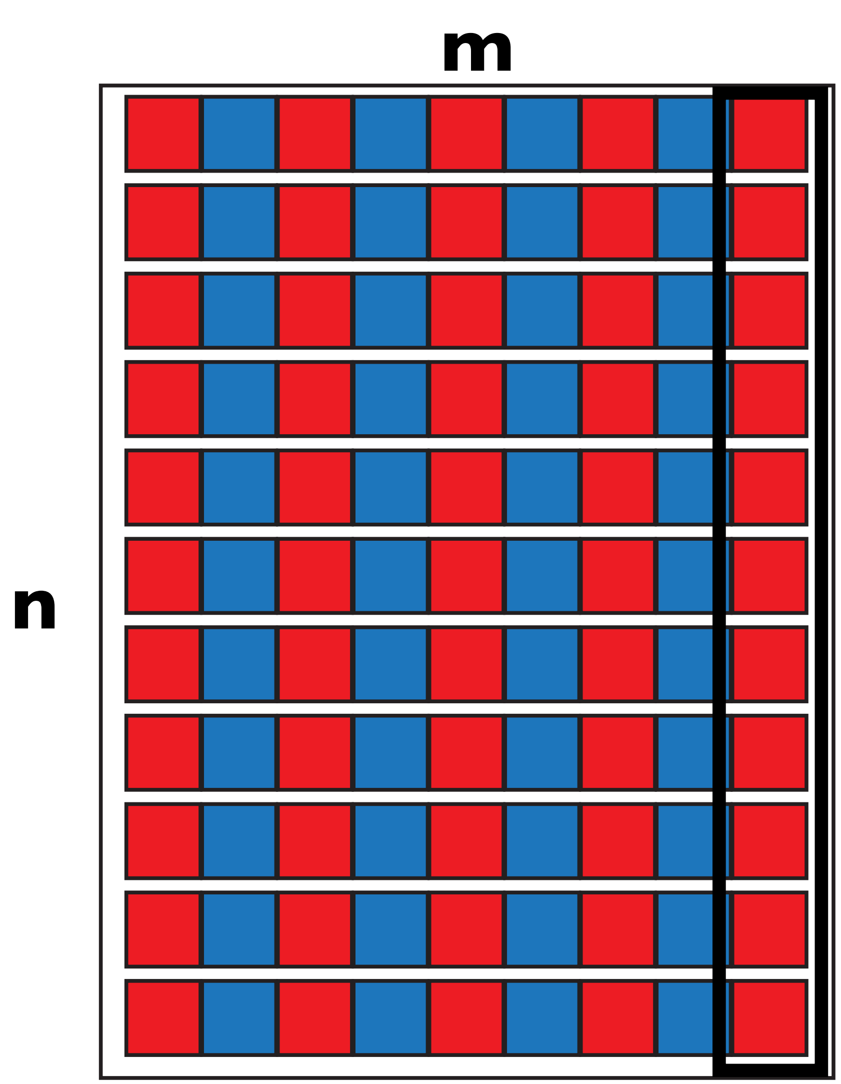
k_t = output from the controller (DNN)

$\langle \cdot, \cdot \rangle$ = similarity measure e.g. cosine distance

Models: Deep Networks with External Memory Stores

But where do the $\boldsymbol{w_t}$ come from? **3. Addressing**

2. Location-based addressing:



memory bank

convolution over vertical axis, separately for each column

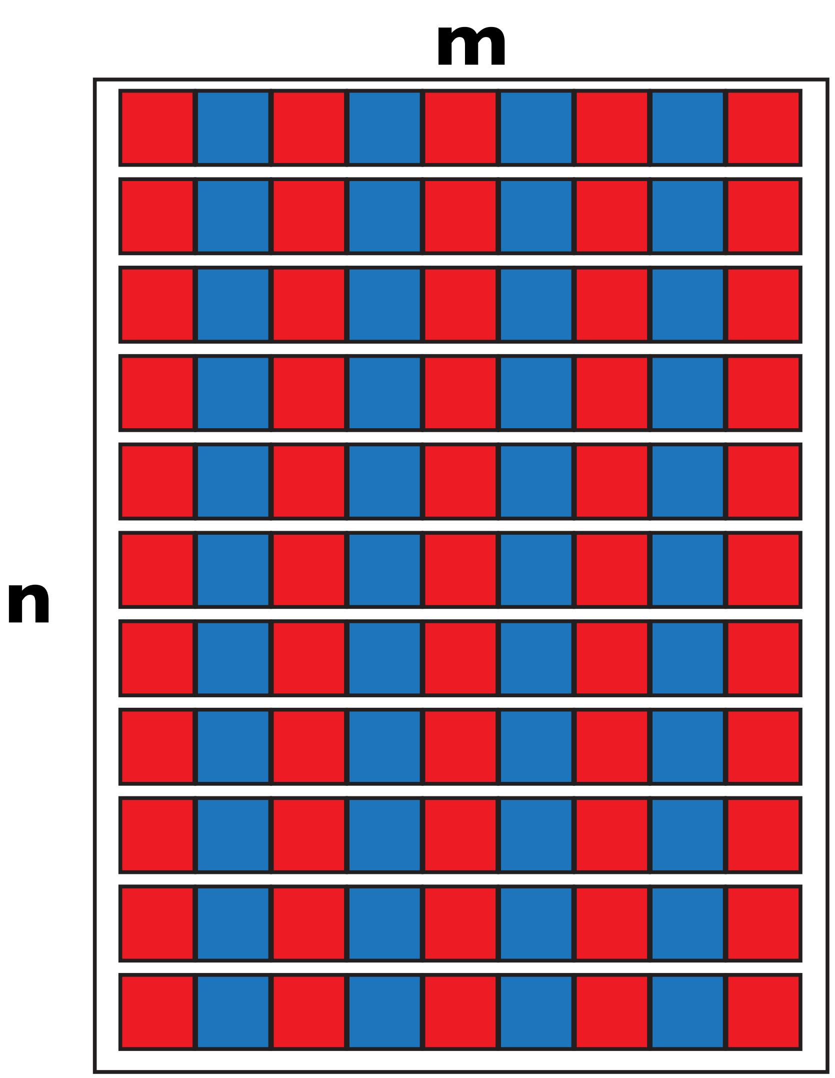
$$w_t(i) = \sum_{j=0}^{n-1} w_t^c(j) \cdot s_t(i-j)$$

$\boldsymbol{s_t}$ = learned probability distribution over locations

Models: Deep Networks with External Memory Stores

But where do the $\boldsymbol{w_t}$ come from?

2. Location-based addressing:



memory bank

Technically:

$$w_t^g(i) = g_t \cdot w_t^c(i) + (1 - g_t) w_{t-1}(i)$$

(gated interpolation)
$g_t \in [0, 1]$

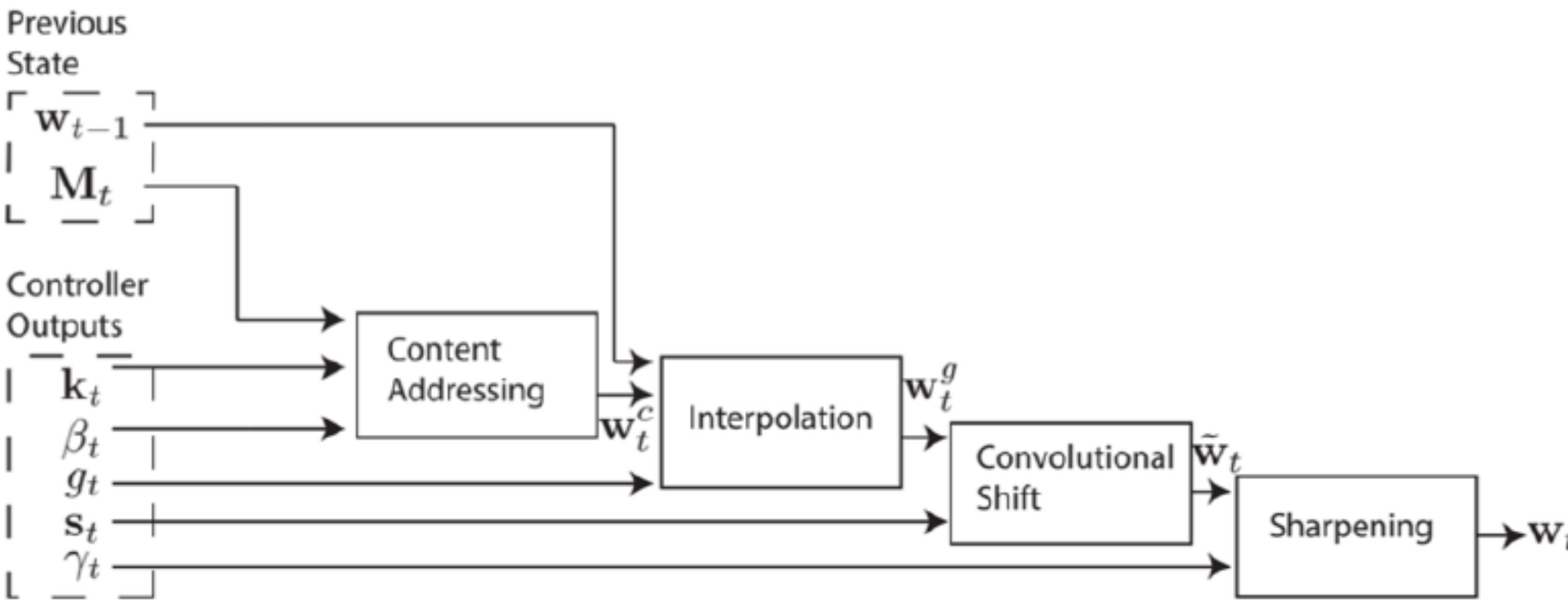
$$\tilde{w}_t(i) = \sum_{j=0}^{n-1} w_t^g(j) \cdot s_t(i - j)$$

(addressing)

$$w_t(i) = \frac{\tilde{w}_t(i)^{\gamma_t}}{\sum_j \tilde{w}_t(j)^{\gamma_t}}$$

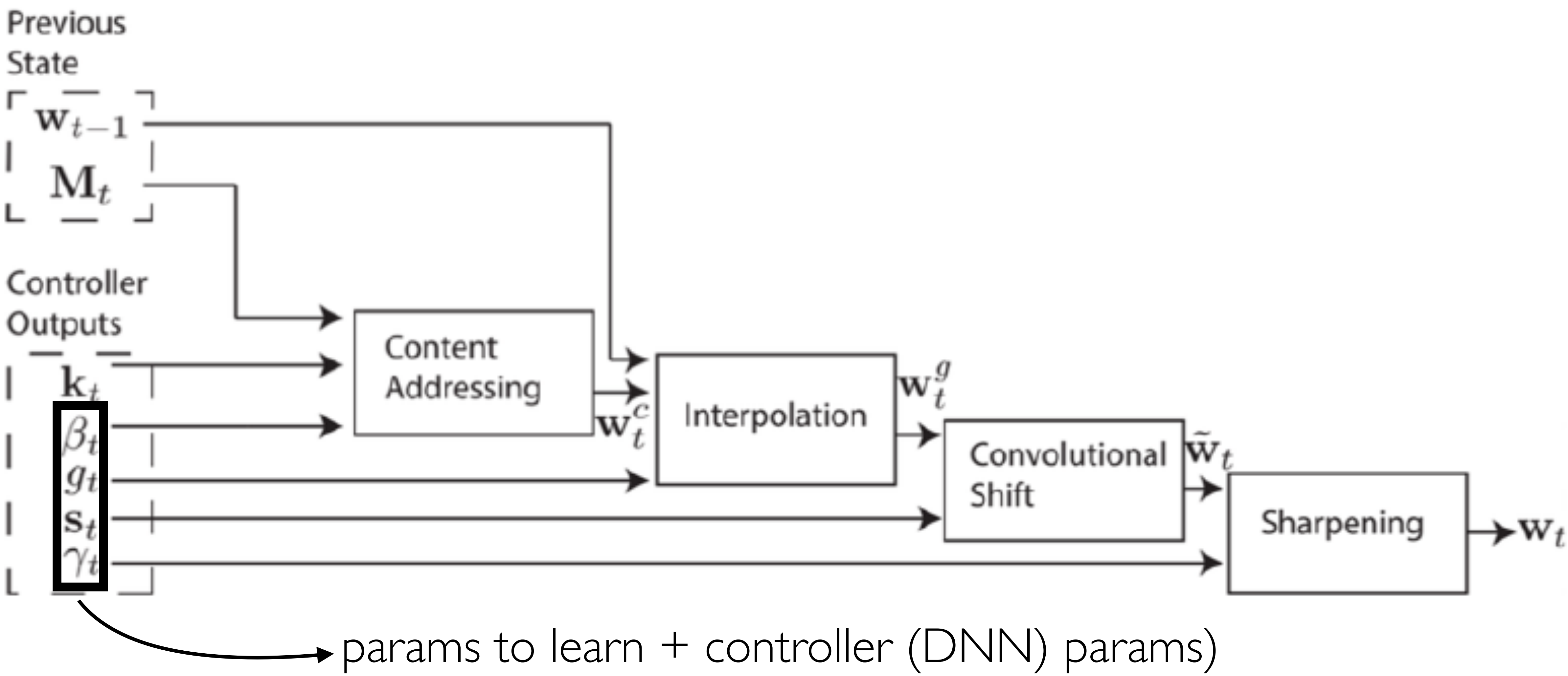
(sharpening)
$\gamma_t \geq 1$

Models: Deep Networks with External Memory Stores



NB: Location-based addressing is a special case of content addressing but easier to have explicit rather than learned.

Models: Deep Networks with External Memory Stores

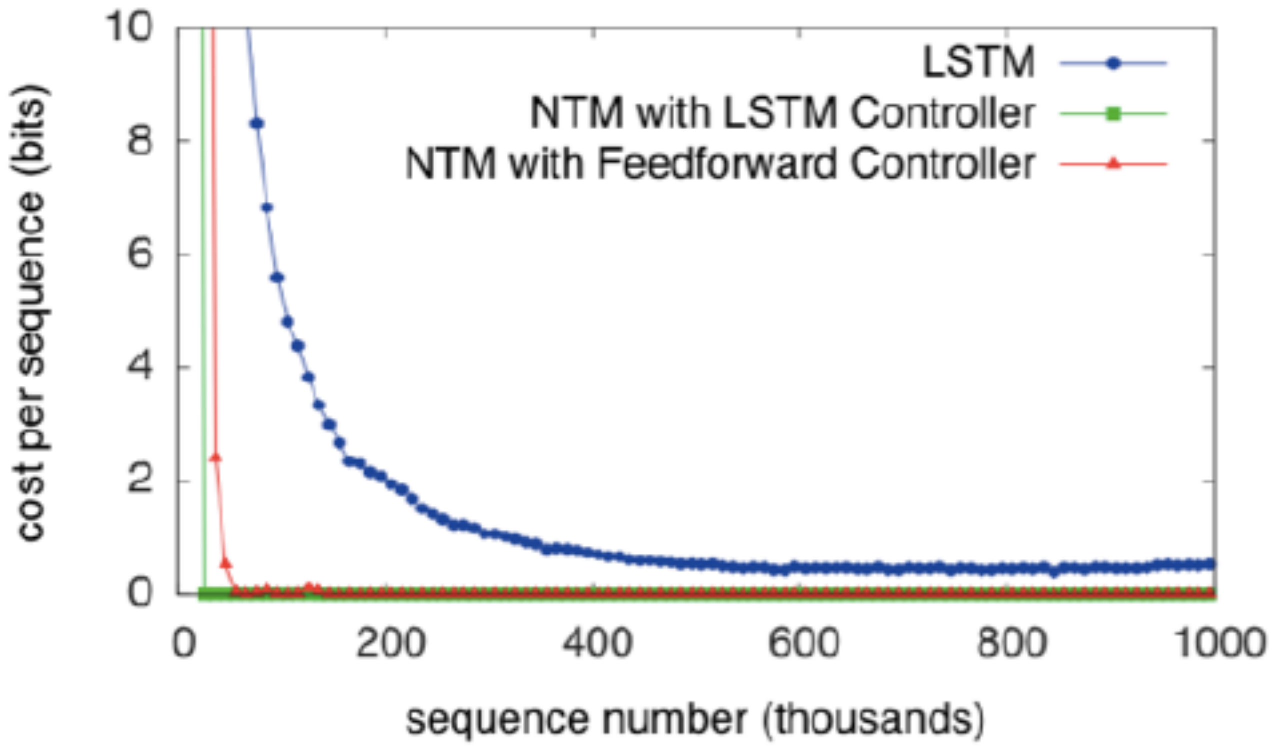


Models: Deep Networks with External Memory Stores

Exp 1: Copy

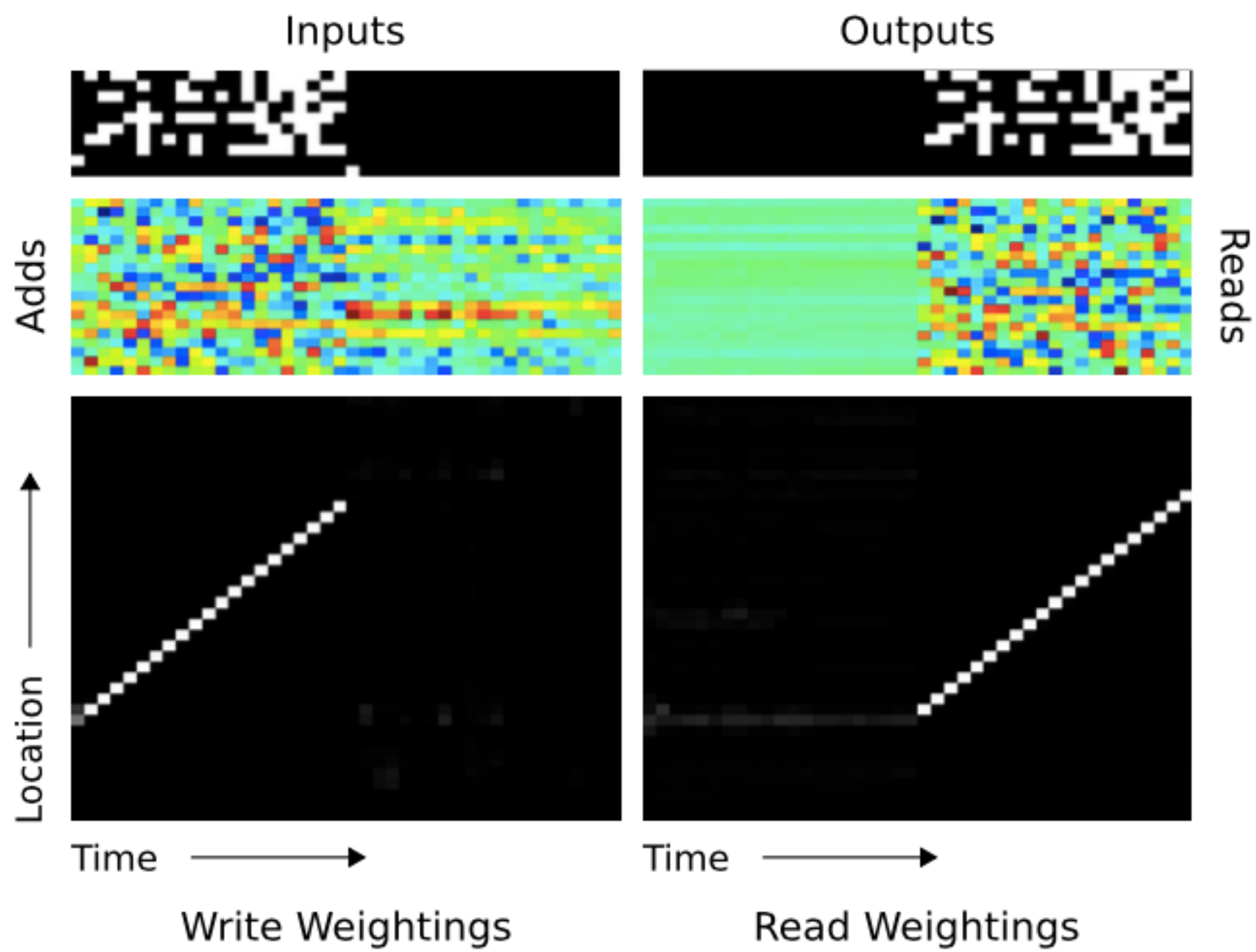
input: random binary vectors V (varying lengths) …. then delimiter

output: V



Exp 1: Copy

Memory usage patterns

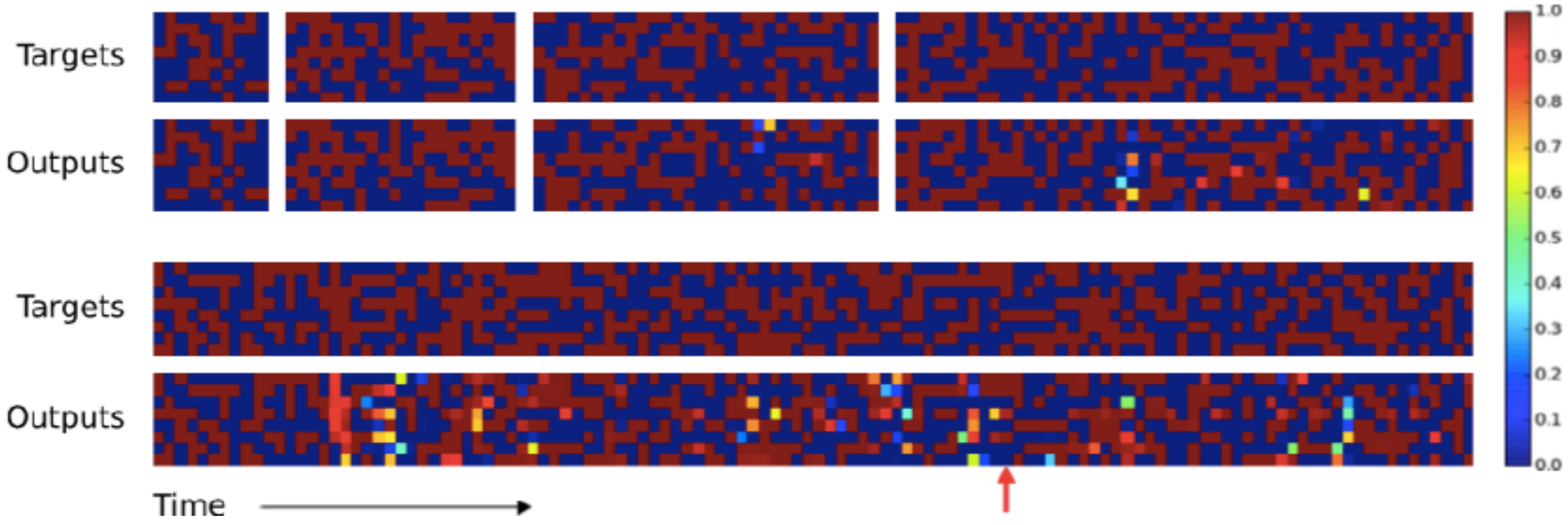


Models: Deep Networks with External Memory Stores

Exp 2: Copy Generalization

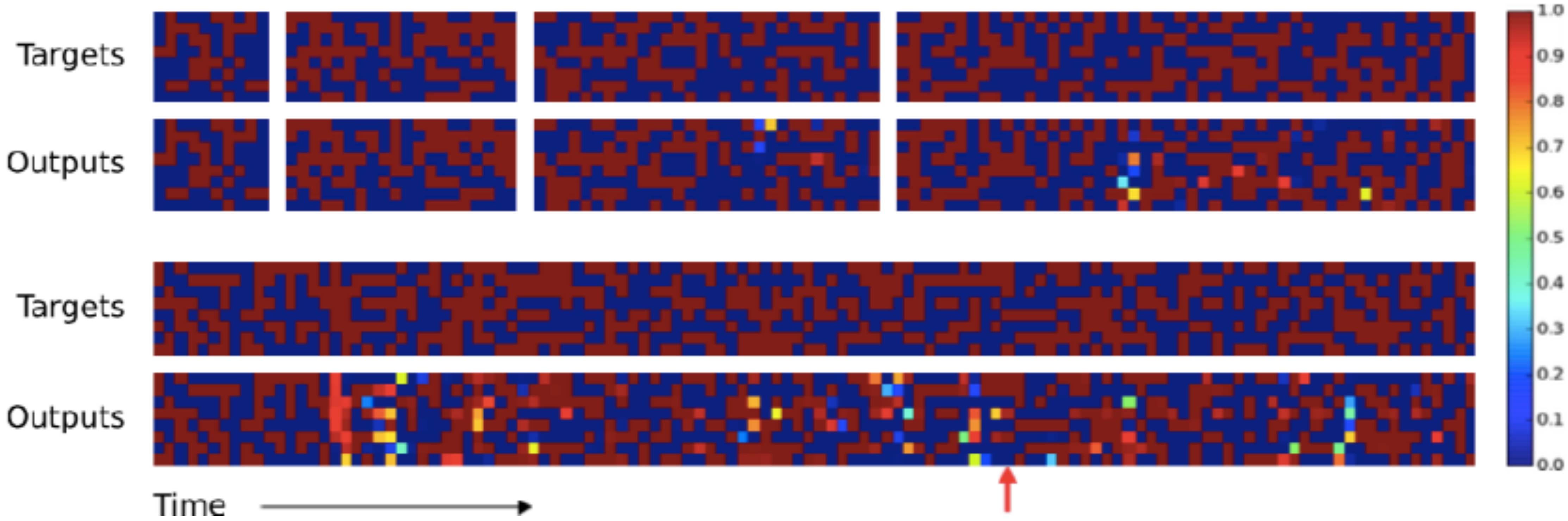
input: random binary vectors V (varying lengths <=N) …. then delimiter

output: V, but in testing lengths >= N

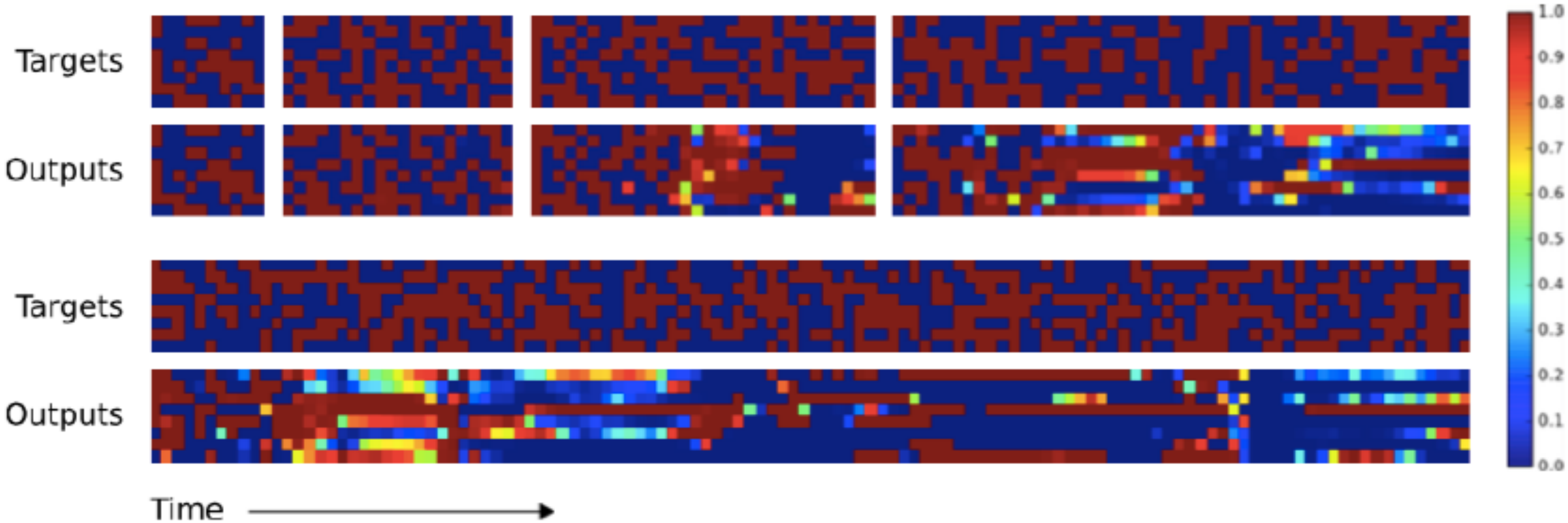


Models: Deep Networks with External Memory Stores

NTM



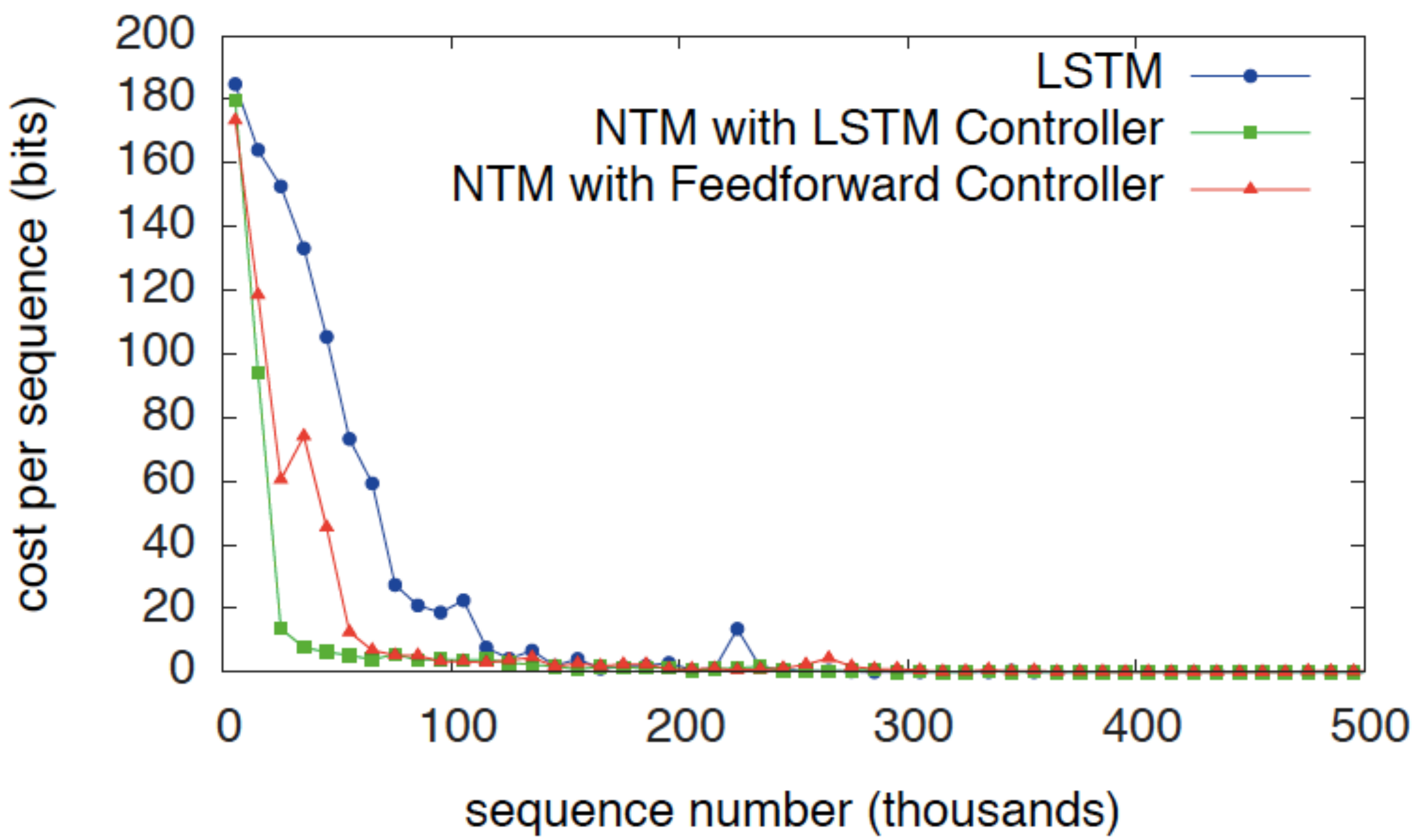
LSTM



Exp 3: Repeat Copy

input: random binary vectors V (varying lengths <=N), integer k

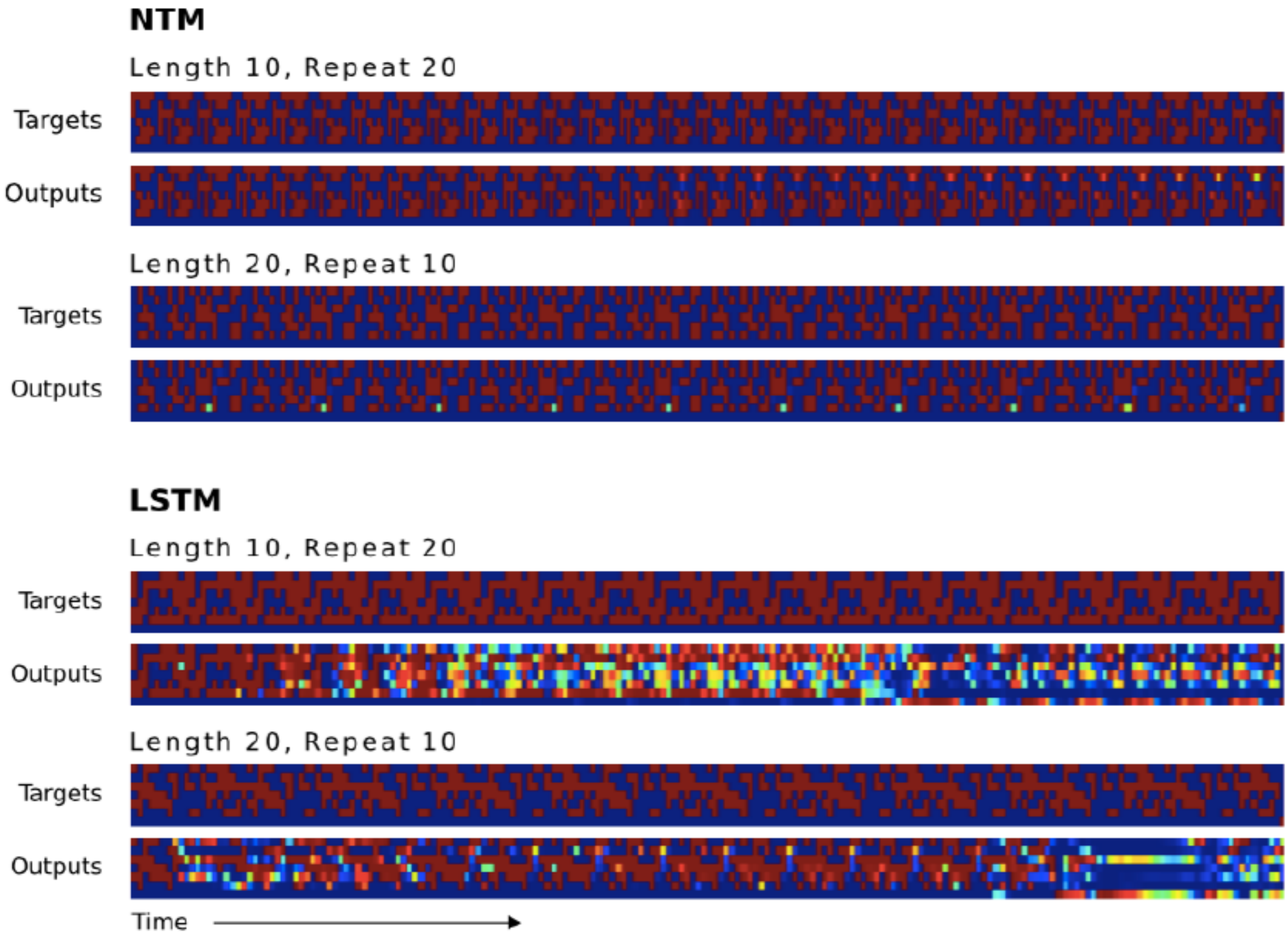
output: V repeated k times, then delimiter



it's been hard to get LSTMs to do well on this sort of task

Figure 7: Repeat Copy Learning Curves.

Models: Deep Networks with External Memory Stores



Exp 4: Associative Recall

input: start codon + sequence of items (random binary) + stop codon

testing: random element of sequence, **output:** next element

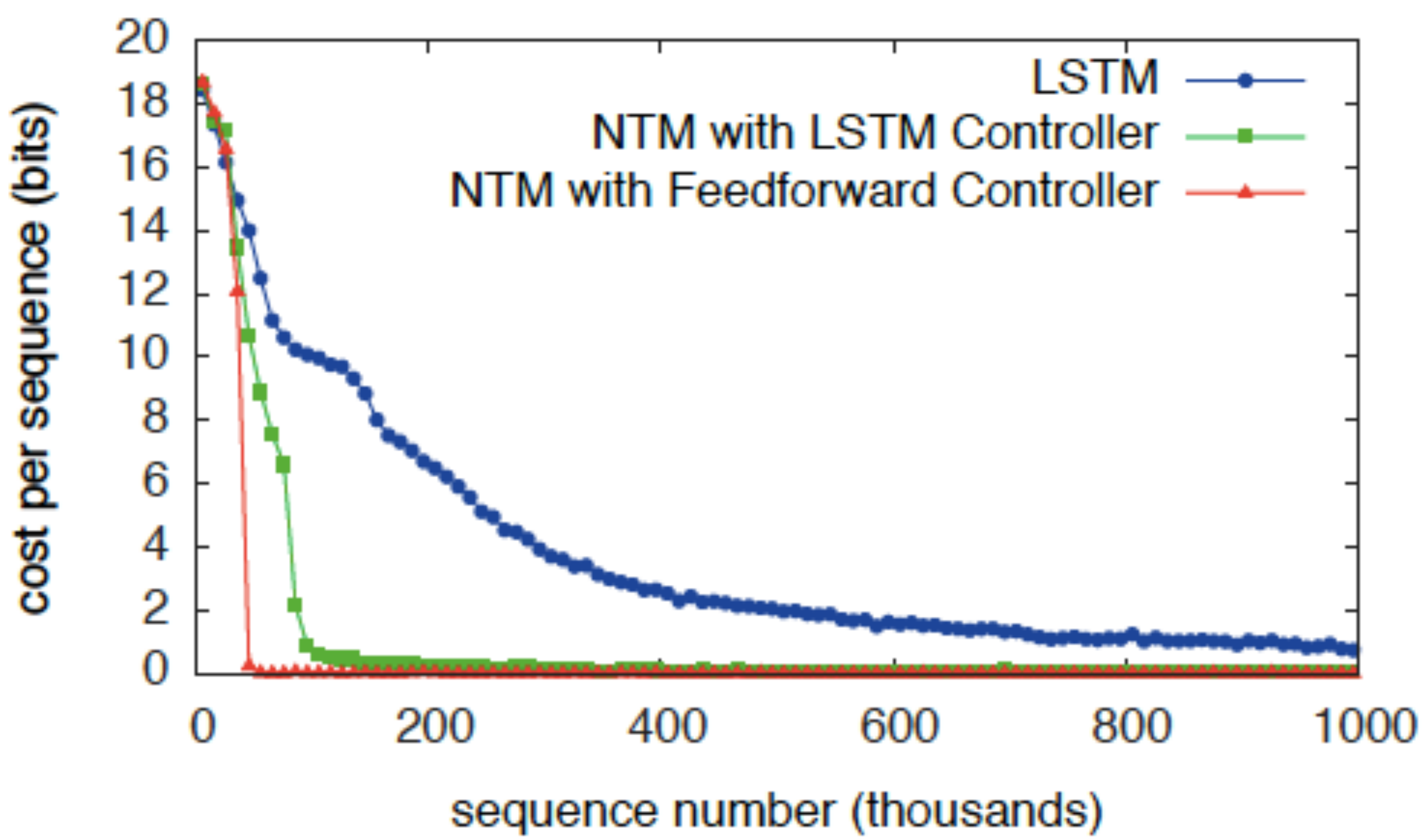


Figure 10: Associative Recall Learning Curves for NTM and LSTM.

Exp 4: Associative Recall

input: start codon + sequence of items (random binary) + stop codon

testing: random element of sequence, **output:** next element

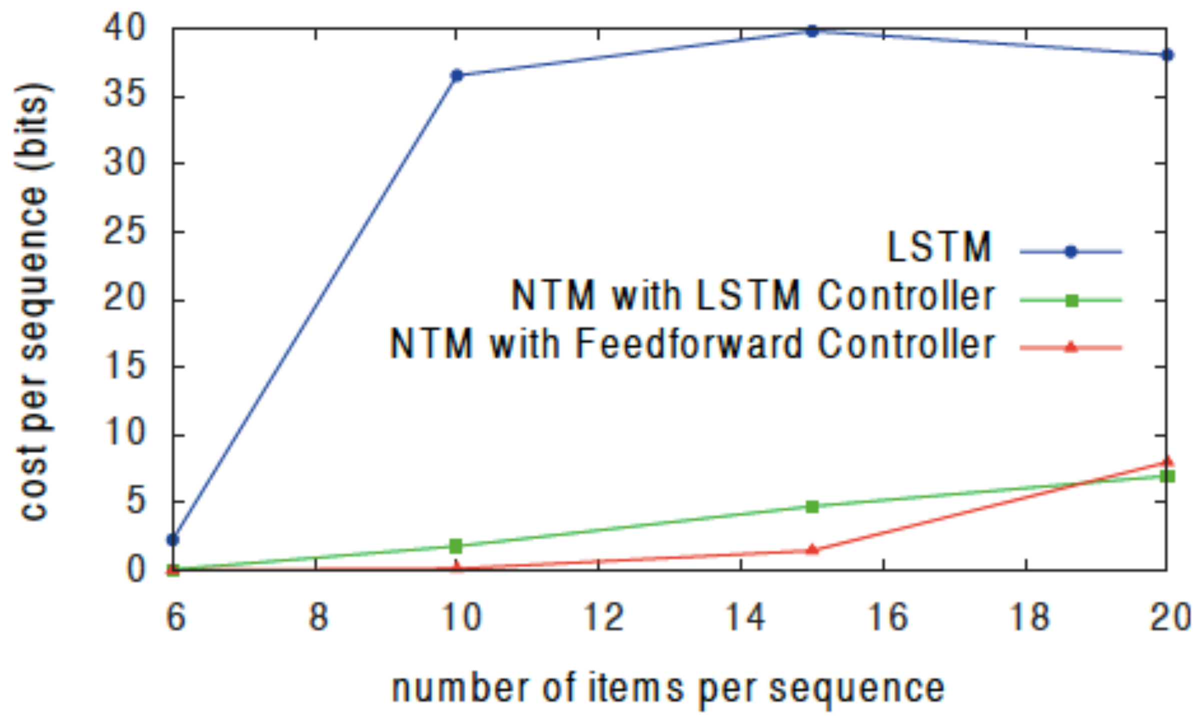
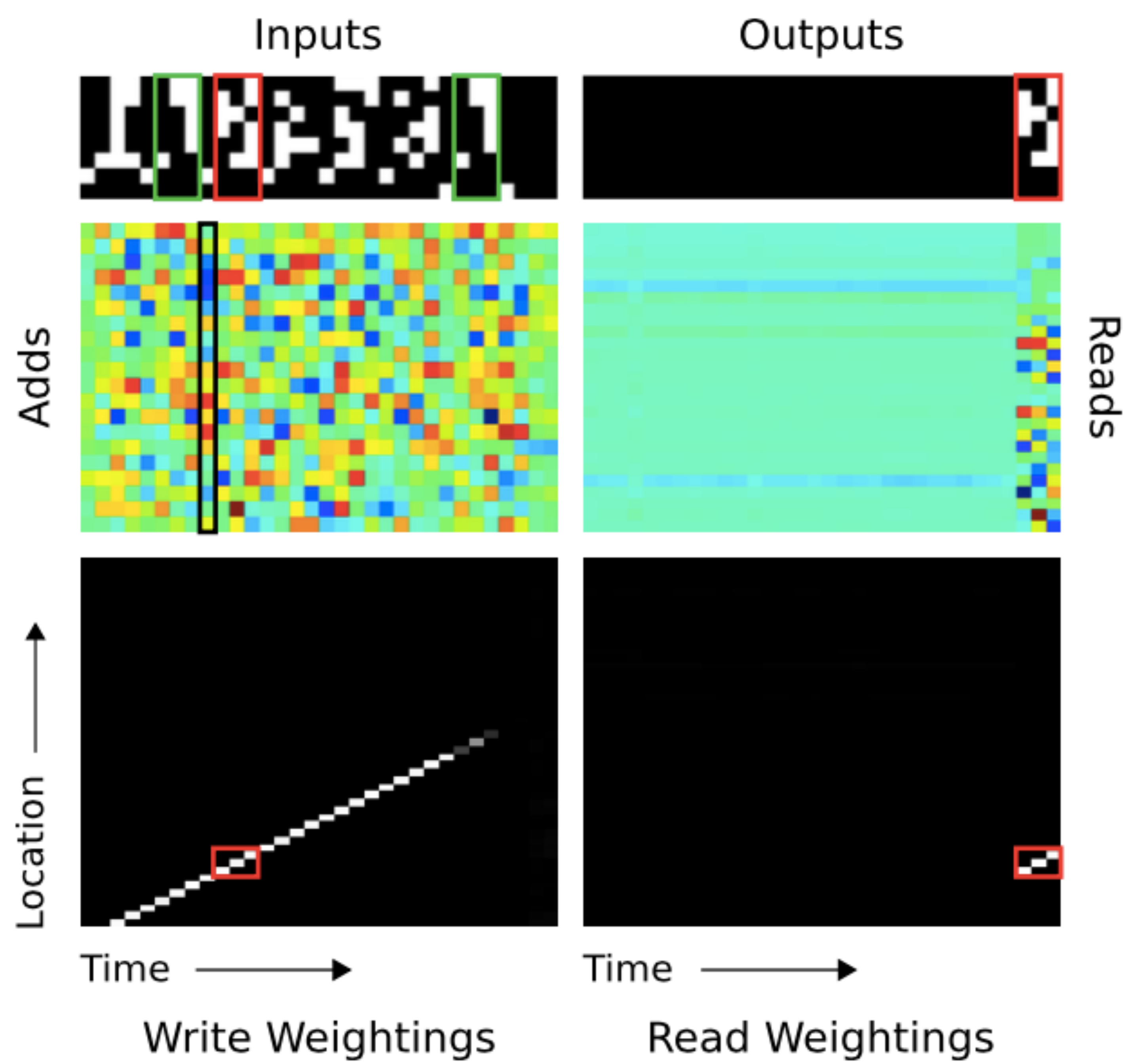


Figure 11: Generalisation Performance on Associative Recall for Longer Item Sequences. The NTM with either a feedforward or LSTM controller generalises to much longer sequences of items than the LSTM alone. In particular, the NTM with a feedforward controller is nearly perfect for item sequences of twice the length of sequences in its training set.

Exp 4: Associative Recall

Memory usage patterns

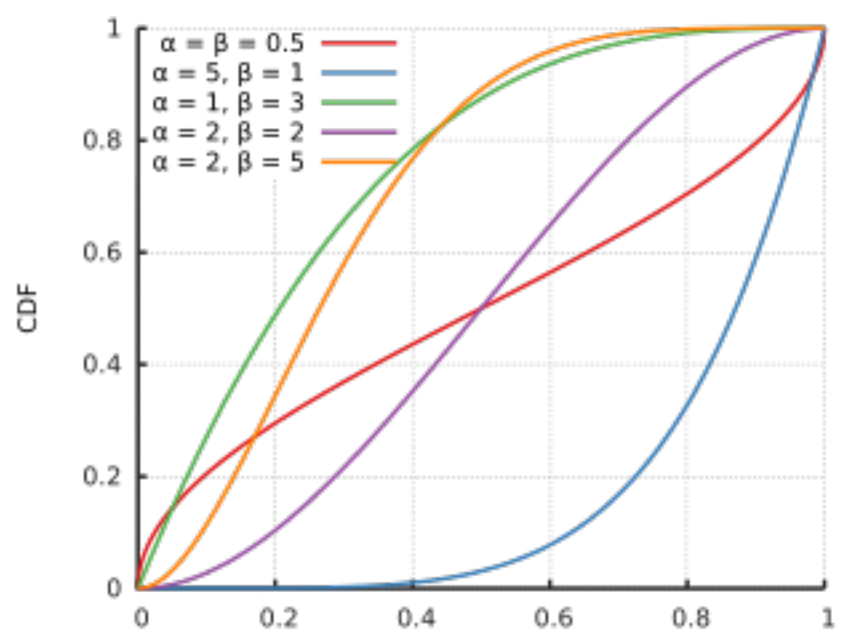


Exp 5: Dynamic N-grams

input: stream generated by binary n-gram model

output: next element of sequence

specifically: 6-gram transition matrices (2x5) generated from beta(1/2, 1/2) distribution of 2x5s



Beta distribution chosen because it describes the statistics of a nice special case of how sequences of temporally-interrelated data often arise (e.g. the Chinese restaurant process, dirichlet distribution)

Exp 5: Dynamic N-grams

input: stream generated by binary n-gram model

output: next element of sequence

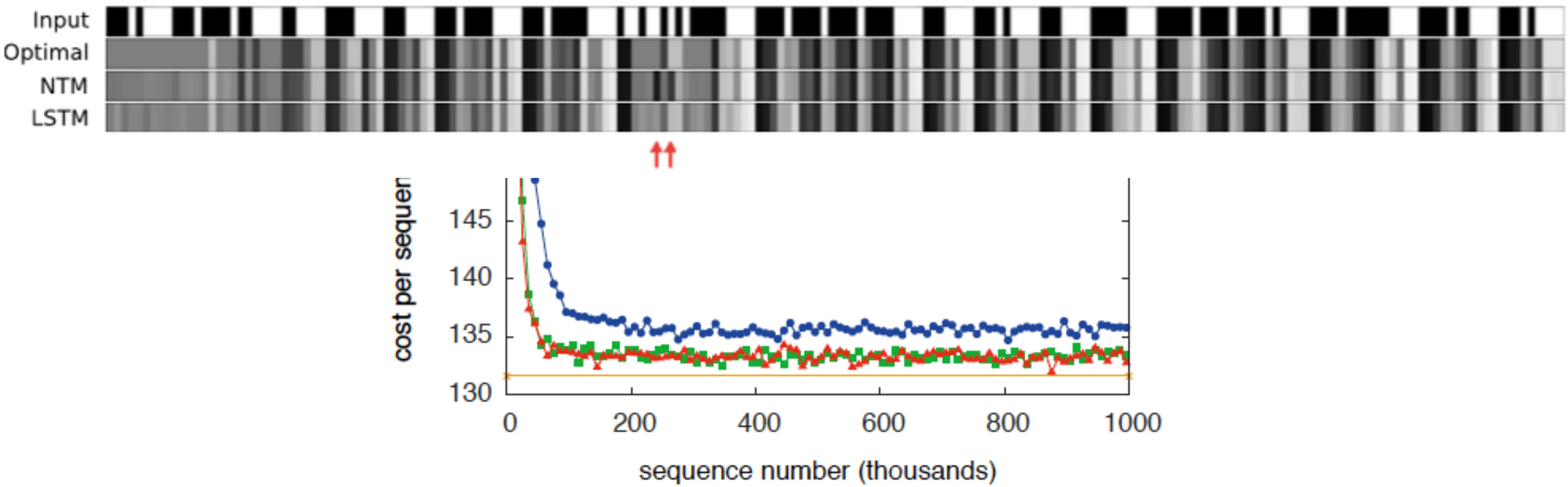
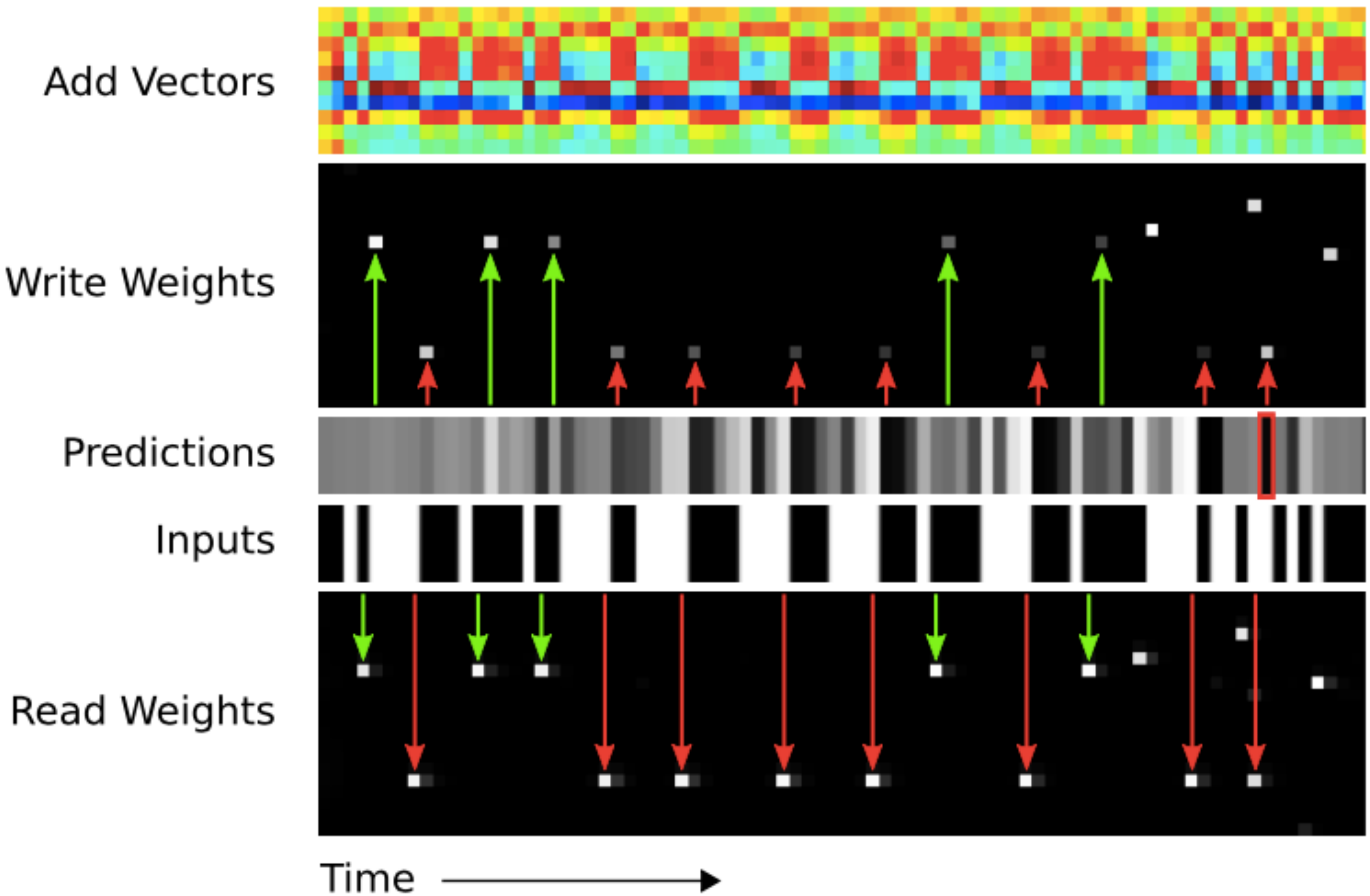


Figure 13: Dynamic N-Gram Learning Curves.

Exp 5: Dynamic N-grams

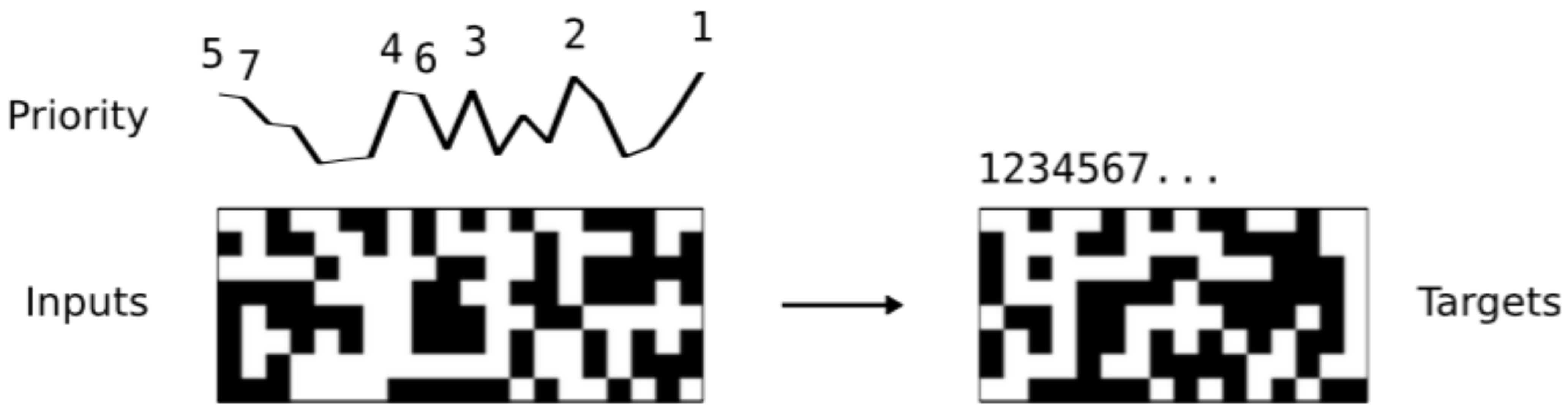
Memory usage patterns



Exp 6: Sorting

input: sequence of binary vectors + scalar priority list

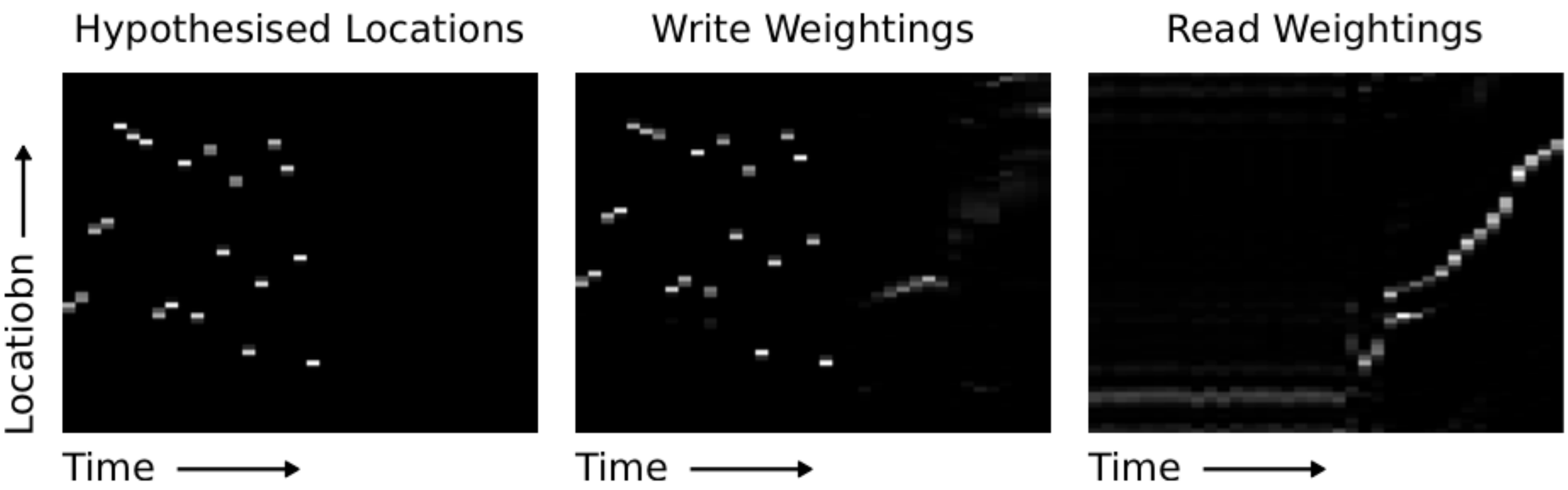
output: vectors sorted by priority



Exp 6: Sorting

input: sequence of binary vectors + scalar priority list

output: vectors sorted by priority



NTM experiment details:

Task	#Heads	Controller Size	Memory Size	Learning Rate	#Parameters
Copy	1	100	128×20	10^{-4}	$17,162$
Repeat Copy	1	100	128×20	10^{-4}	$16,712$
Associative	4	256	128×20	10^{-4}	$146,845$
N-Grams	1	100	128×20	3×10^{-5}	$14,656$
Priority Sort	8	512	128×20	3×10^{-5}	$508,305$

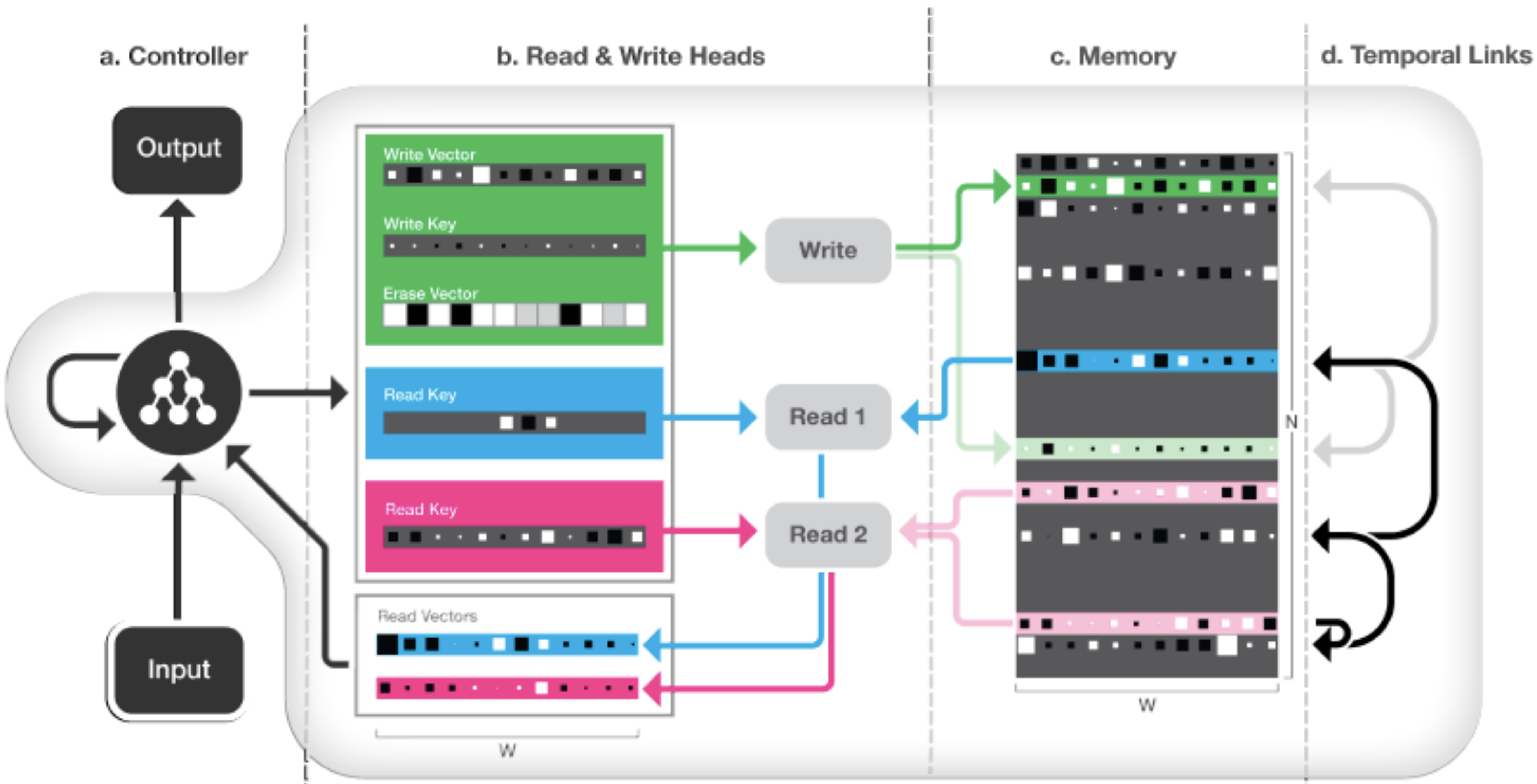
Table 1: NTM with Feedforward Controller Experimental Settings

Models: Deep Networks with External Memory Stores

More recent (somewhat more powerful) version:

Symbolic Reasoning with Differentiable Neural Computers

Alex Graves*, Greg Wayne*, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gomez, Edward Grefenstette, Tiago Ramalho, John Agapiou, Adrià Puigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen

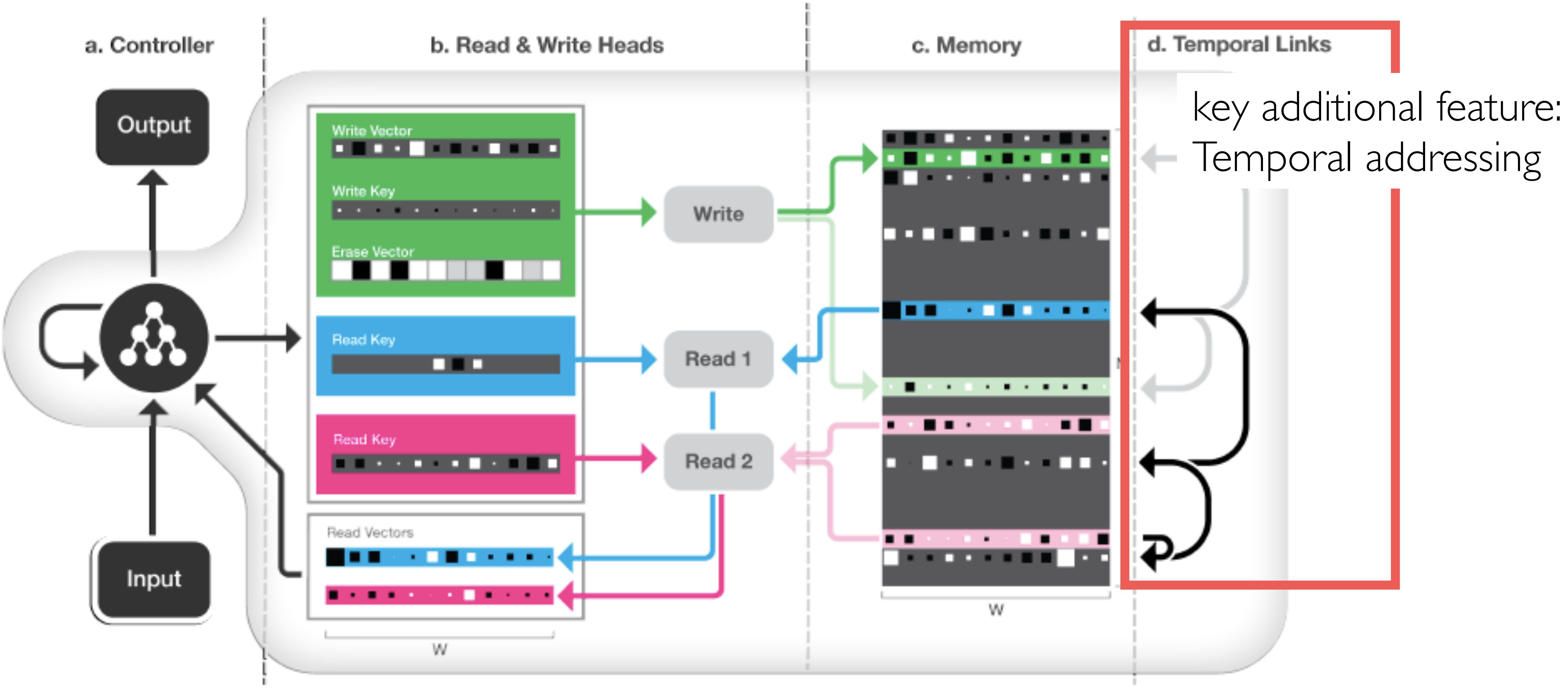


Models: Deep Networks with External Memory Stores

More recent (somewhat more powerful) version:

Symbolic Reasoning with Differentiable Neural Computers

Alex Graves*, Greg Wayne*, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gomez, Edward Grefenstette, Tiago Ramalho, John Agapiou, Adrià Puigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen



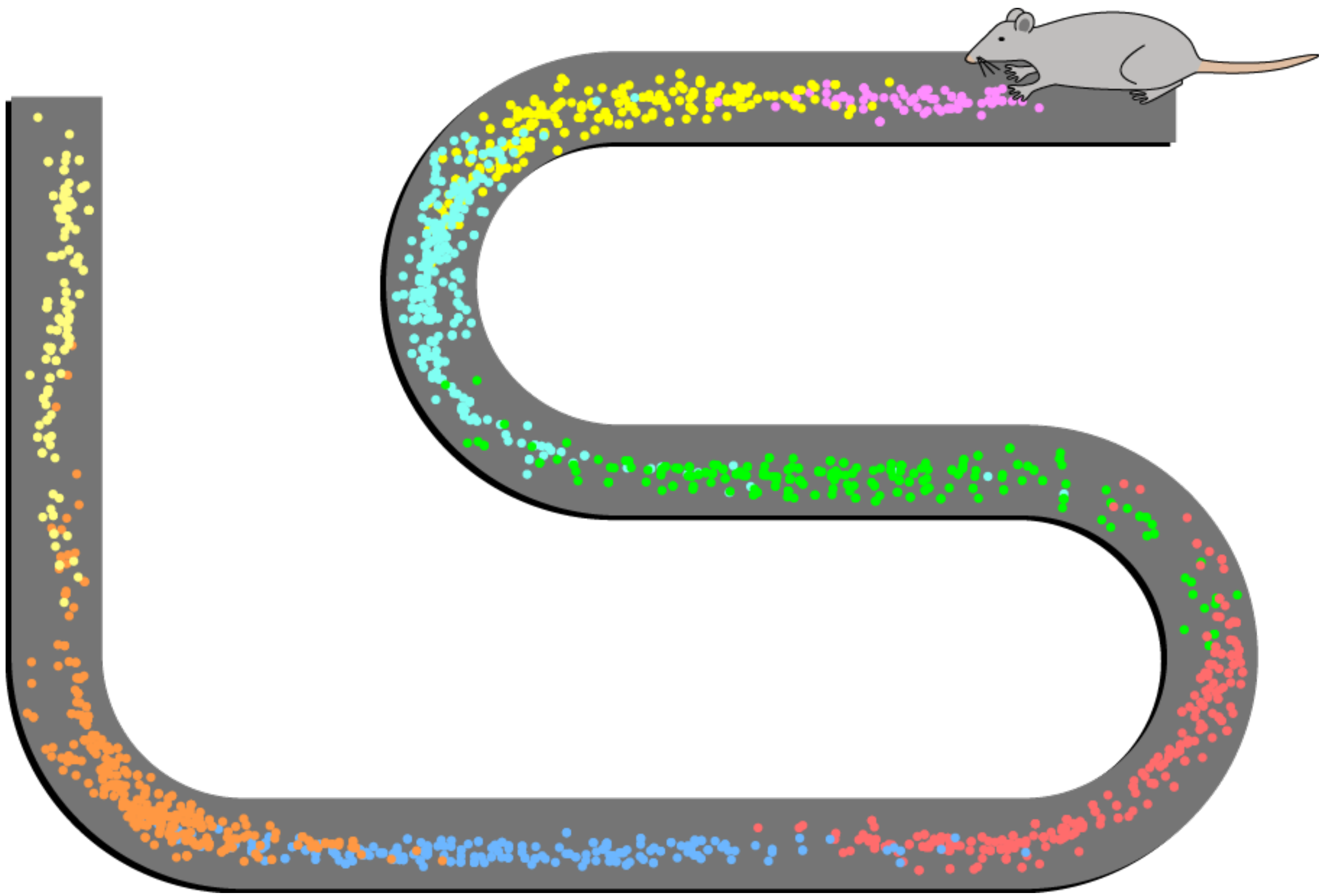
Functions of the hippocampus

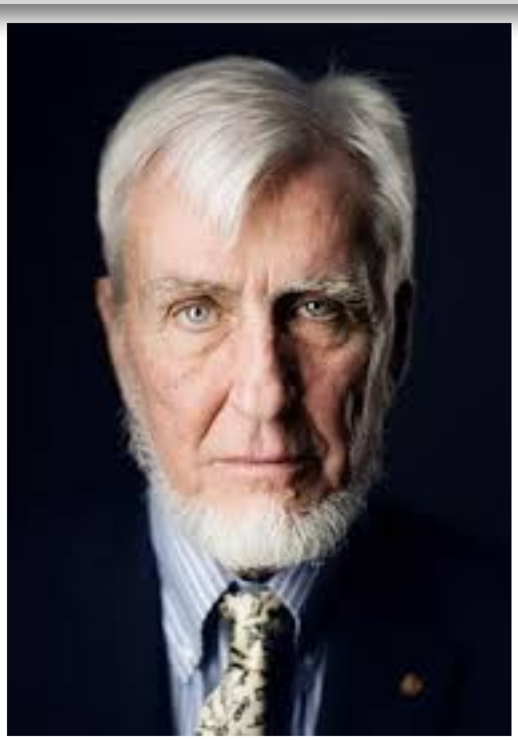
1. Behavioral inhibition theory ("slam on the breaks")

2. Memory (Milner & Scoville from HM)

3. Spatial cognition

Hippocampus as a cognitive map

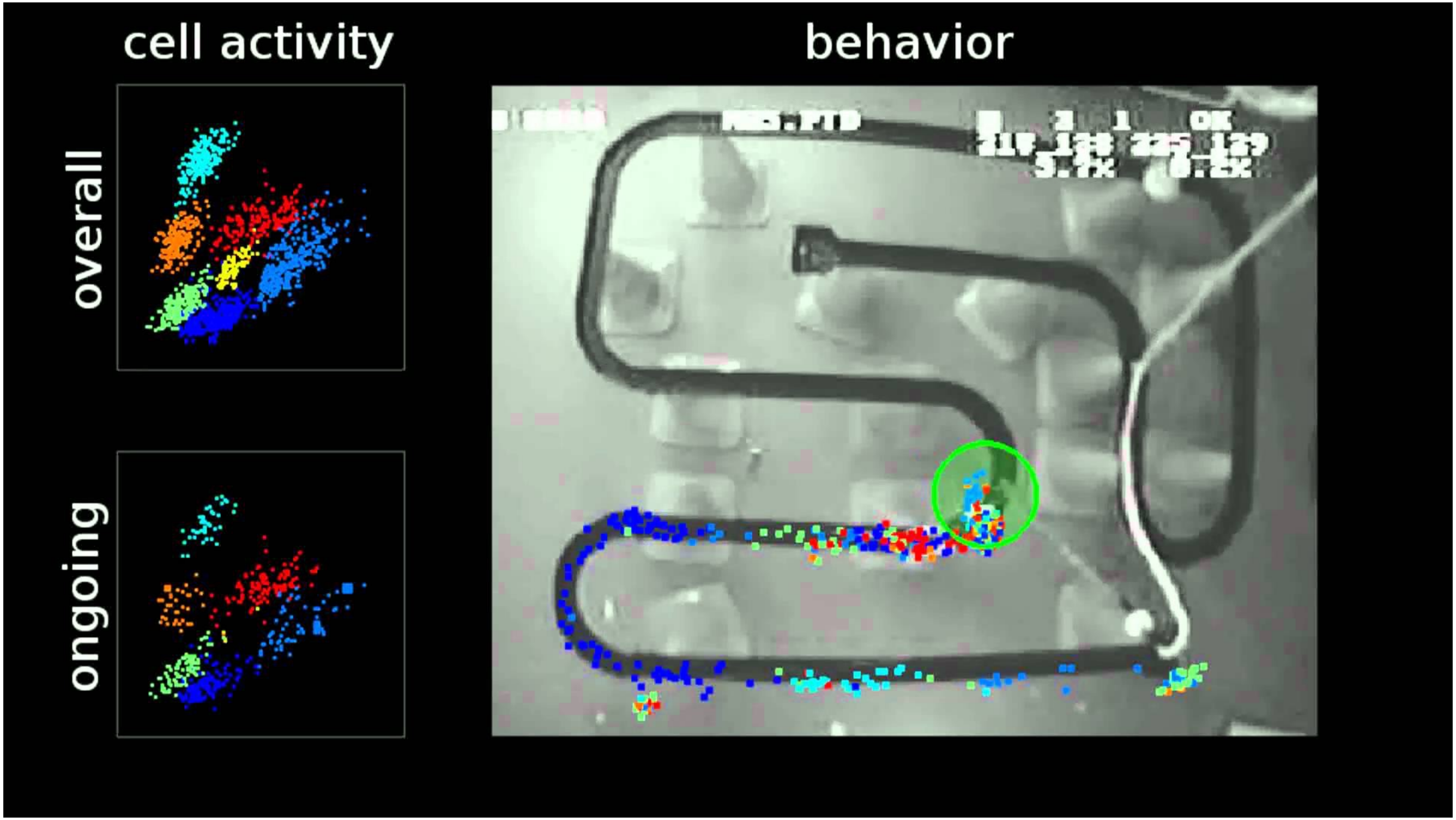




John O'keefe

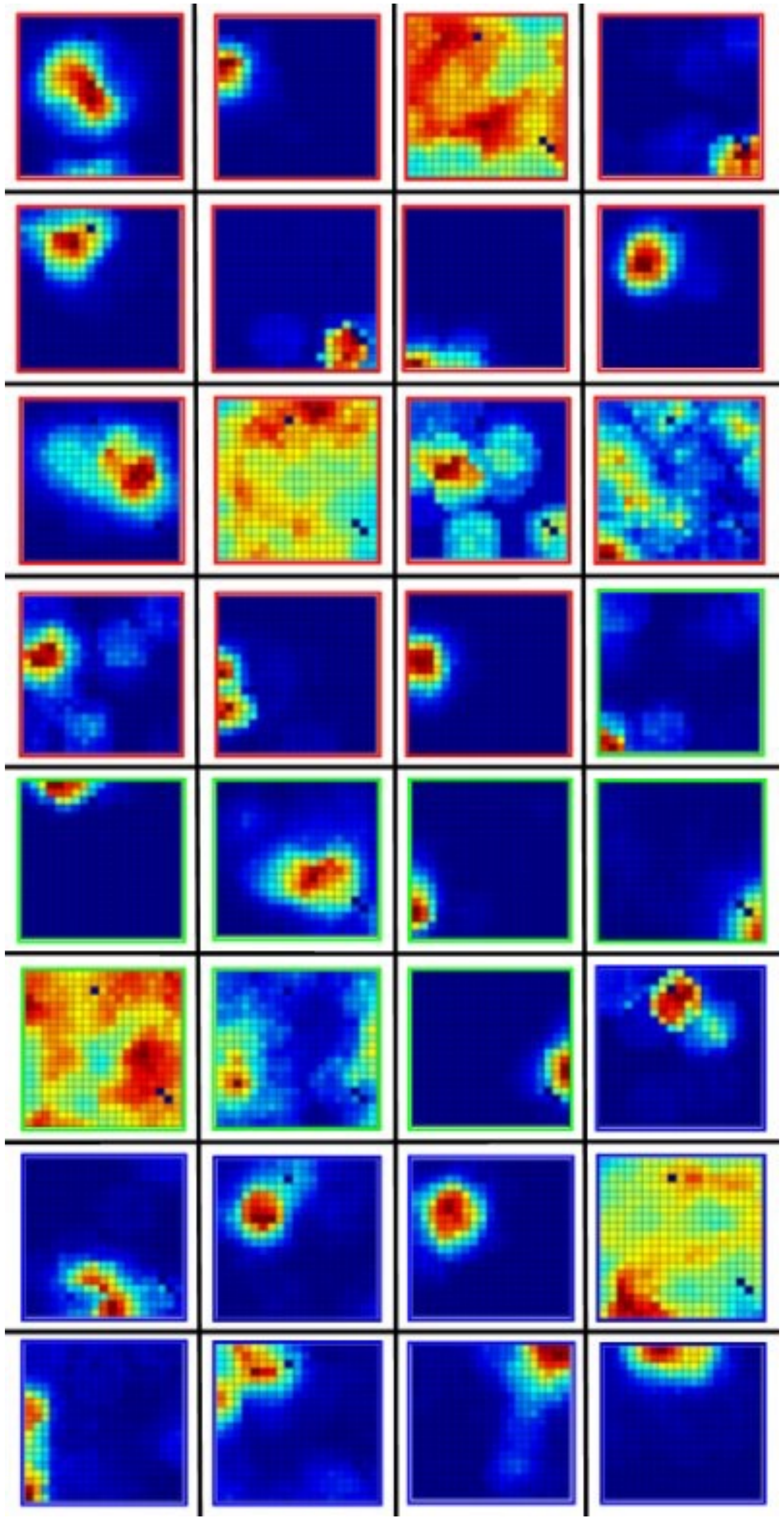
Hippocampus as a cognitive map

https://www.youtube.com/watch?v=lfNVv0A8QvI



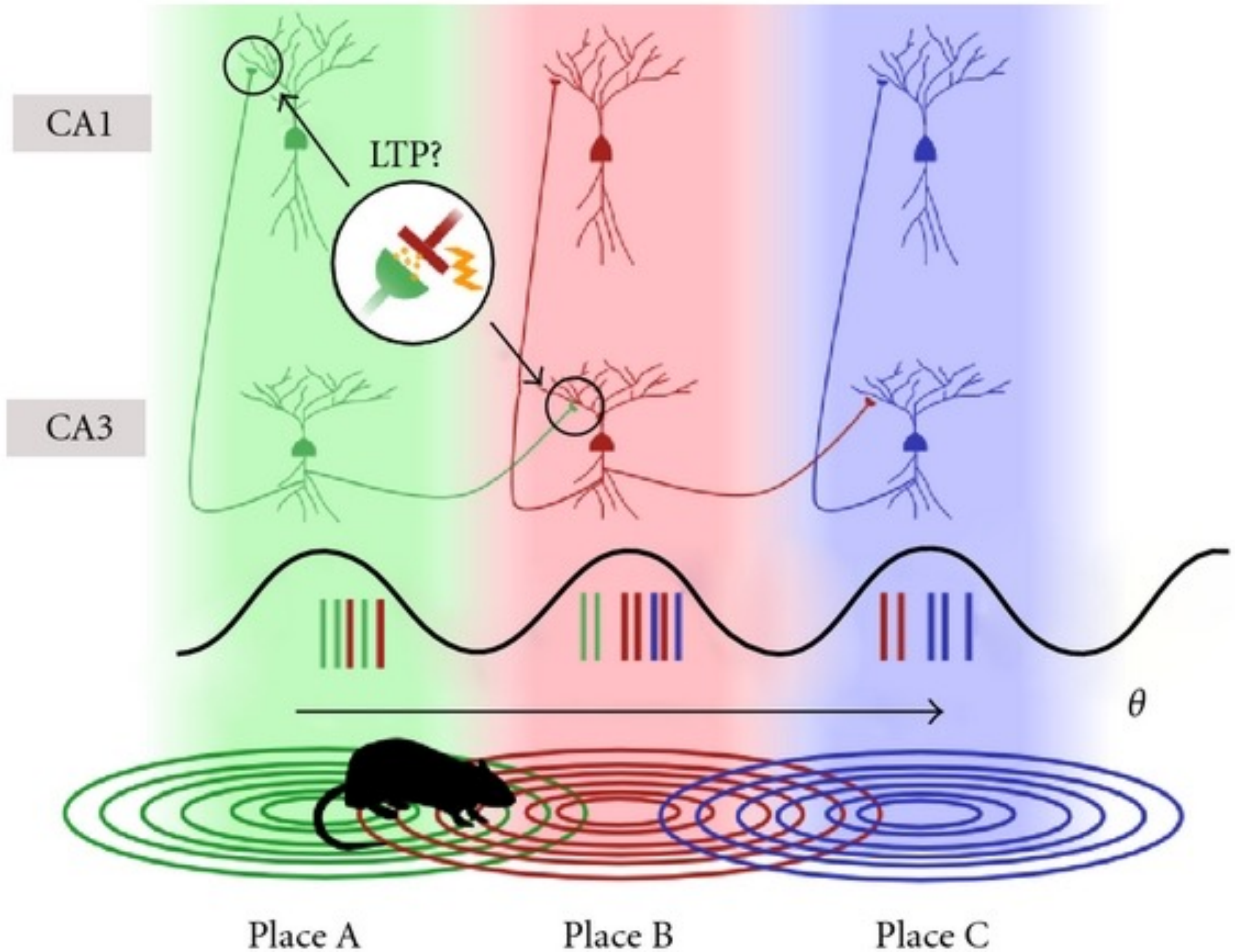
Hippocampus as a cognitive map

Hippocampal Place Cells *(O'keefe & Nadel, 1970s)*



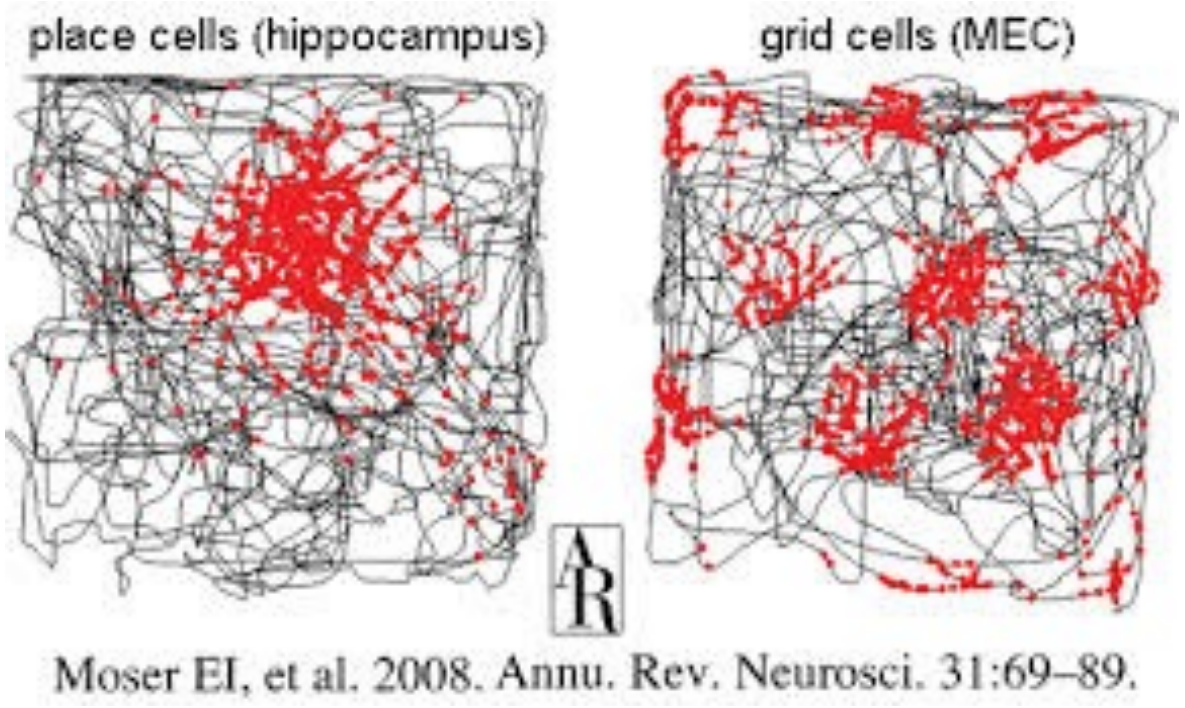
Hippocampal Place Cells

Get formed quickly and just a quickly remapped



Entorhinal Grid Cells

Entorhinal Grid Cells (Mozers, 2005)



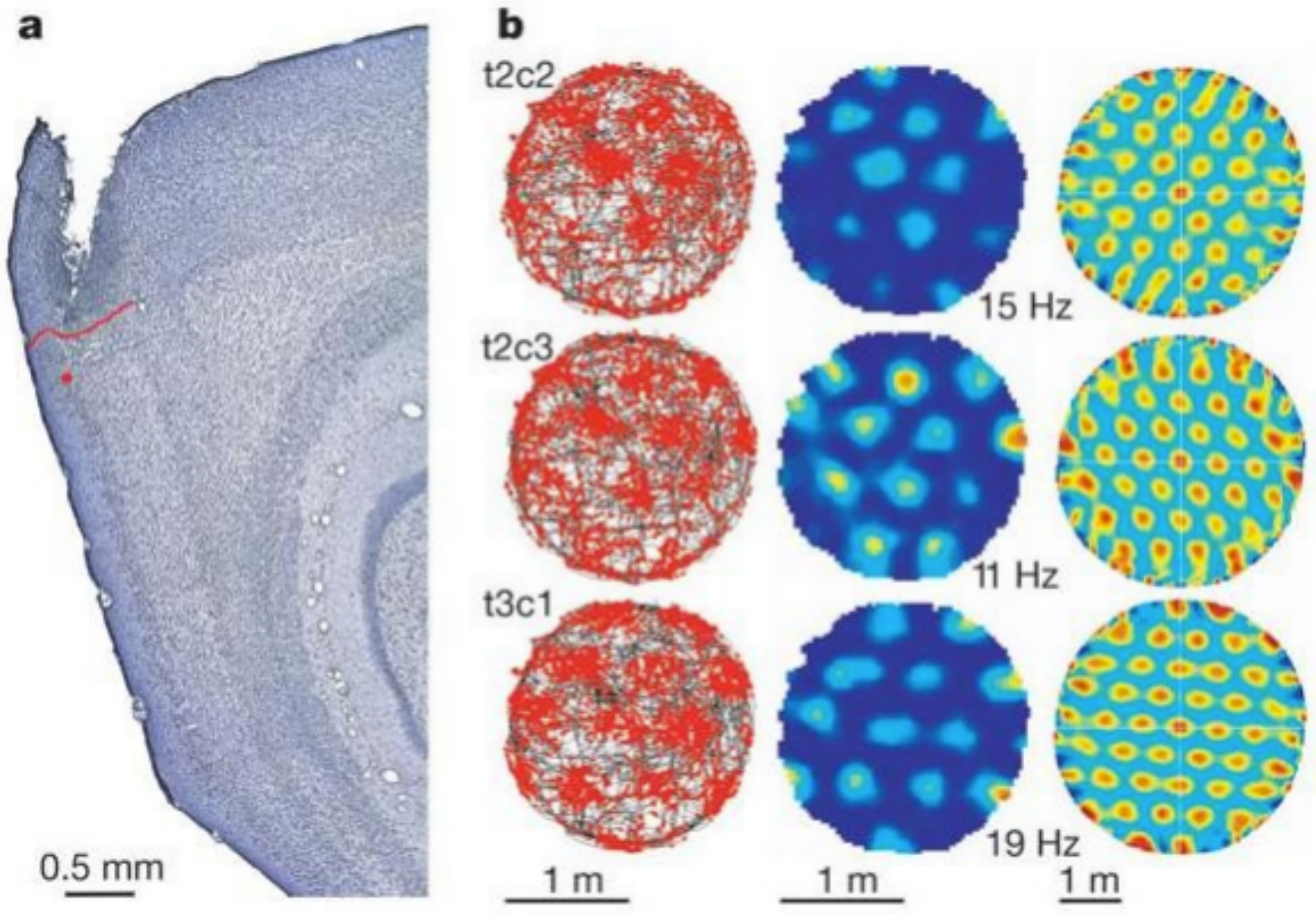
May-Britt Moser

Edvard Moser

Entorhinal Grid Cells

Entorhinal Grid Cells (Mozers, 2005)

grids are hexagonal and independent of arena size



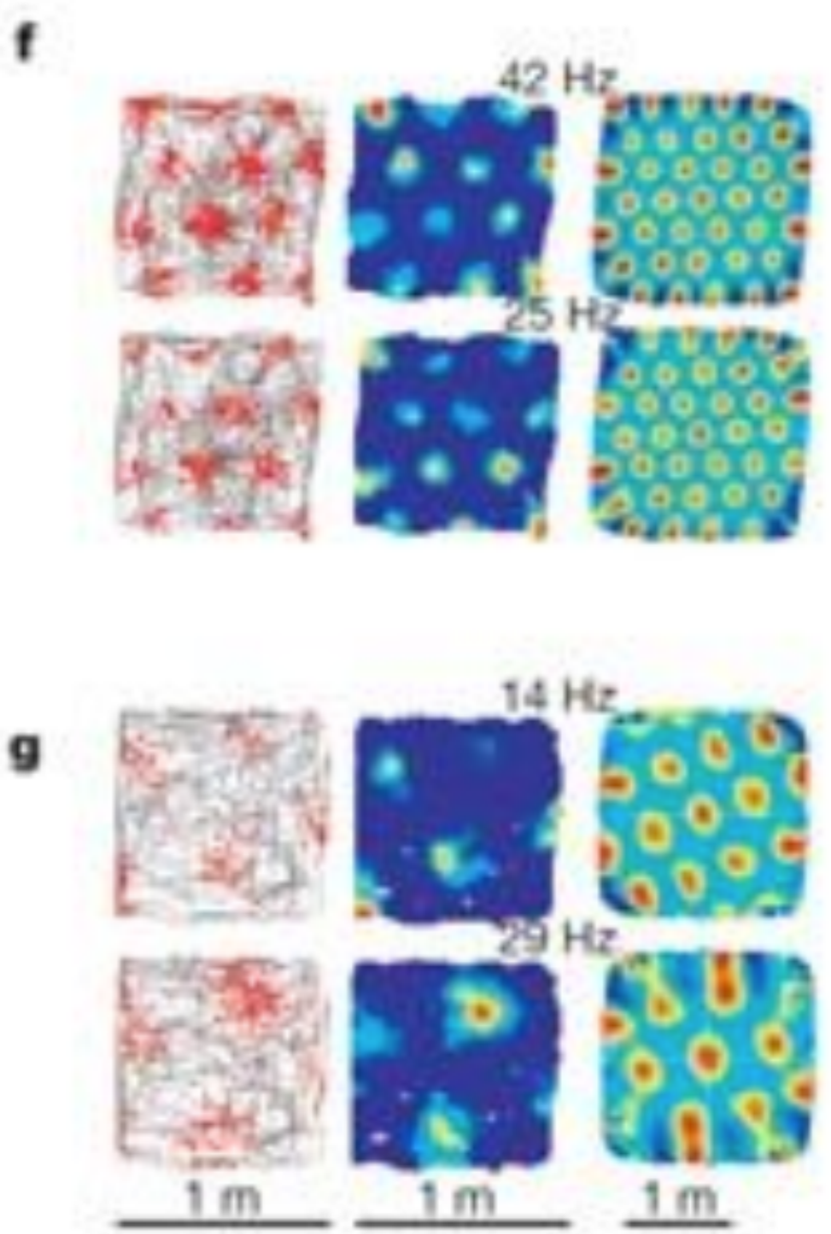
Maintain alignment with visual landmarks.

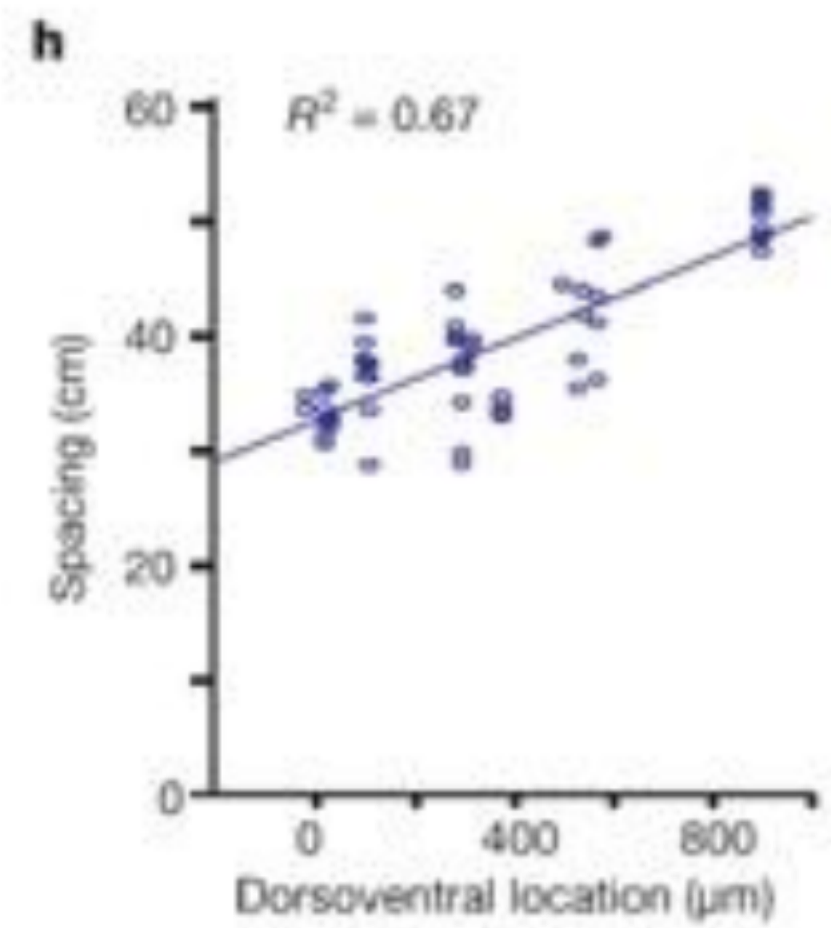
May-Britt Moser

Edvard Moser

Entorhinal Grid Cells

Entorhinal Grid Cells (Mozers, 2005)





Hafting et al , 2005

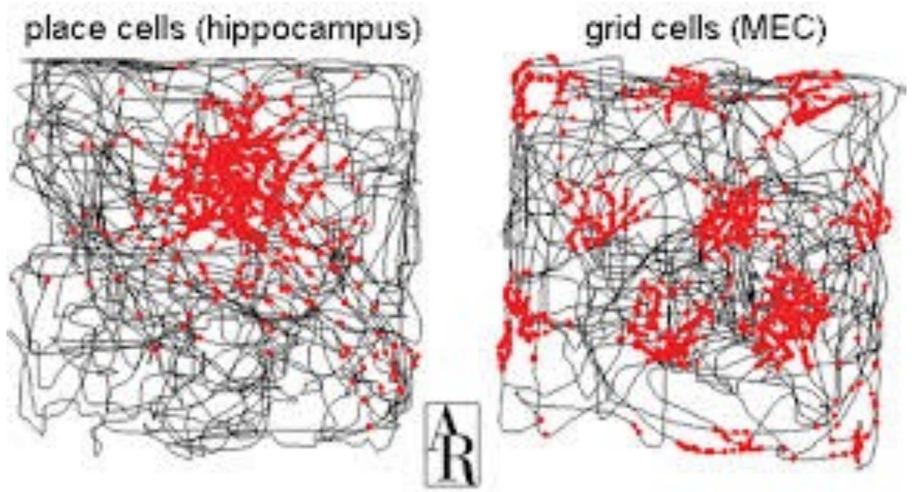
May-Britt Moser

Edvard Moser

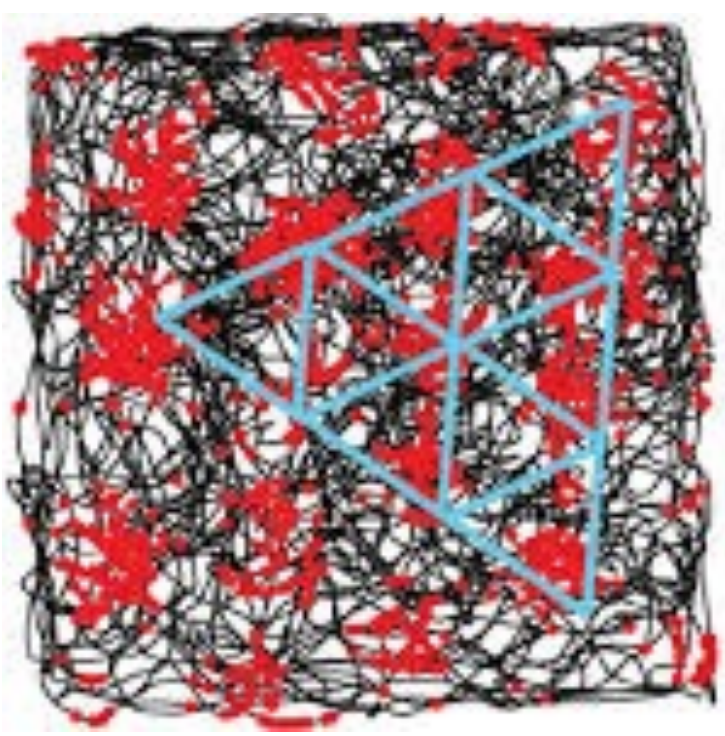
There are multiple maps of different grid spacings.

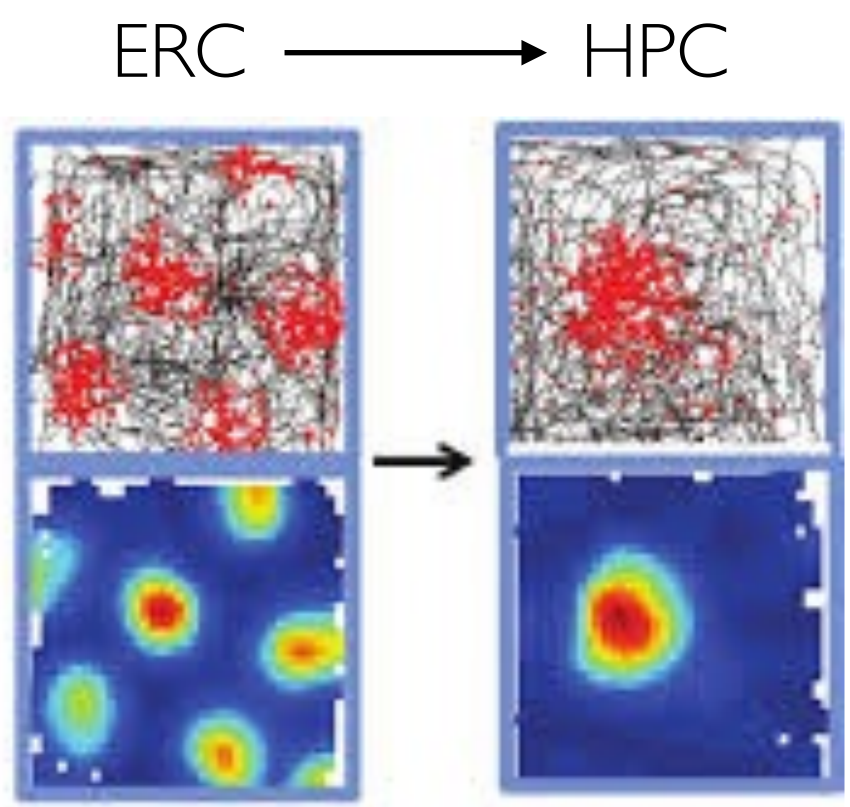
Entorhinal Grid Cells

Entorhinal Grid Cells (Mozers, 2005)



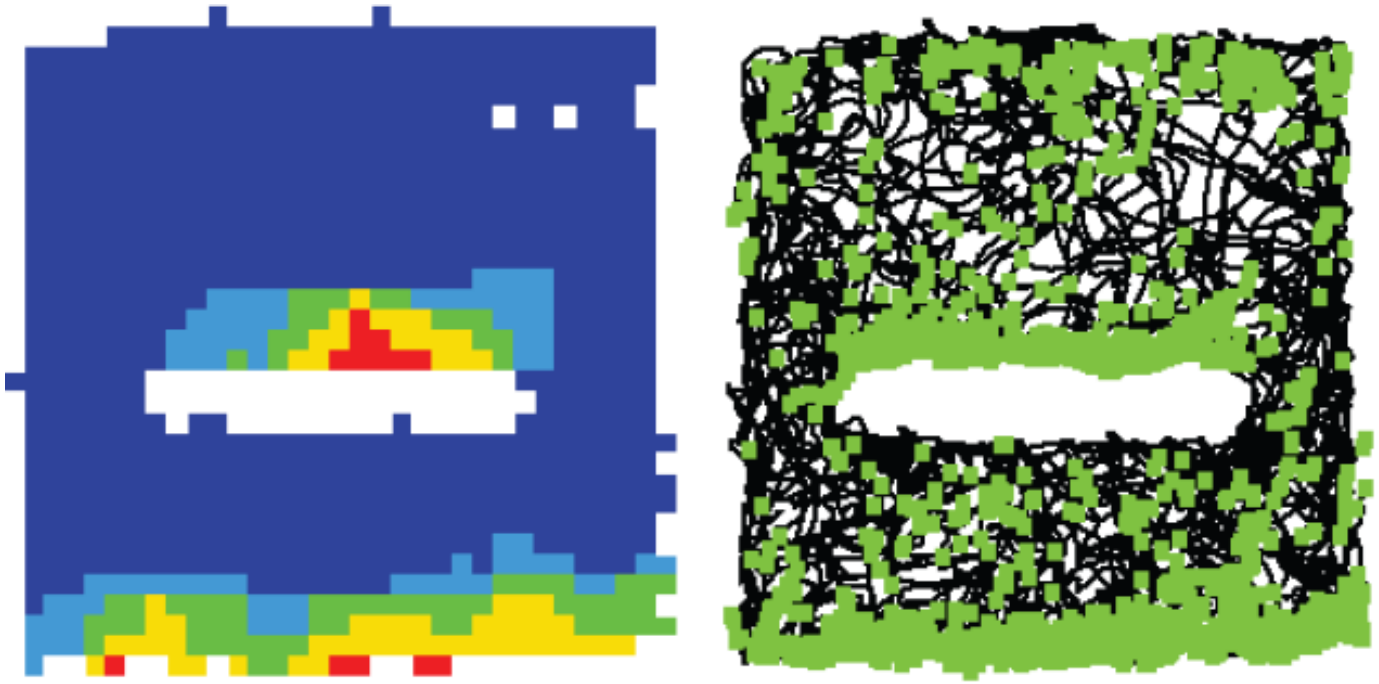
Moser EI, et al. 2008. Annu. Rev. Neurosci. 31:69–89.





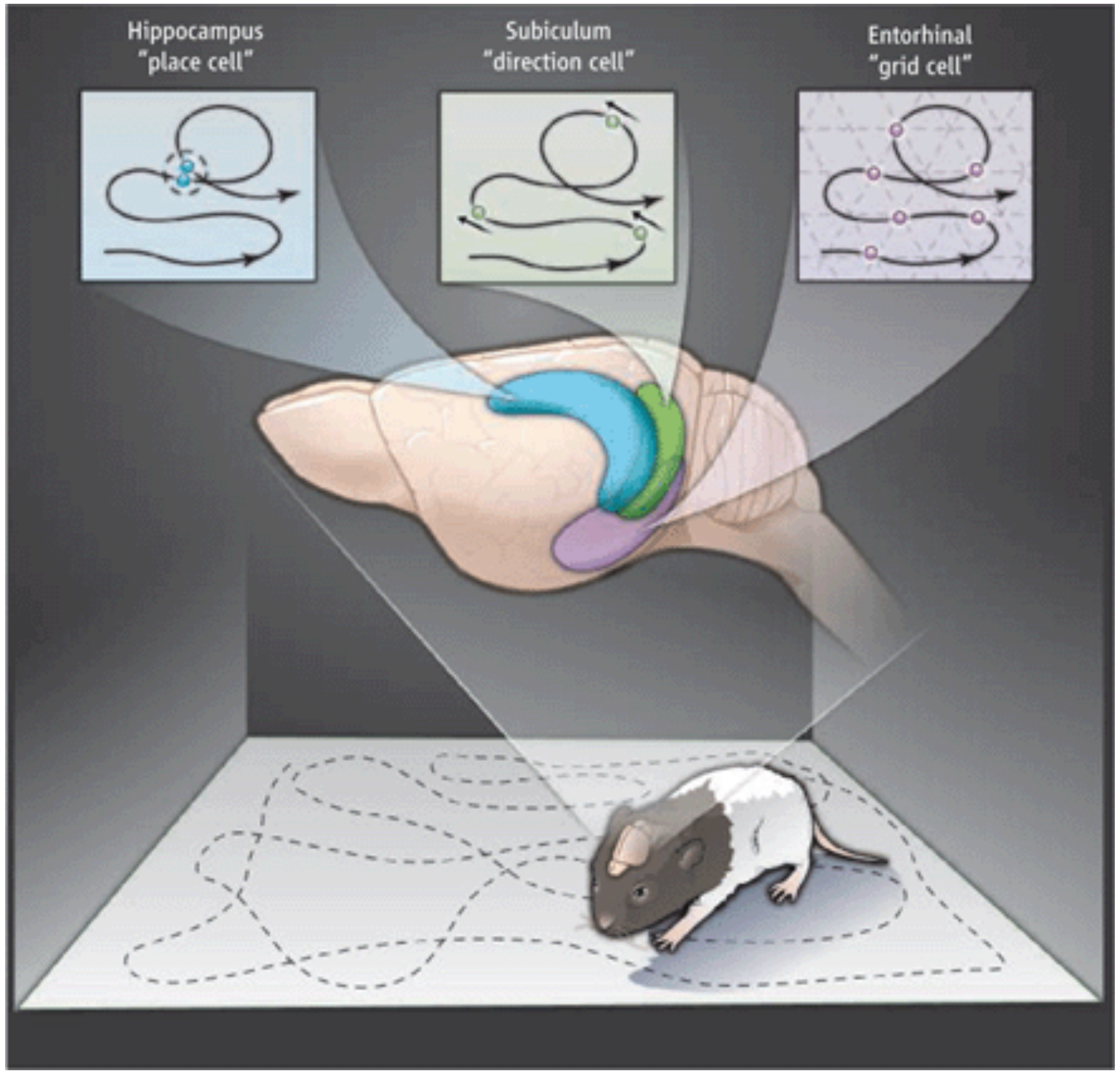
Hippocampus as a cognitive map

Boundary cells also found in subiculum (part of HPC) and ERC.

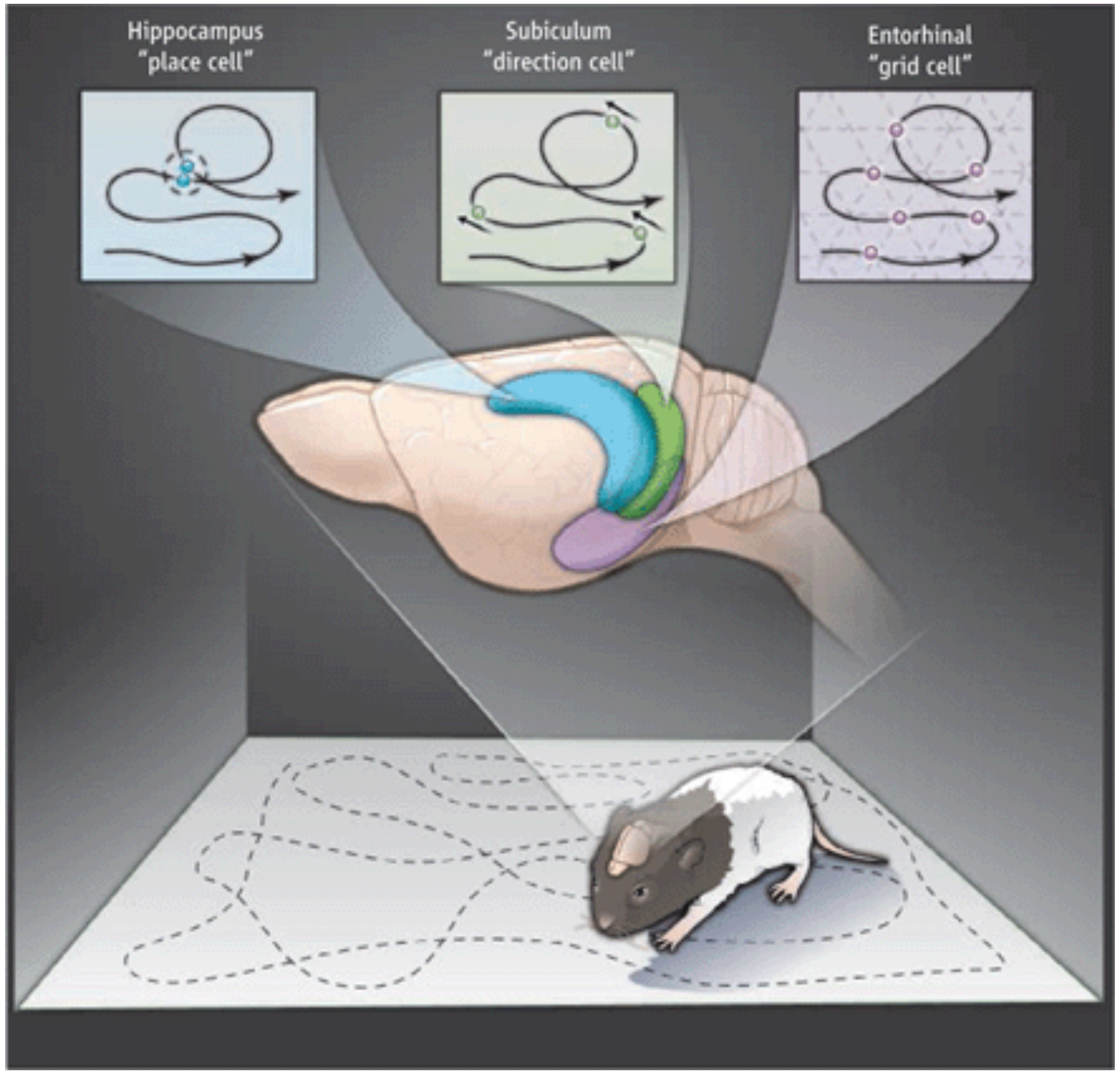


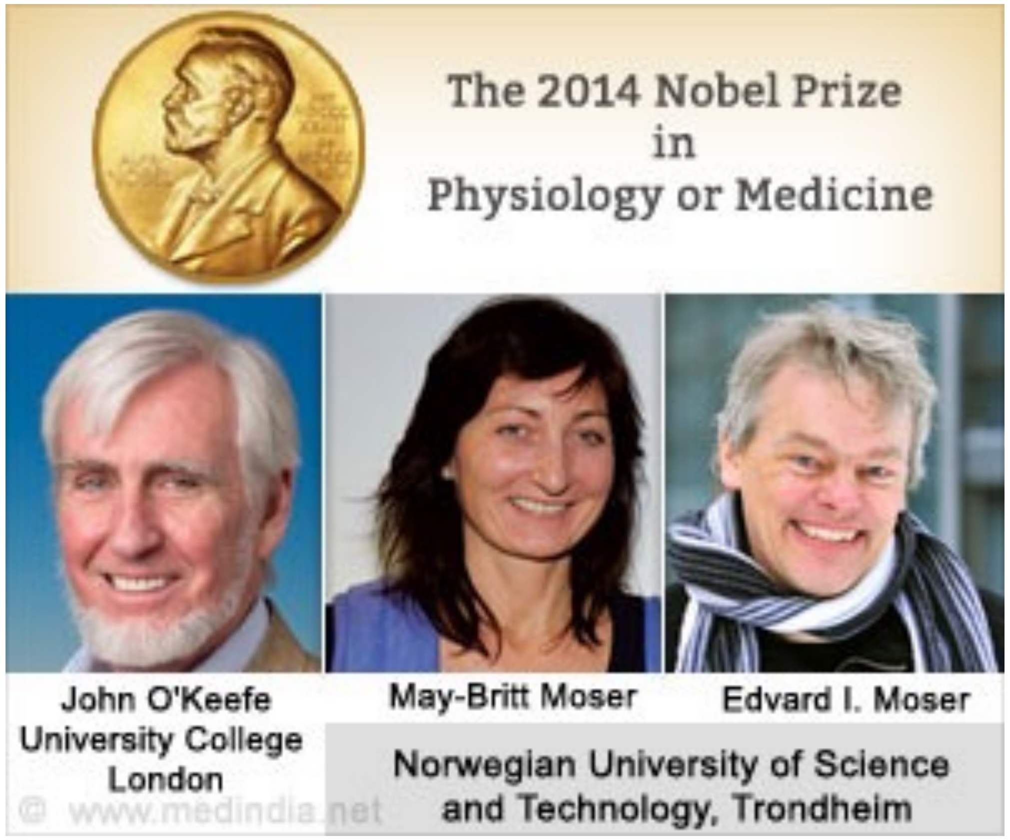
Firing of a boundary cell recorded in rat subiculum in 1 x 1 metre square-walled box with 50 cm-high walls. A 50 cm-long barrier inserted into box elicits second field along north side of barrier in addition to original field along south wall.

Hippocampus as a cognitive map

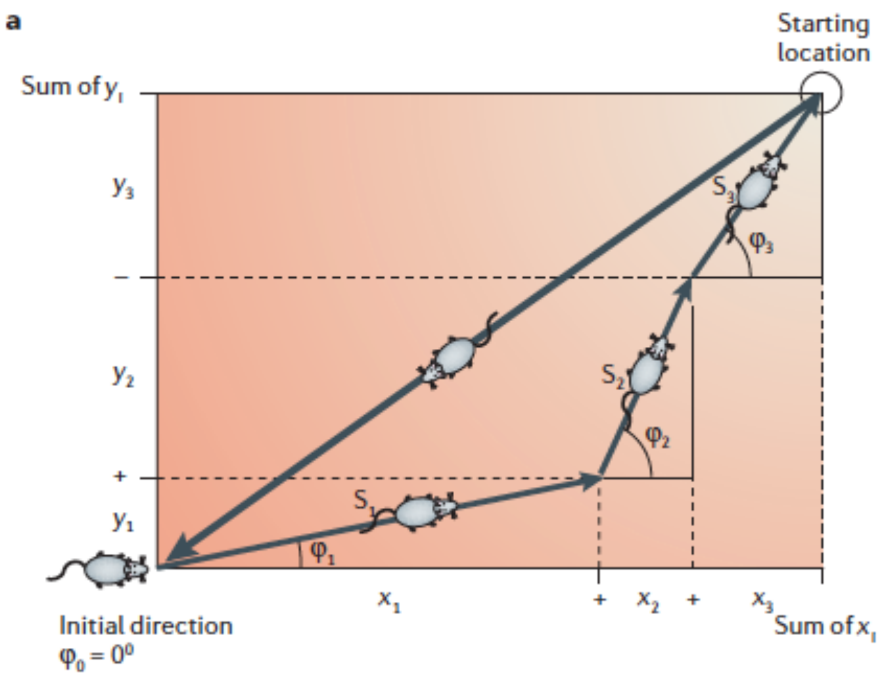


Hippocampus as a cognitive map

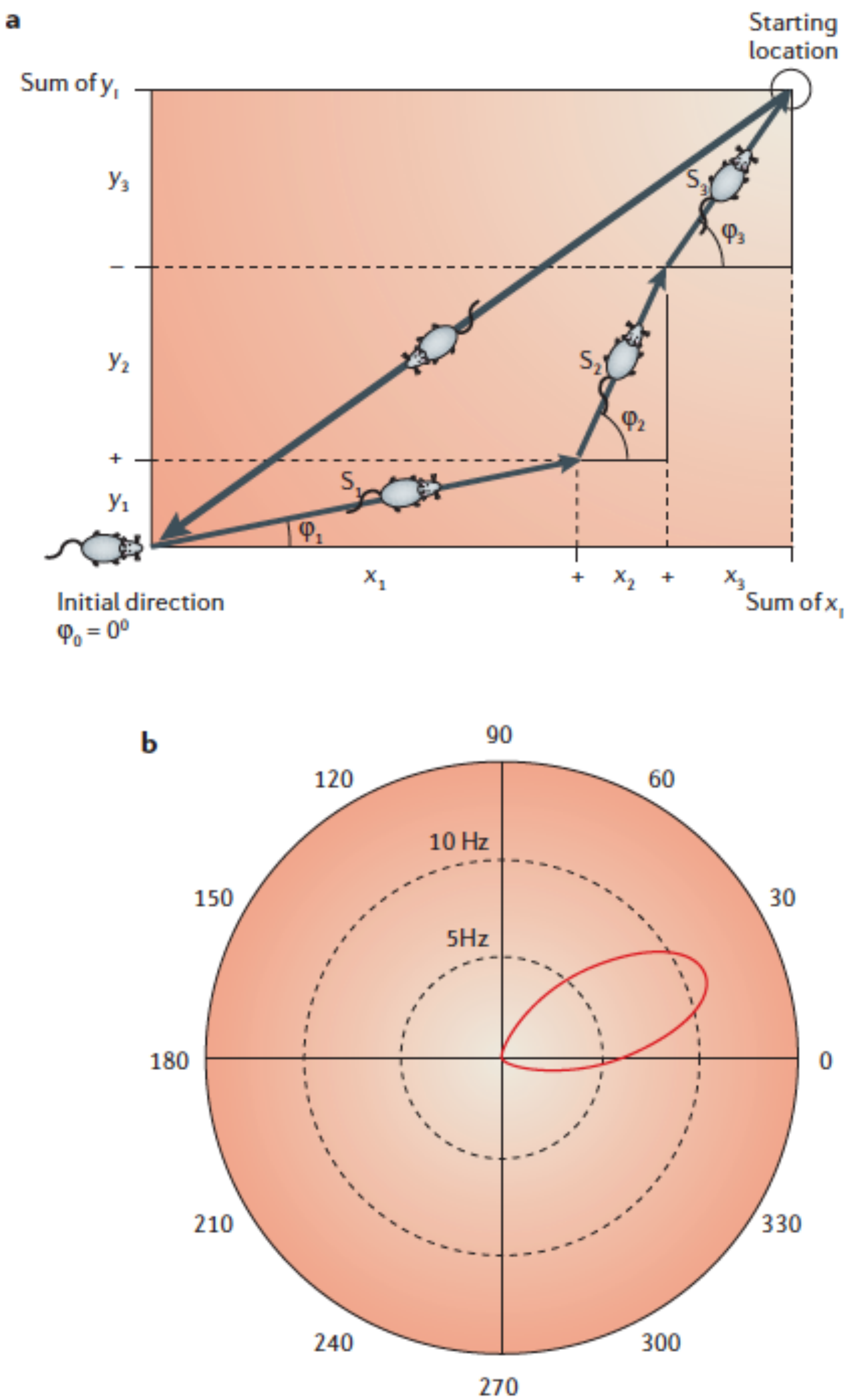




Models: Spatial Attractor of Path Integrator



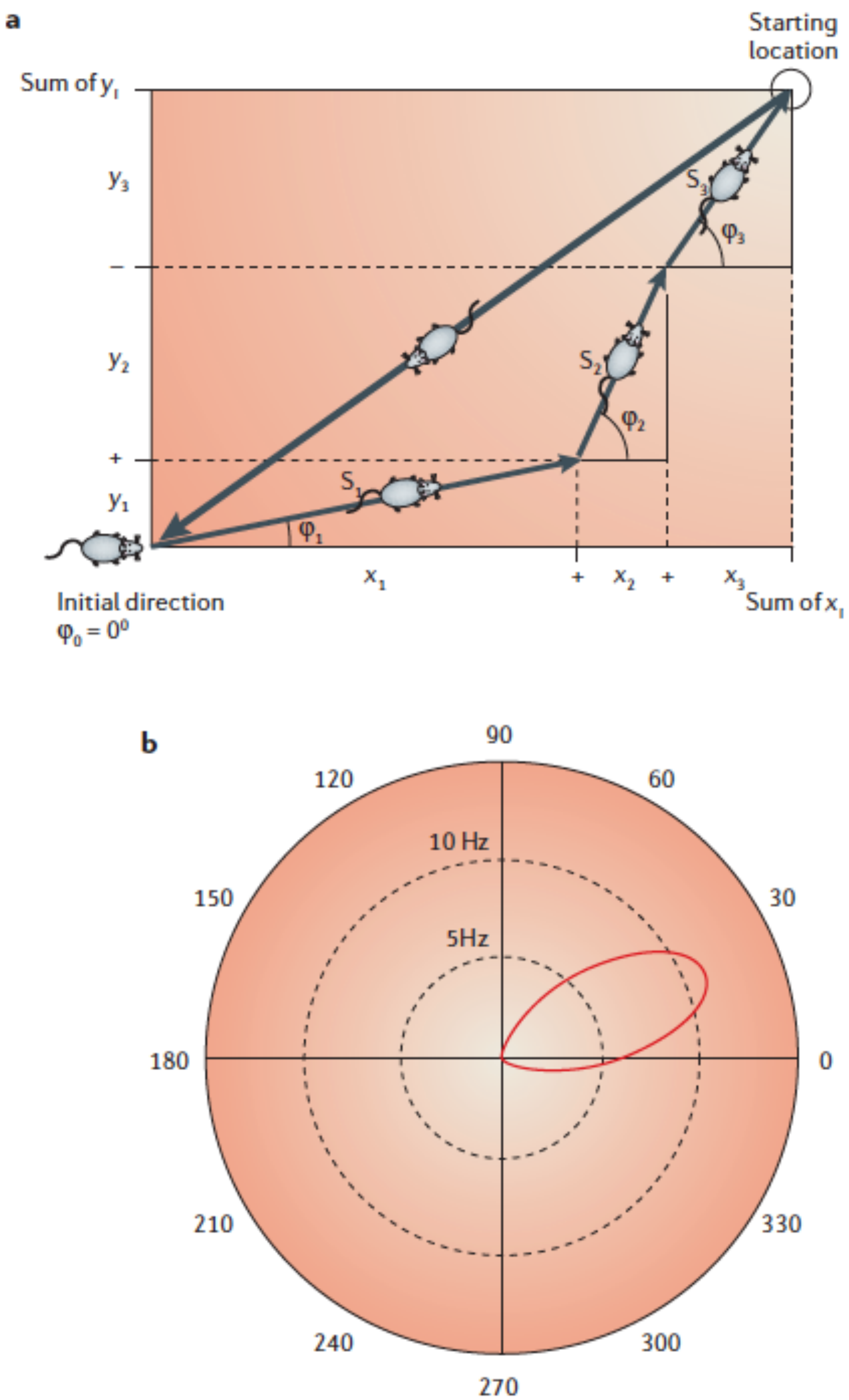
Models: Spatial Attractor of Path Integrator



Path integration and the neural basis of the ‘cognitive map’

Bruce L. McNaughton¶, Francesco P. Battaglia§, Ole Jensen||, Edvard I. Moser¶ and May-Britt Moser¶*

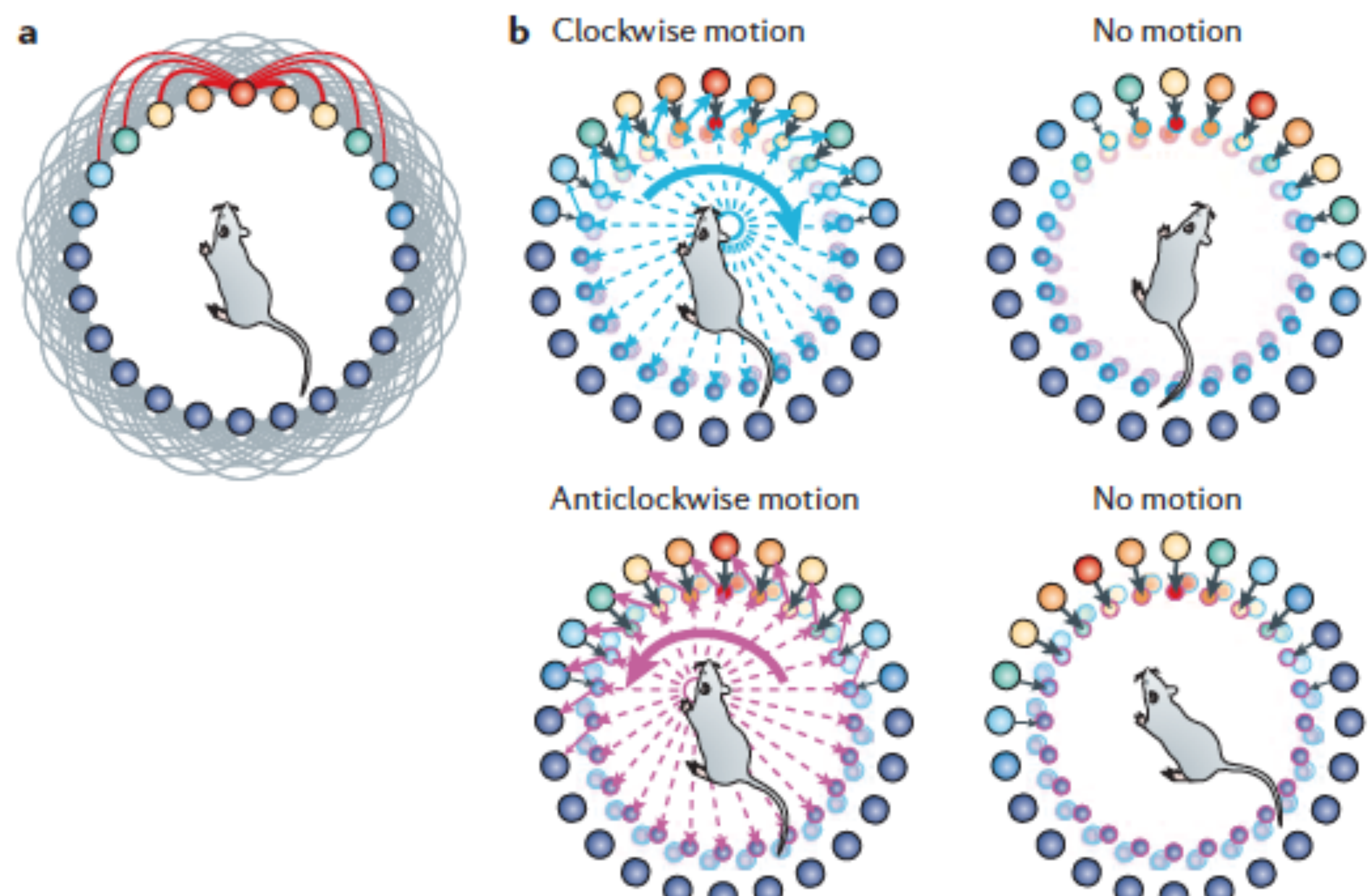
Models: Spatial Attractor of Path Integrator



Path integration and the neural basis of the 'cognitive map'

*Bruce L. McNaughton**[¶], *Francesco P. Battaglia*[§], *Ole Jensen*[||], *Edvard I. Moser*[¶] *and May-Britt Moser*[¶]

One-d attractor map



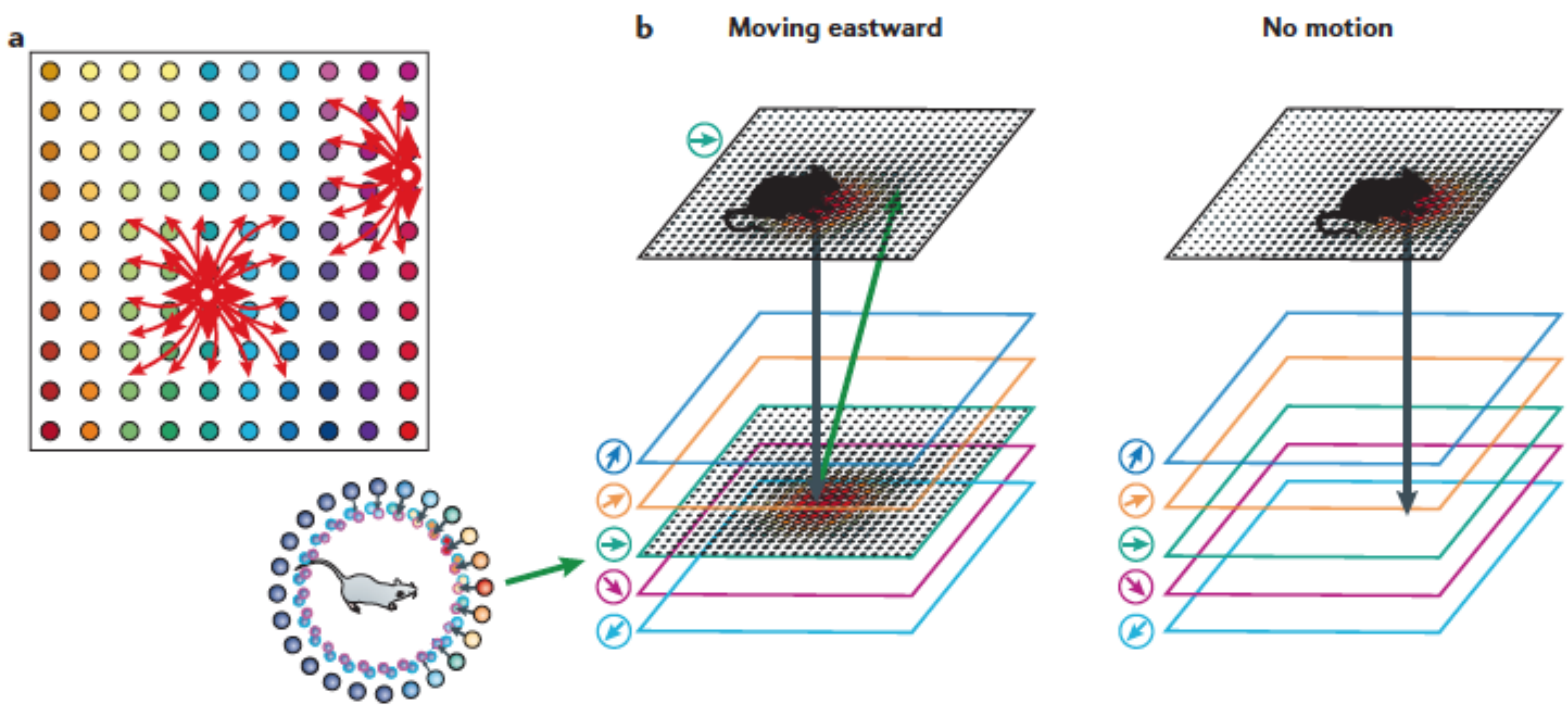
one ring of cells for clockwise,
one ring for counterclockwise

Models: Spatial Attractor of Path Integrator

Path integration and the neural basis of the 'cognitive map'

Bruce L. McNaughton[*¶], Francesco P. Battaglia[§], Ole Jensen[||], Edvard I. Moser[¶] and May-Britt Moser[¶]*

Two-D grid version



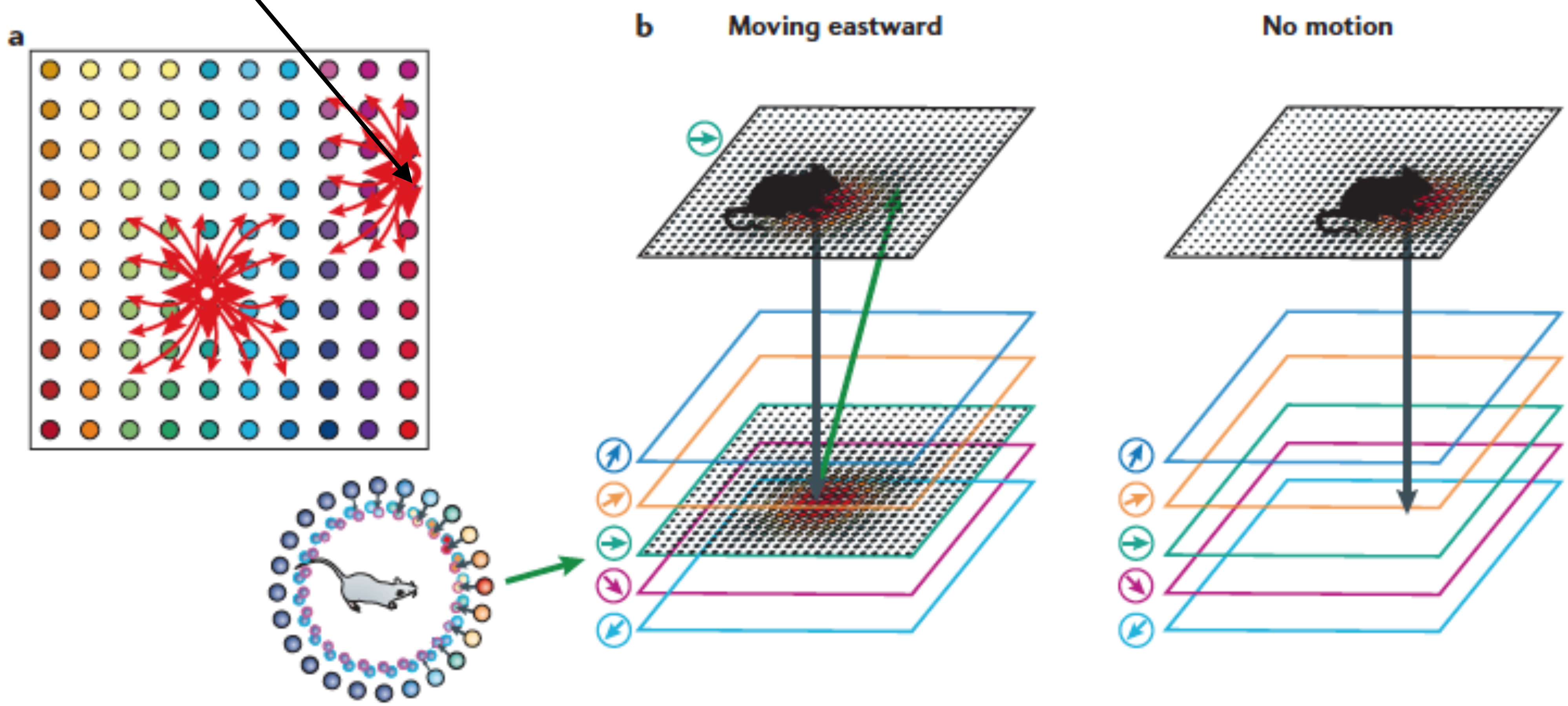
Models: Spatial Attractor of Path Integrator

Path integration and the neural basis of the 'cognitive map'

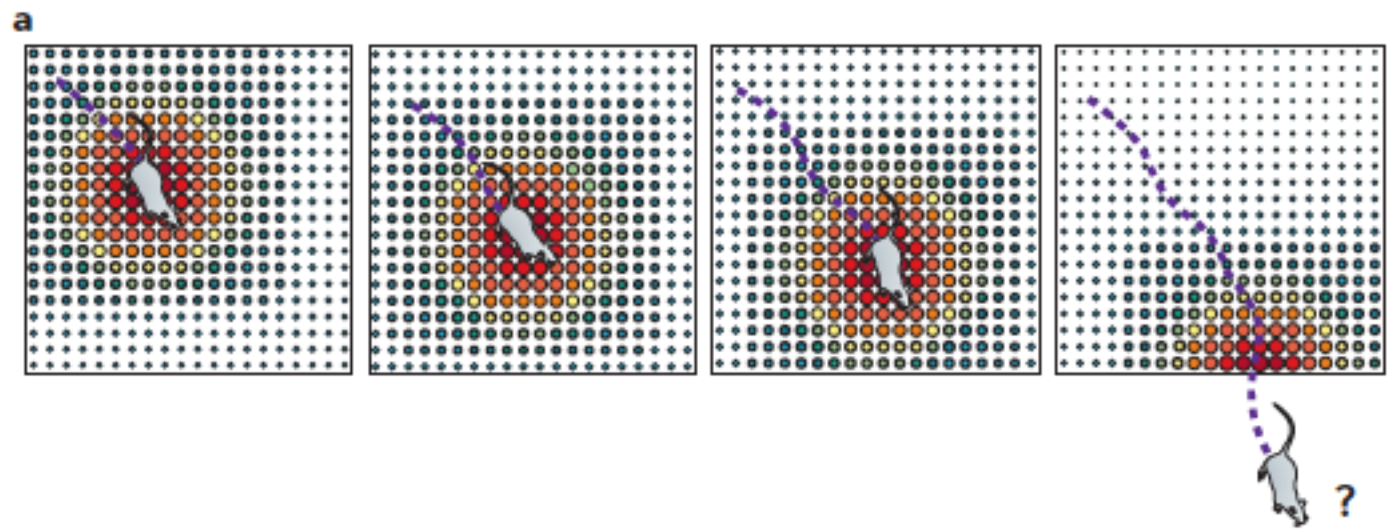
Bruce L. McNaughton[*¶], Francesco P. Battaglia[§], Ole Jensen[||], Edvard I. Moser[¶] and May-Britt Moser[¶]*

but *weirdness* at boundary

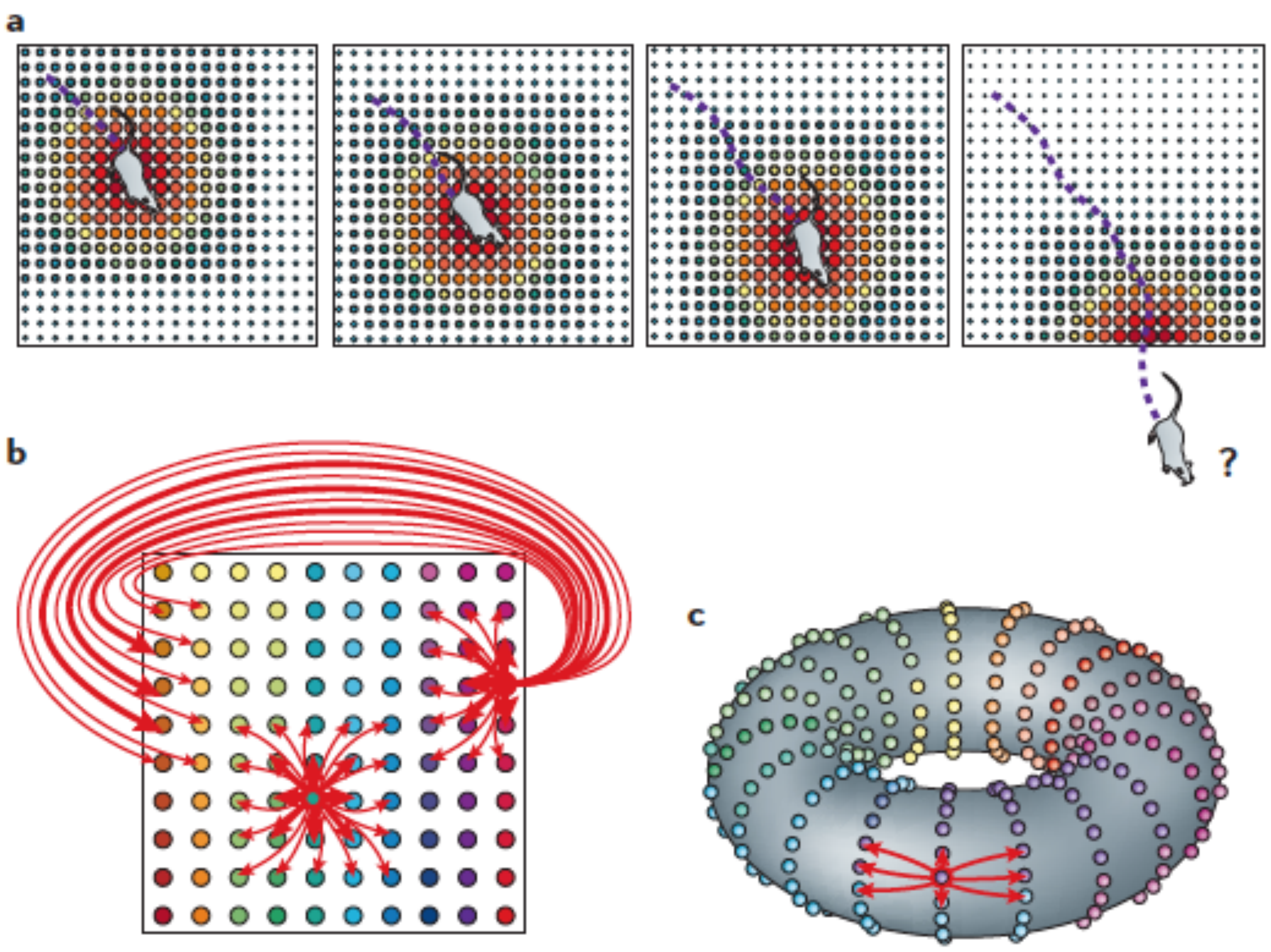
Two-D grid version



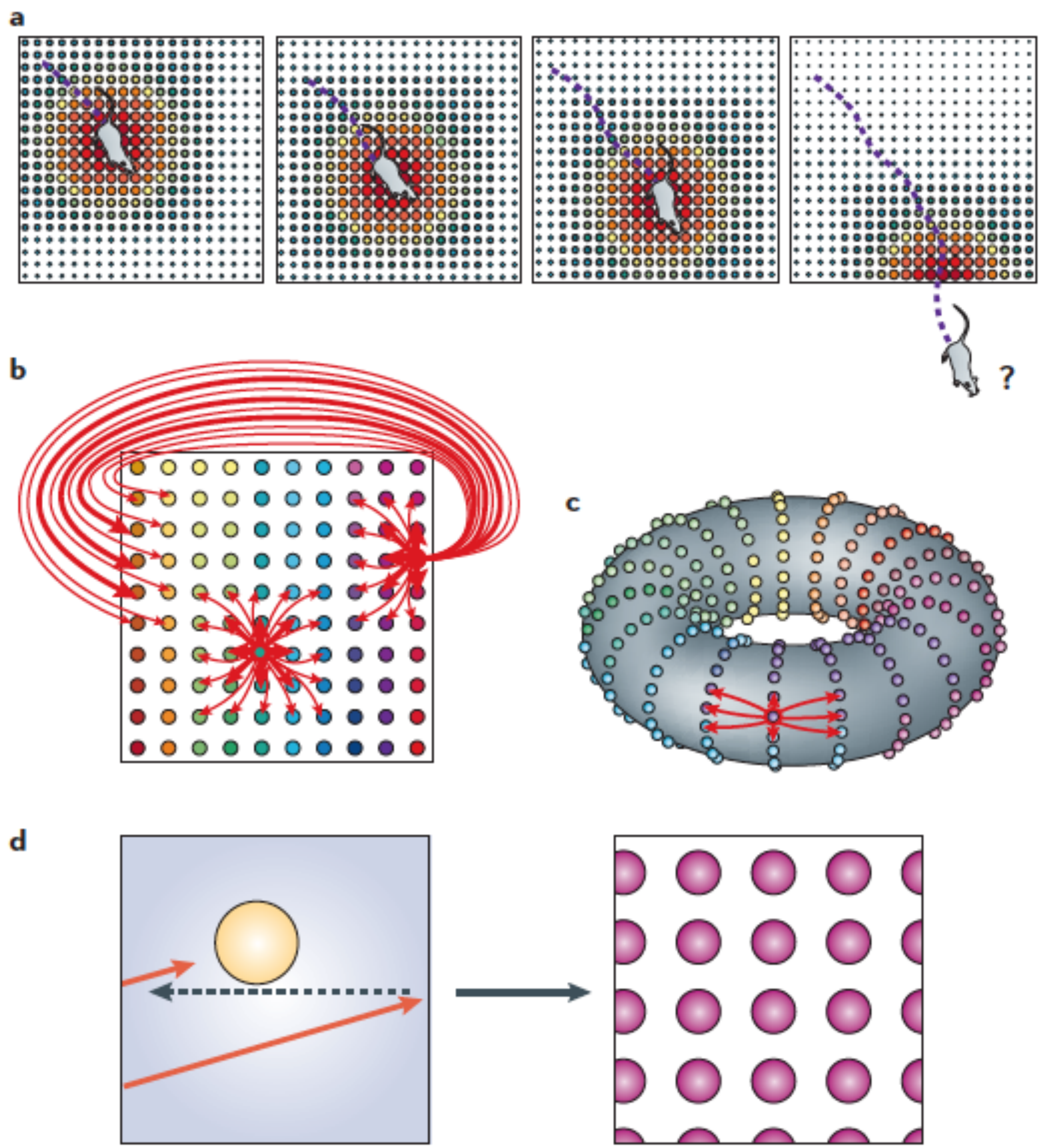
Models: Spatial Attractor of Path Integrator



Models: Spatial Attractor of Path Integrator



Models: Spatial Attractor of Path Integrator



Models: Spatial Attractor of Path Integrator

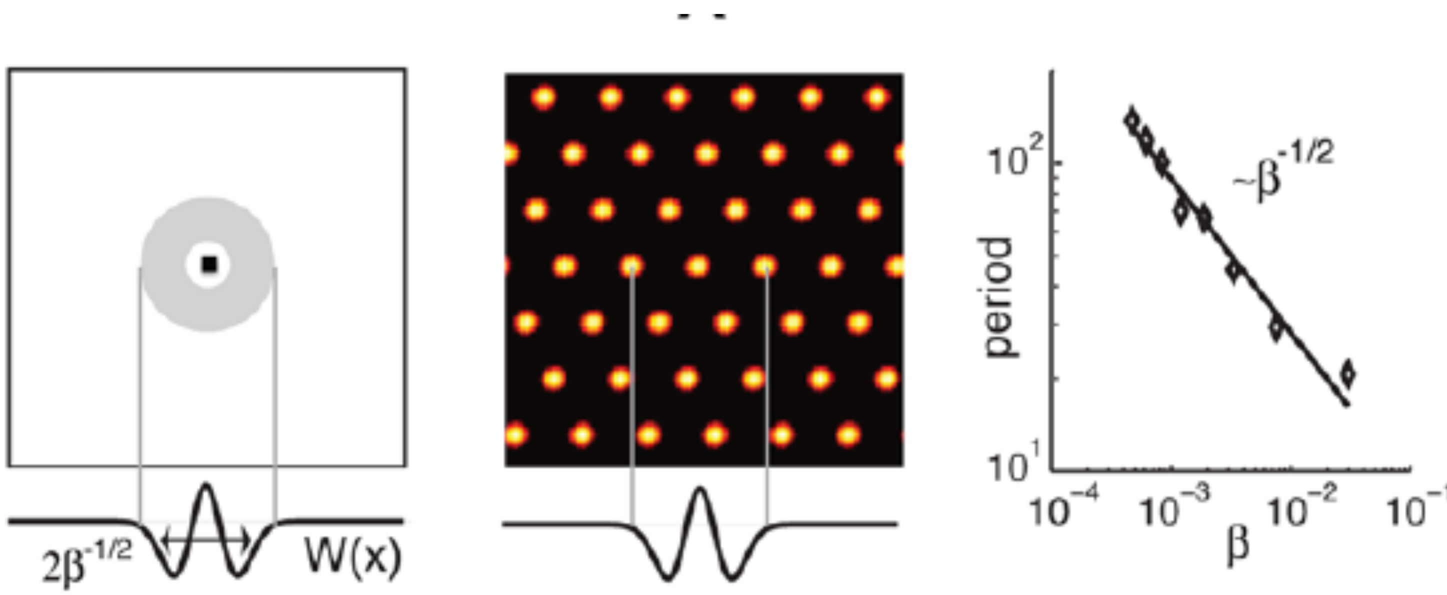
Accurate Path Integration in Continuous Attractor Network Models of Grid Cells

Yoram Burak[1,2*], Ila R. Fiete[2,3]

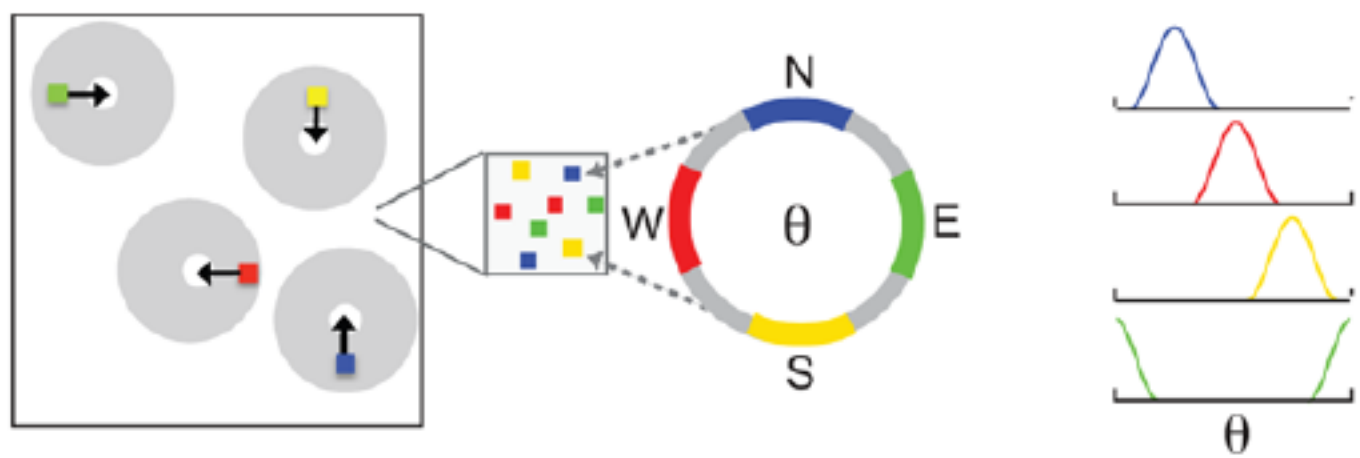
1 Center for Brain Science, Harvard University, Cambridge, Massachusetts, United States of America, 2 Kavli Institute for Theoretical Physics, University of California Santa Barbara, Santa Barbara, California, United States of America, 3 Computation and Neural Systems, Division of Biology, California Institute of Technology, Pasadena, California, United States of America

Square grid (128x128), with toroidal wraparound

inhibitory input from surround ring of neurons



dependence of emergent pattern on strength of inhibition



coupling to velocity (from subiculum?)

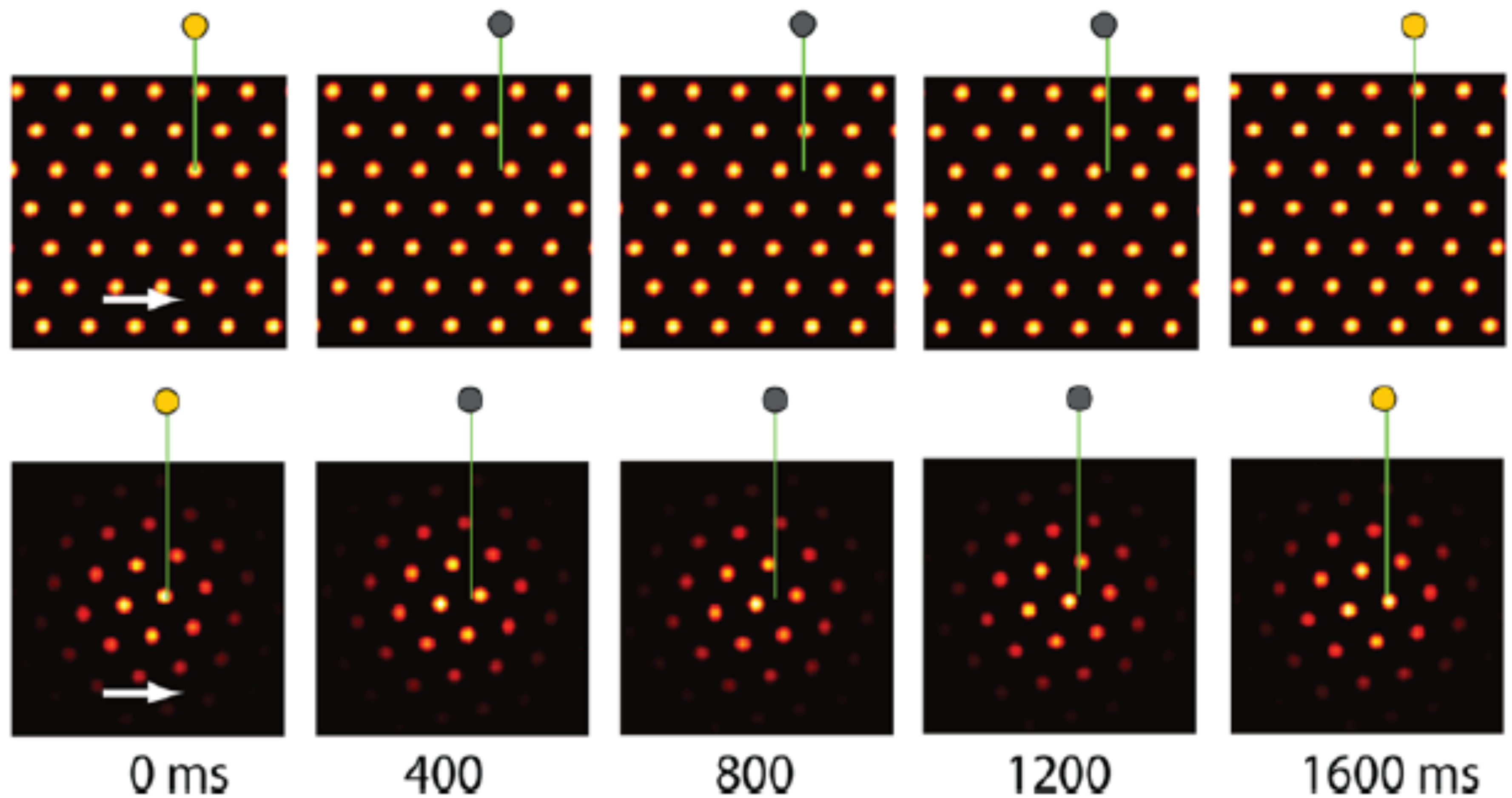
Models: Spatial Attractor of Path Integrator

Accurate Path Integration in Continuous Attractor Network Models of Grid Cells

Yoram Burak[1,2*], **Ila R. Fiete**[2,3]

1 Center for Brain Science, Harvard University, Cambridge, Massachusetts, United States of America, 2 Kavli Institute for Theoretical Physics, University of California Santa Barbara, Santa Barbara, California, United States of America, 3 Computation and Neural Systems, Division of Biology, California Institute of Technology, Pasadena, California, United States of America

periodic boundary



aperiodic boundary

Models: Spatial Attractor of Path Integrator

Accurate Path Integration in Continuous Attractor Network Models of Grid Cells

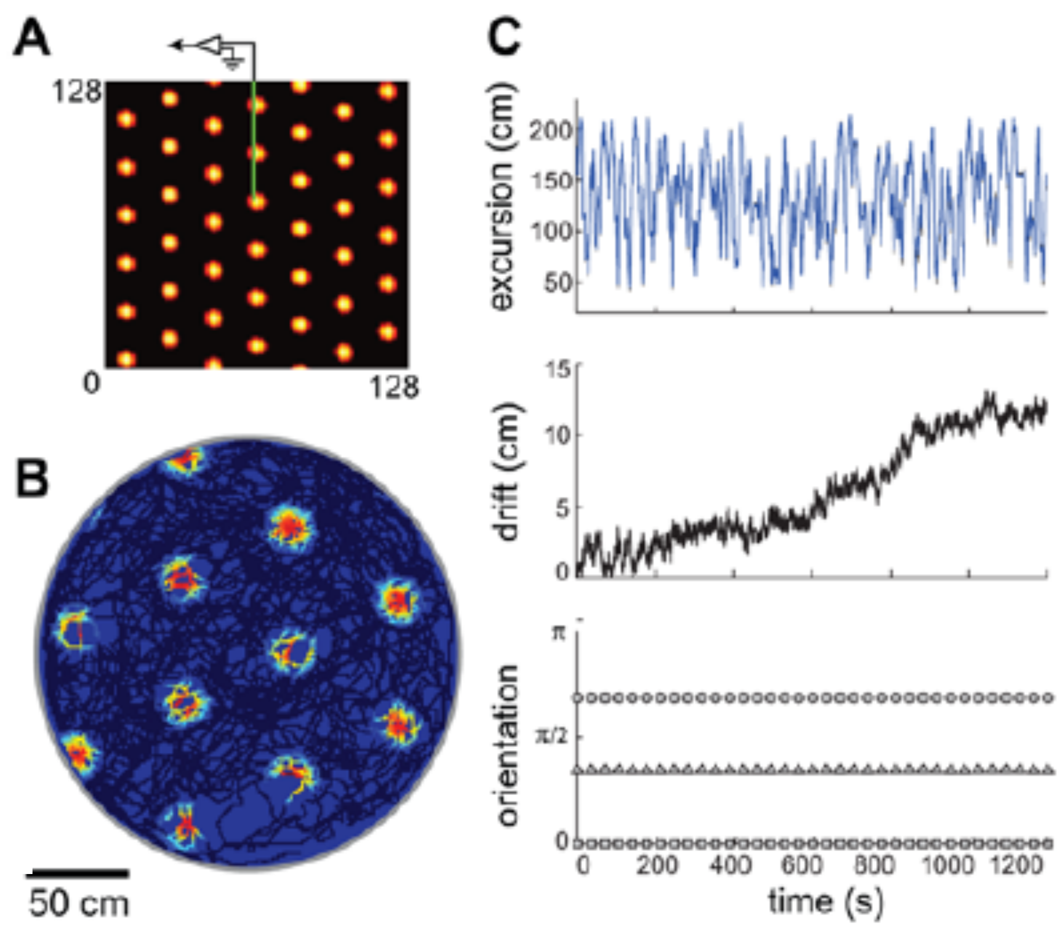
Yoram Burak[1,2]*, **Ila R. Fiete**[2,3]

1 Center for Brain Science, Harvard University, Cambridge, Massachusetts, United States of America, 2 Kavli Institute for Theoretical Physics, University of California Santa Barbara, Santa Barbara, California, United States of America, 3 Computation and Neural Systems, Division of Biology, California Institute of Technology, Pasadena, California, United States of America

Now driven by real rat motion data

instantaneous activity of model units

time-average grid-cell response of real rat



drift

Goal-Driven Models

Published as a conference paper at ICLR 2018

EMERGENCE OF GRID-LIKE REPRESENTATIONS BY TRAINING RECURRENT NEURAL NETWORKS TO PERFORM SPATIAL LOCALIZATION

Christopher J. Cueva,* Xue-Xin Wei*
Columbia University
New York, NY 10027, USA
{ccueva,weixxpku}@gmail.com

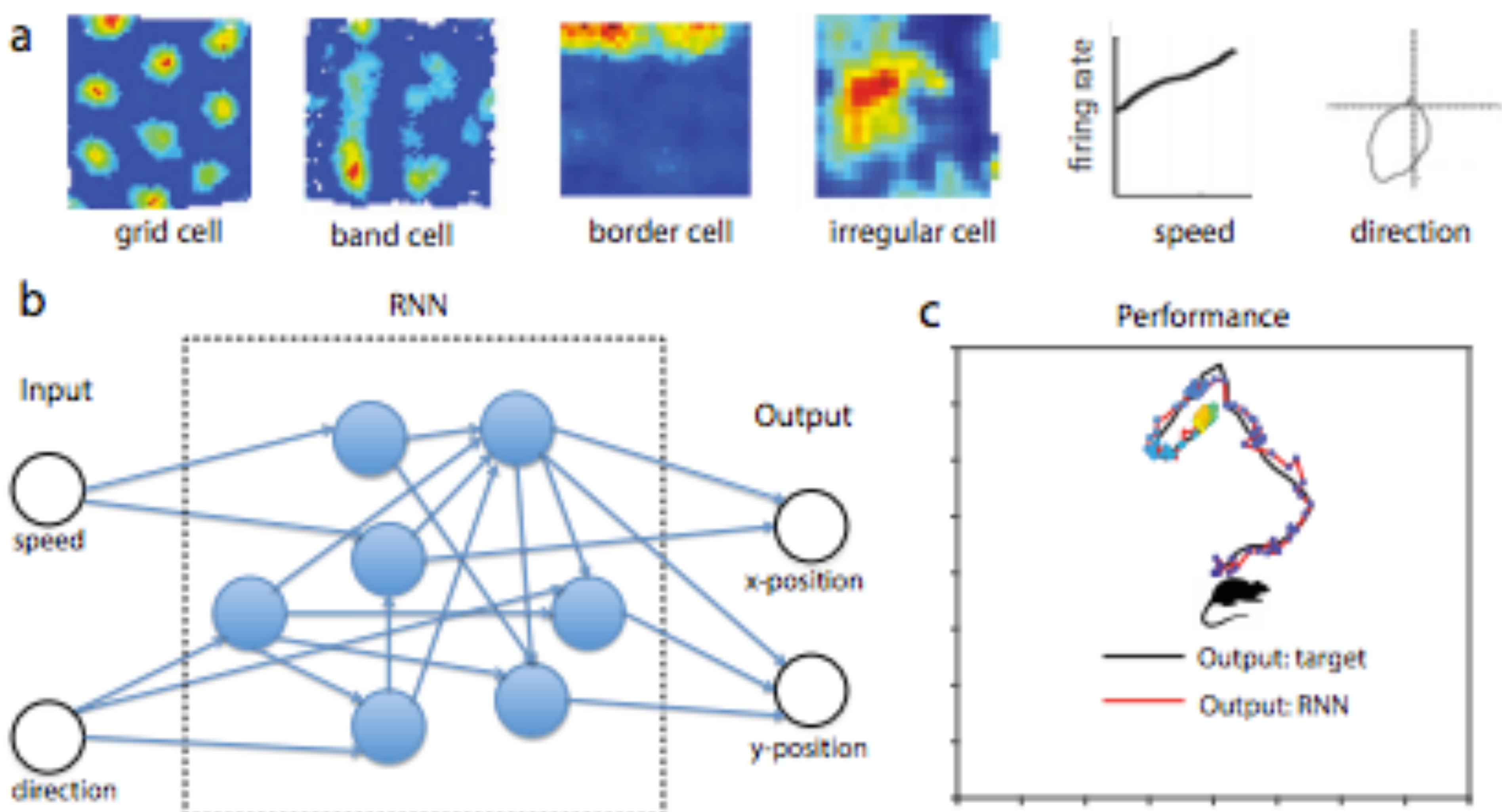
ABSTRACT

Decades of research on the neural code underlying spatial navigation have revealed a diverse set of neural response properties. The Entorhinal Cortex (EC) of the mammalian brain contains a rich set of spatial correlates, including grid cells which encode space using tessellating patterns. However, the mechanisms and functional significance of these spatial representations remain largely mysterious. As a new way to understand these neural representations, we trained recurrent neural networks (RNNs) to perform navigation tasks in 2D arenas based on velocity inputs. Surprisingly, we find that grid-like spatial response patterns emerge in trained networks, along with units that exhibit other spatial correlates, including border cells and band-like cells. All these different functional types of neurons have been observed experimentally. The order of the emergence of grid-like and border cells is also consistent with observations from developmental studies. Together, our results suggest that grid cells, border cells and others as observed in EC may be a natural solution for representing space efficiently given the predominant recurrent connections in the neural circuits.

Goal-Driven Models

EMERGENCE OF GRID-LIKE REPRESENTATIONS BY TRAINING RECURRENT NEURAL NETWORKS TO PERFORM SPATIAL LOCALIZATION

Christopher J. Cueva,* Xue-Xin Wei*
Columbia University
New York, NY 10027, USA
{ccueva,weixxpku}@gmail.com



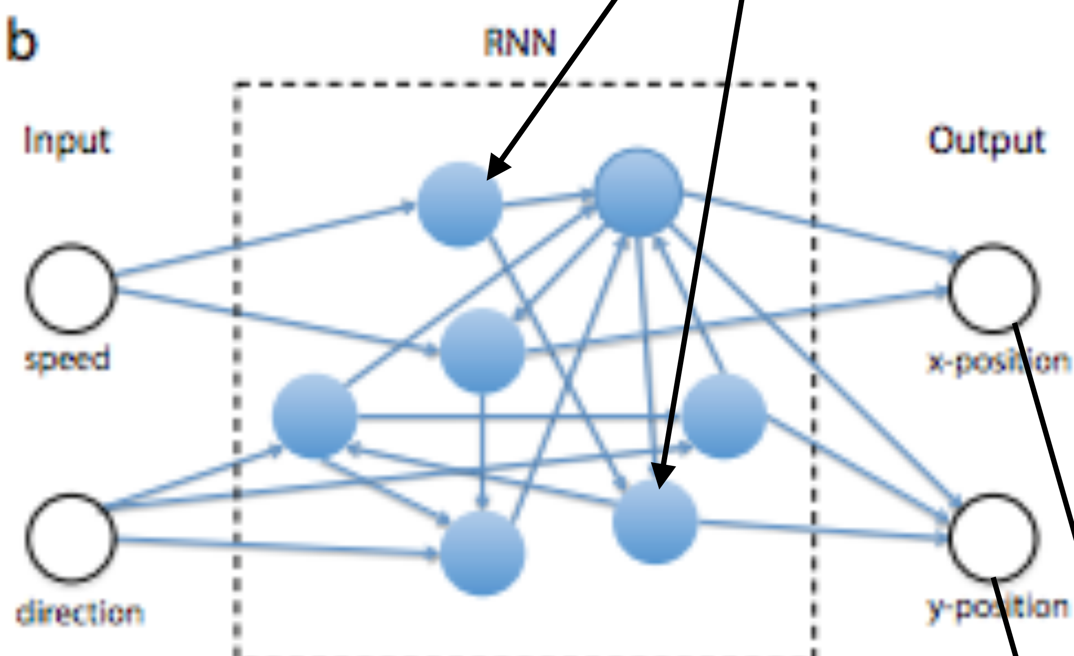
Goal-Driven Models

EMERGENCE OF GRID-LIKE REPRESENTATIONS BY TRAINING RECURRENT NEURAL NETWORKS TO PERFORM SPATIAL LOCALIZATION

Christopher J. Cueva,* Xue-Xin Wei*
Columbia University
New York, NY 10027, USA
{ccueva,weixxpku}@gmail.com

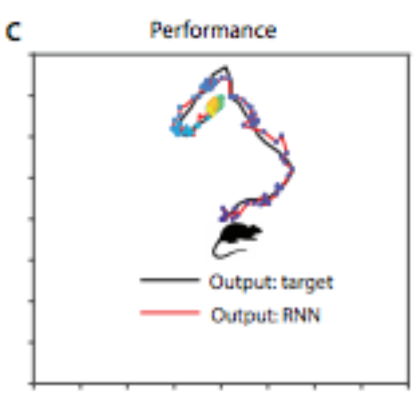
RNN governing equation:

$$\tau \frac{dx_i}{dt} = -x_i(t) + \sum_{j=1}^{N_{\text{rec}}} W_{ij}^{\text{rec}} \tanh(x_i(t)) + \sum_{k=1}^{N_{\text{inp}}} W_{ik}^{\text{inp}} I_k(t) + \xi_i(t)$$



x_i's are the internal neurons

$N_{rec} = 10($



weights and biases t
produced desired c

y_j's are the desired readouts

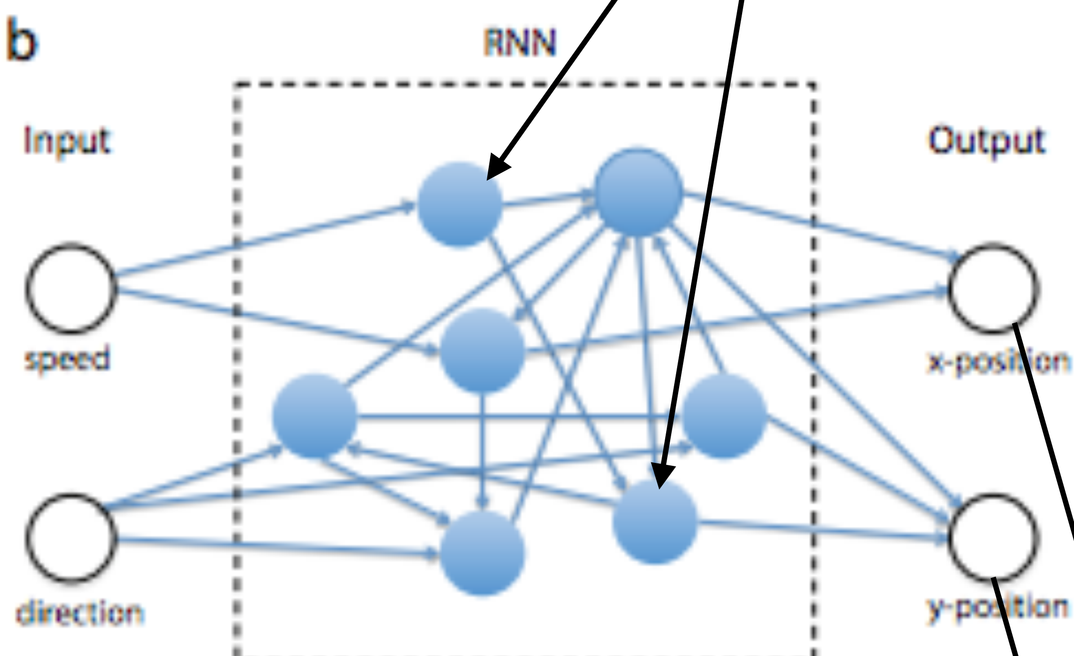
$$y_j(t) = \sum_{i=1}^{N_{\text{rec}}} W_{ji}^{\text{out}} \tanh(x_i(t))$$

EMERGENCE OF GRID-LIKE REPRESENTATIONS BY TRAINING RECURRENT NEURAL NETWORKS TO PERFORM SPATIAL LOCALIZATION

Christopher J. Cueva,* Xue-Xin Wei*
Columbia University
New York, NY 10027, USA
{ccueva,weixxpku}@gmail.com

RNN governing equation:

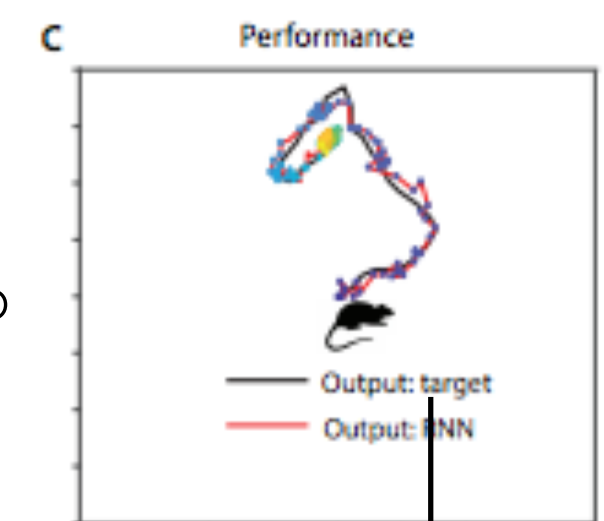
$$\tau \frac{dx_i}{dt} = -x_i(t) + \sum_{j=1}^{N_{\text{rec}}} W_{ij}^{\text{rec}} \tanh(x_i(t)) + \sum_{k=1}^{N_{\text{inp}}} W_{ik}^{\text{inp}} I_k(t) + \xi_i(t)$$



x_i's are the internal neurons

$$N_{rec} = 100$$

weights and biases trained to produced desired output →



y_j's are the desired readouts

$$y_j(t) = \sum_{i=1}^{N_{\text{rec}}} W_{ji}^{\text{out}} \tanh(x_i(t))$$

generated as modified Brownian motion

Goal-Driven Models

EMERGENCE OF GRID-LIKE REPRESENTATIONS BY TRAINING RECURRENT NEURAL NETWORKS TO PERFORM SPATIAL LOCALIZATION

Christopher J. Cueva,* Xue-Xin Wei*
Columbia University
New York, NY 10027, USA
{ccueva,weixxpku}@gmail.com

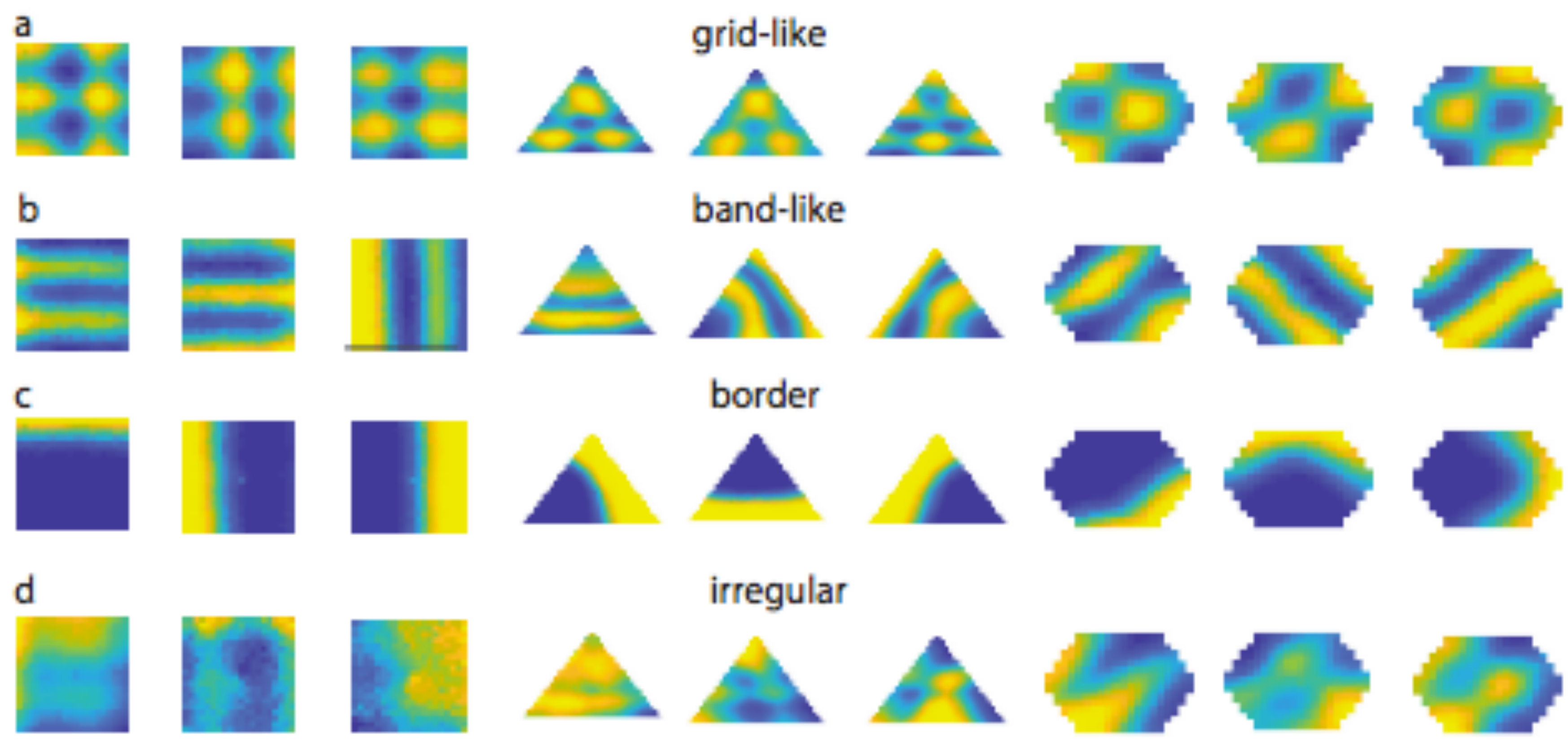


Figure 2: Different types of spatial selective responses of units in the trained RNN. Example simulation results for three different environments (square, triangular, hexagon) are presented. Blue (yellow) represents low (high) activity. **a)** Grid-like responses. **b)** Band-like responses; **c)** Border-related responses; **d)** Spatially irregular responses. These responses can be spatially selective but they do not form a regular pattern defined in the conventional sense.

Goal-Driven Models

EMERGENCE OF GRID-LIKE REPRESENTATIONS BY TRAINING RECURRENT NEURAL NETWORKS TO PERFORM SPATIAL LOCALIZATION

Christopher J. Cueva,* Xue-Xin Wei*
Columbia University
New York, NY 10027, USA
{ccueva,weixxpku}@gmail.com

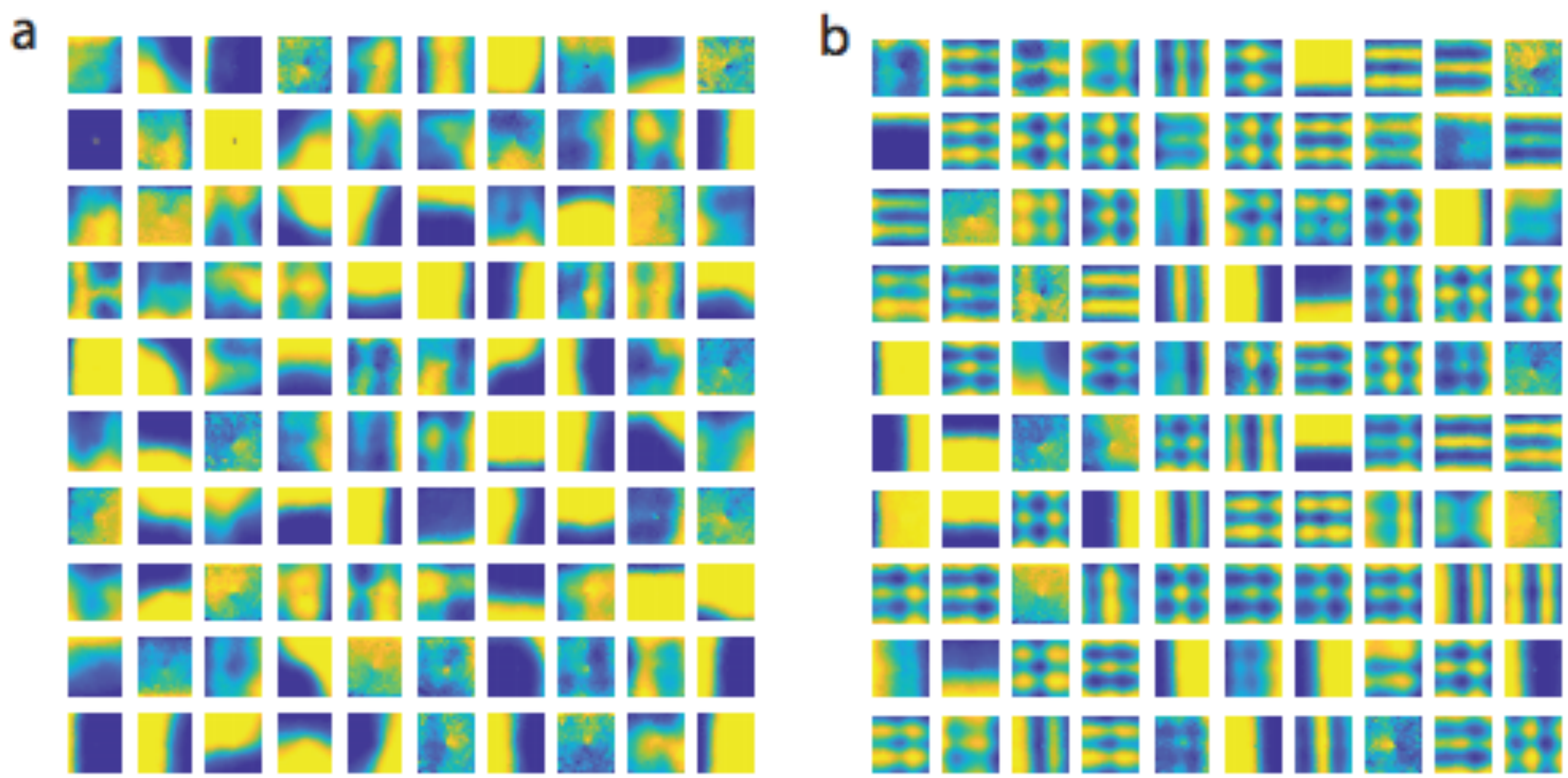


Figure 5: Complete set of spatial response profiles for 100 neurons in a RNN trained in a square environment. **a)** Without proper regularization, complex and periodic spatial response patterns do not emerge. **b)** With proper regularization, a rich set of periodic response patterns emerge, including grid-like responses. Regularization can also be adjusted to achieve spatial profiles intermediate between these two examples.

Goal-Driven Models

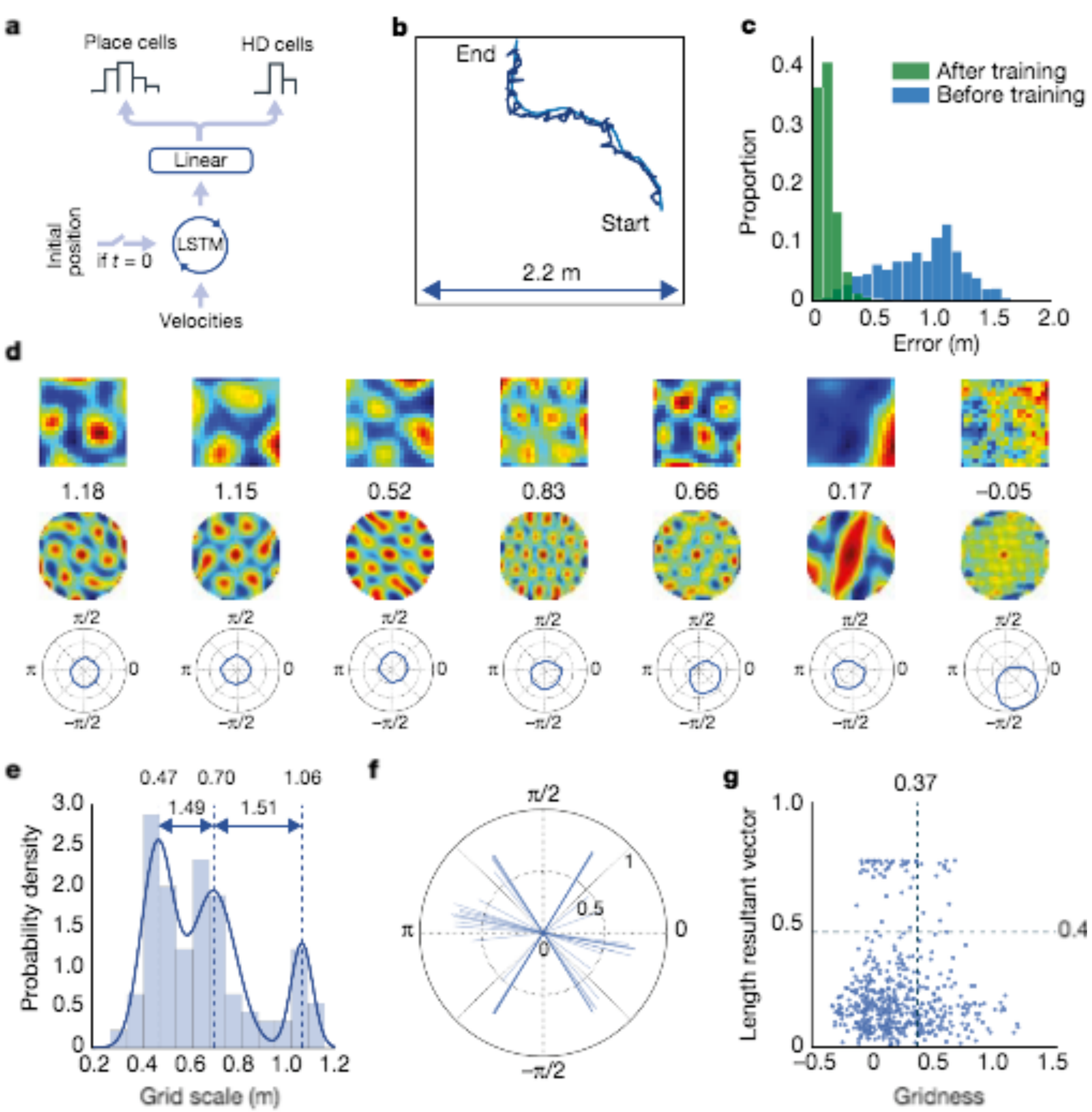
LETTER

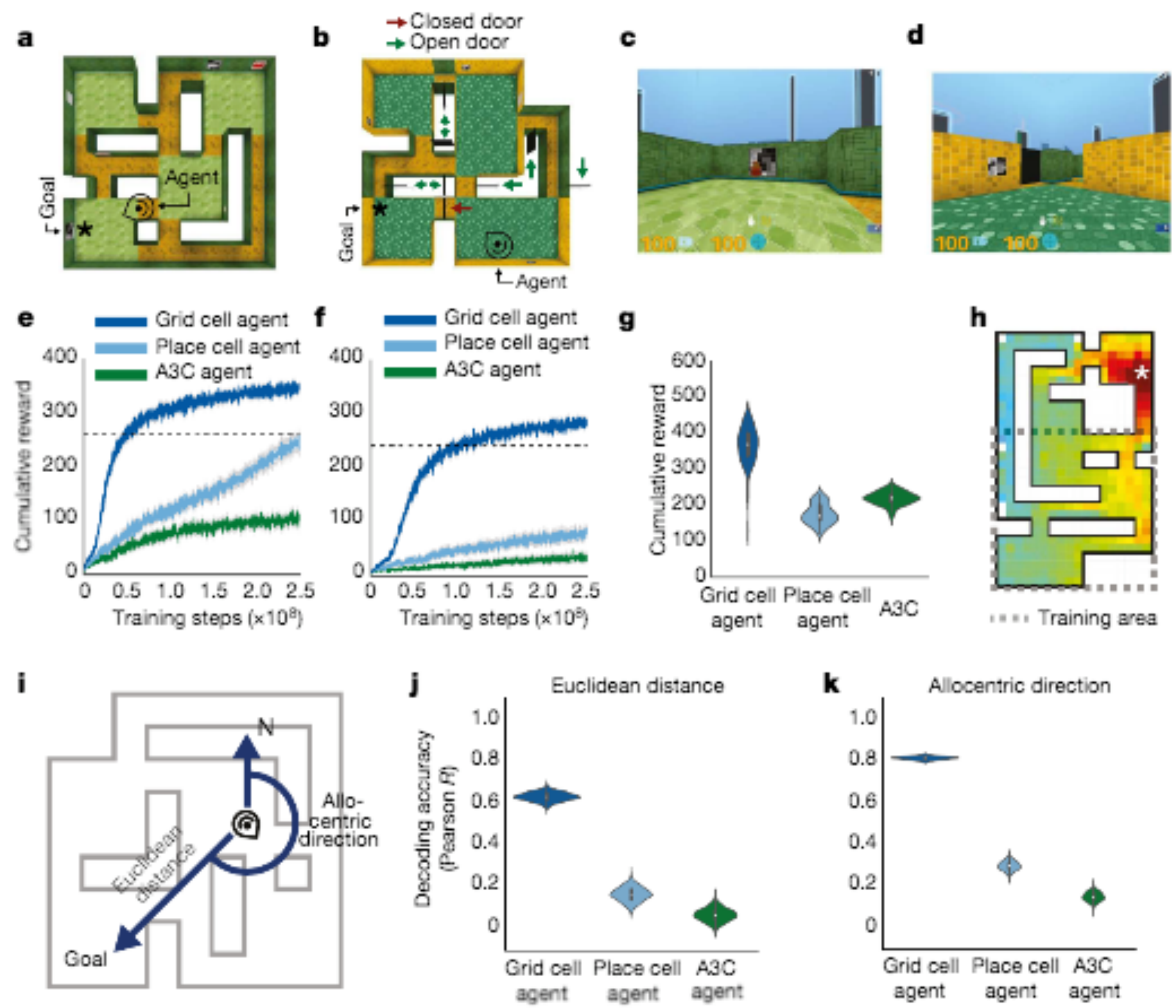
https://doi.org/10.1038/s41586-018-0102-6

Vector-based navigation using grid-like representations in artificial agents

Andrea Banino[1,2,3,5]*, Caswell Barry[2,5]*, Benigno Uria[1], Charles Blundell[1], Timothy Lillicrap[1], Piotr Mirowski[1], Alexander Pritzel[1], Martin J. Chadwick[1], Thomas Degris[1], Joseph Modayil[1], Greg Wayne[1], Hubert Soyer[1], Fabio Viola[1], Brian Zhang[1], Ross Goroshin[1], Neil Rabinowitz[1], Razvan Pascanu[1], Charlie Beattie[1], Stig Petersen[1], Amir Sadik[1], Stephen Gaffney[1], Helen King[1], Koray Kavukcuoglu[1], Demis Hassabis[1,4], Raia Hadsell[1] & Dharshan Kumaran[1,3]*

Similar results from another group about the same time.

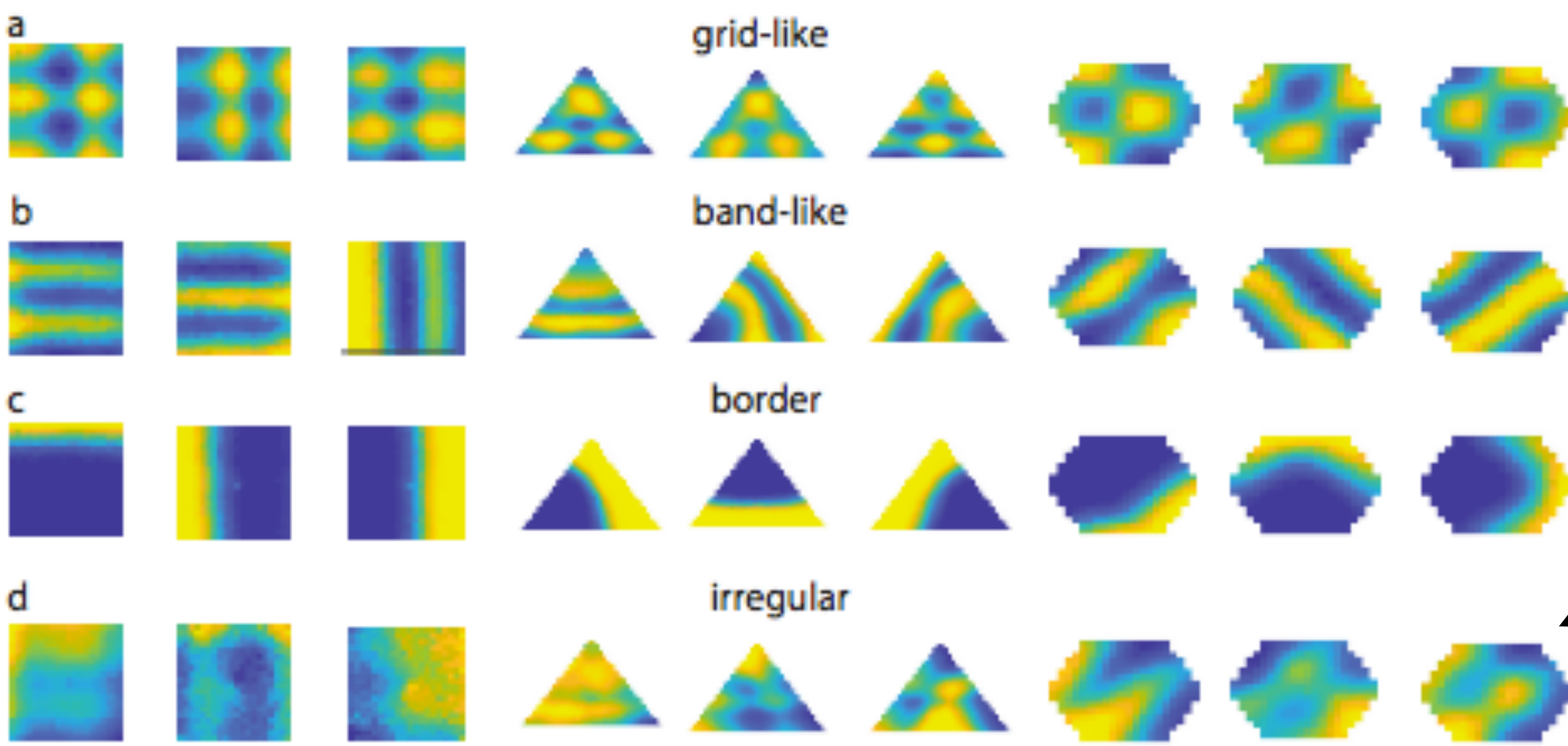




Goal-Driven Models

EMERGENCE OF GRID-LIKE REPRESENTATIONS BY TRAINING RECURRENT NEURAL NETWORKS TO PERFORM SPATIAL LOCALIZATION

Christopher J. Cueva,* Xue-Xin Wei*
Columbia University
New York, NY 10027, USA
{ccueva,weixxpku}@gmail.com



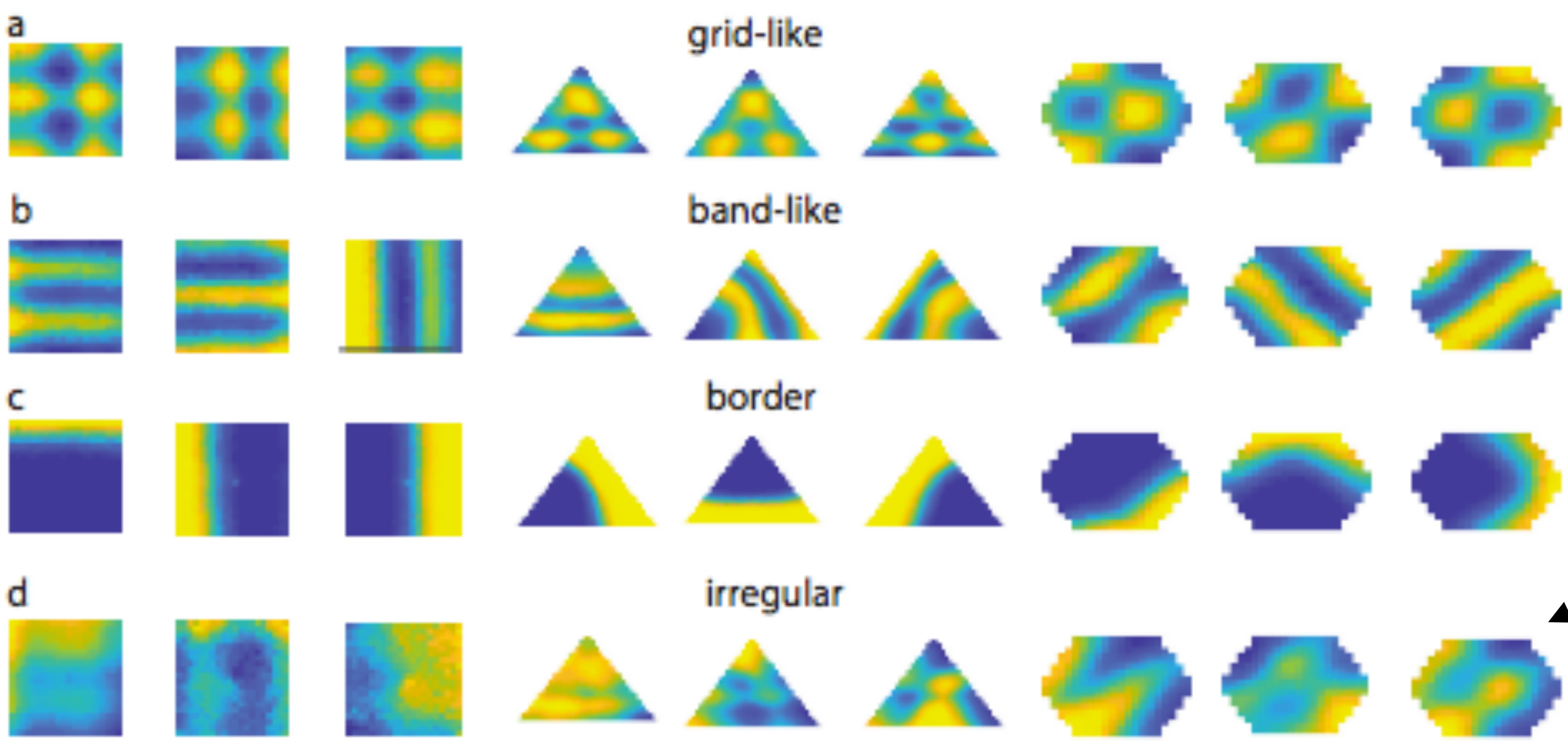
Many units in model are irregular — neither grid-like nor band-like nor border-like …

gure 2: Different types of spatial selective responses of units in the trained RNN. Example sim-
tion results for three different environments (square, triangular, hexagon) are presented. Blue
ellow) represents low (high) activity. **a)** Grid-like responses. **b)** Band-like responses; **c)** Border-
ated responses; **d)** Spatially irregular responses. These responses can be spatially selective but
y do not form a regular pattern defined in the conventional sense.

Goal-Driven Models

EMERGENCE OF GRID-LIKE REPRESENTATIONS BY TRAINING RECURRENT NEURAL NETWORKS TO PERFORM SPATIAL LOCALIZATION

Christopher J. Cueva,* Xue-Xin Wei*
Columbia University
New York, NY 10027, USA
{ccueva,weixxpku}@gmail.com



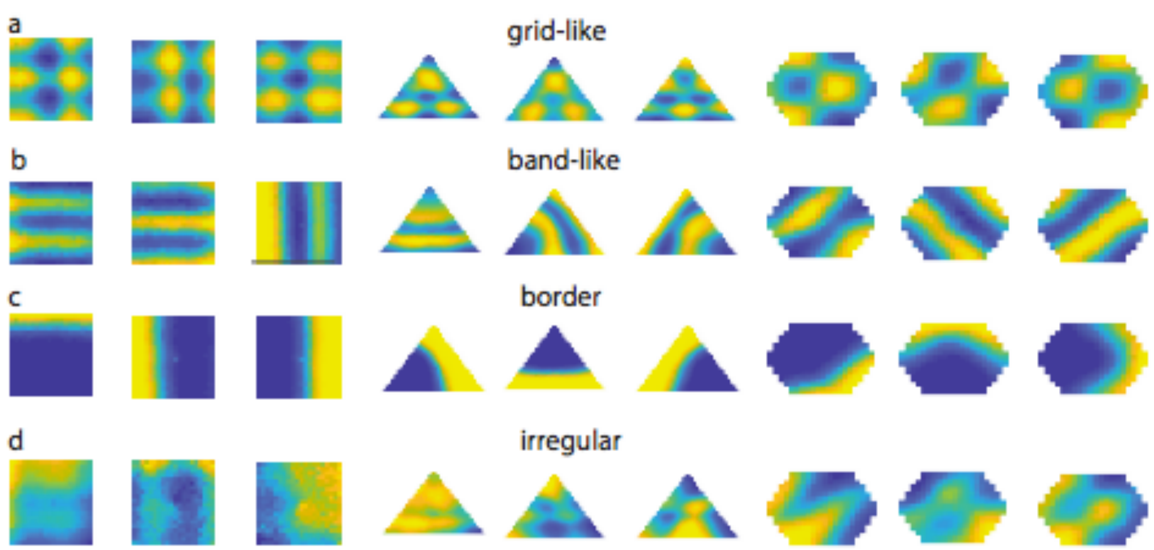
gure 2: Different types of spatial selective responses of units in the trained RNN. Example sim-
ation results for three different environments (square, triangular, hexagon) are presented. Blue
ellow) represents low (high) activity. **a)** Grid-like responses. **b)** Band-like responses; **c)** Border-
ated responses; **d)** Spatially irregular responses. These responses can be spatially selective but
y do not form a regular pattern defined in the conventional sense.

Many units in model are irregular — neither grid-like nor band-like nor border-like …

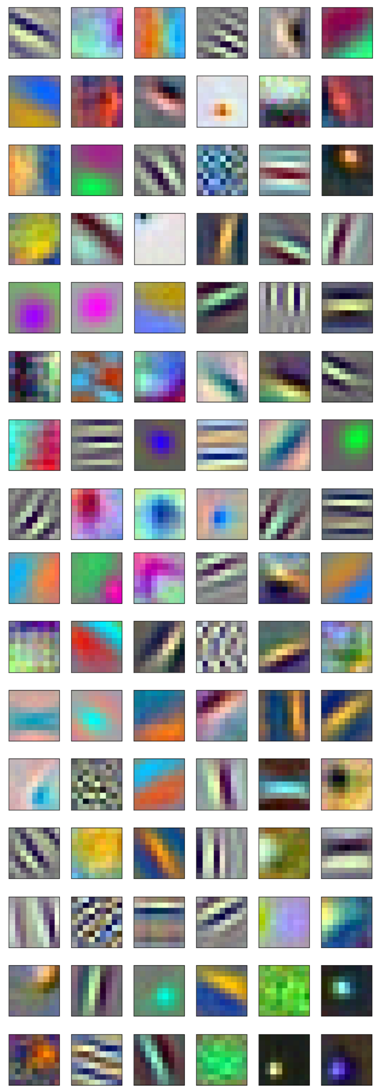
… but this is actually true in real entorhinal cortex as well. According to Lisa Giacomo, perhaps 70% of ERC cells are "irregular".

Analogy to visual system results

In both cases, striking qualitative features of "characteristic neurons" that neuroscientists feel are important can be shown to just "emerge" from the system achieving down-stream computational goal …



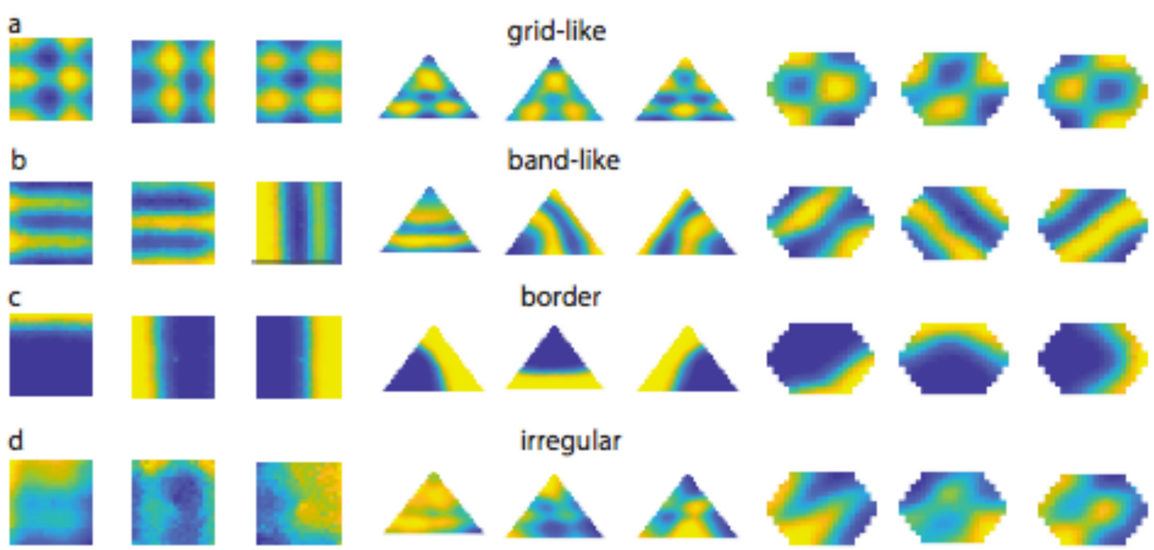
grid-cell-like tuning in navigation-based NN model of ERC



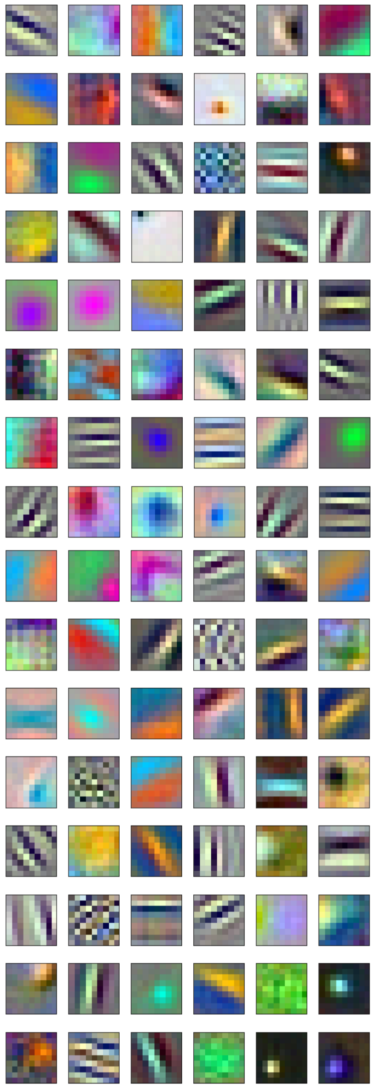
Gabor-like and center-surround tuning in early layer of categorization-based NN model of ventral stream

Analogy to visual system results

In both cases, striking qualitative features of "characteristic neurons" that neuroscientists feel are important can be shown to just "emerge" from the system achieving down-stream computational goal …



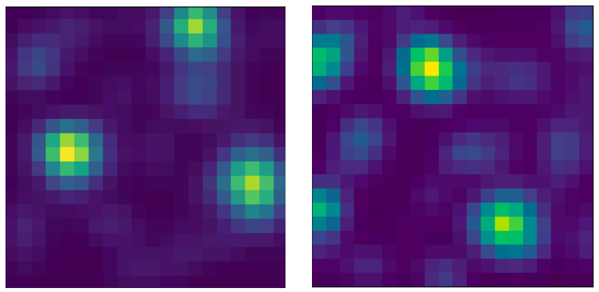
grid-cell-like tuning in navigation-based NN model of ERC



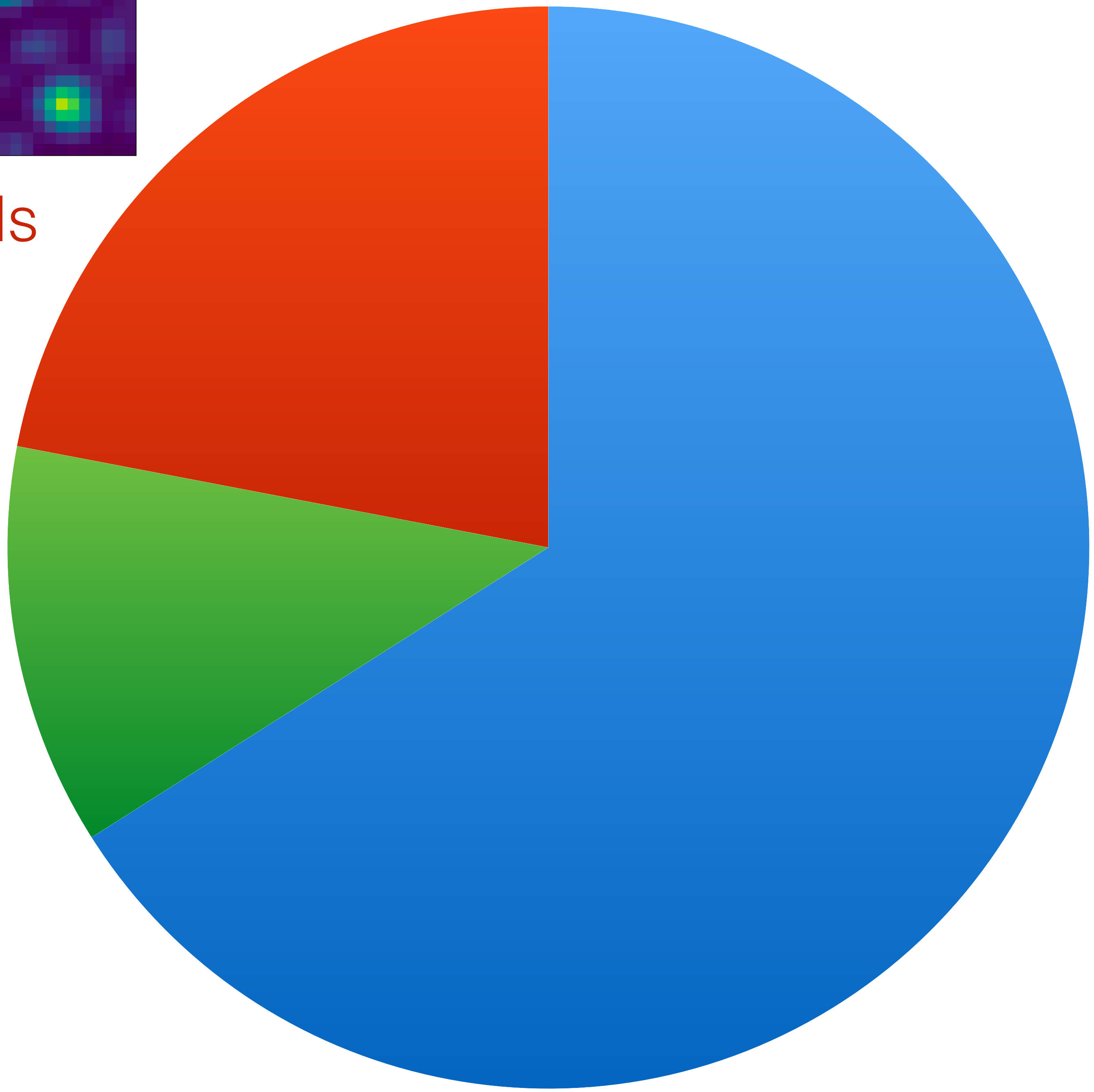
Gabor-like and center-surround tuning in early layer of categorization-based NN model of ventral stream

… but actually, just as interesting, these goal-driven NN models have many "non-characteristic" units that differ from the characteristic neurons — and in fact, ***so do the real brain areas***. So, perhaps the goal-driven models go substantially beyond the intuitions of neuroscientists in a way that is brain-like.

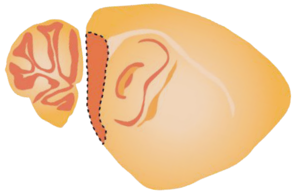
Accounting for heterogeneous code?



Grid Cells



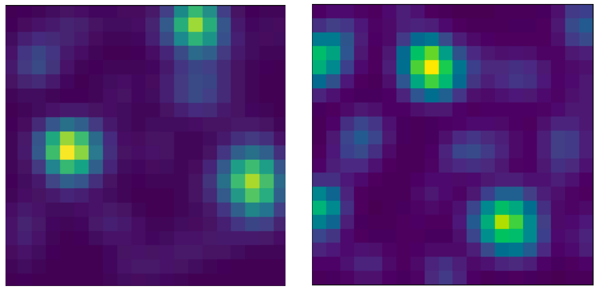
Data from: *Mallory et al. 2021*



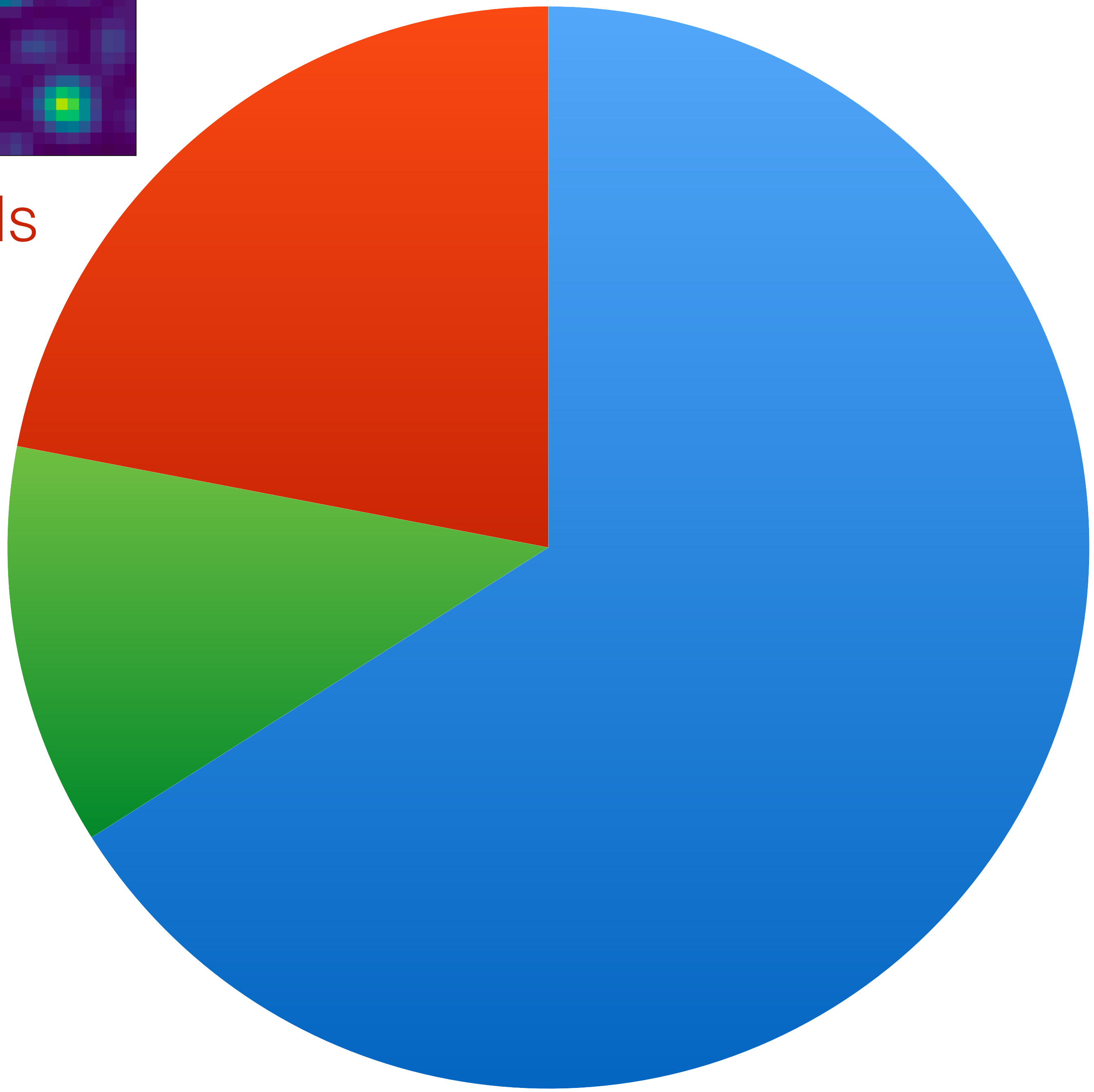
Caitlin Mallory

Accounting for heterogeneous code?

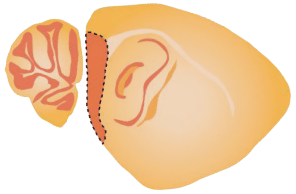
More like ~2-3%!



Grid Cells

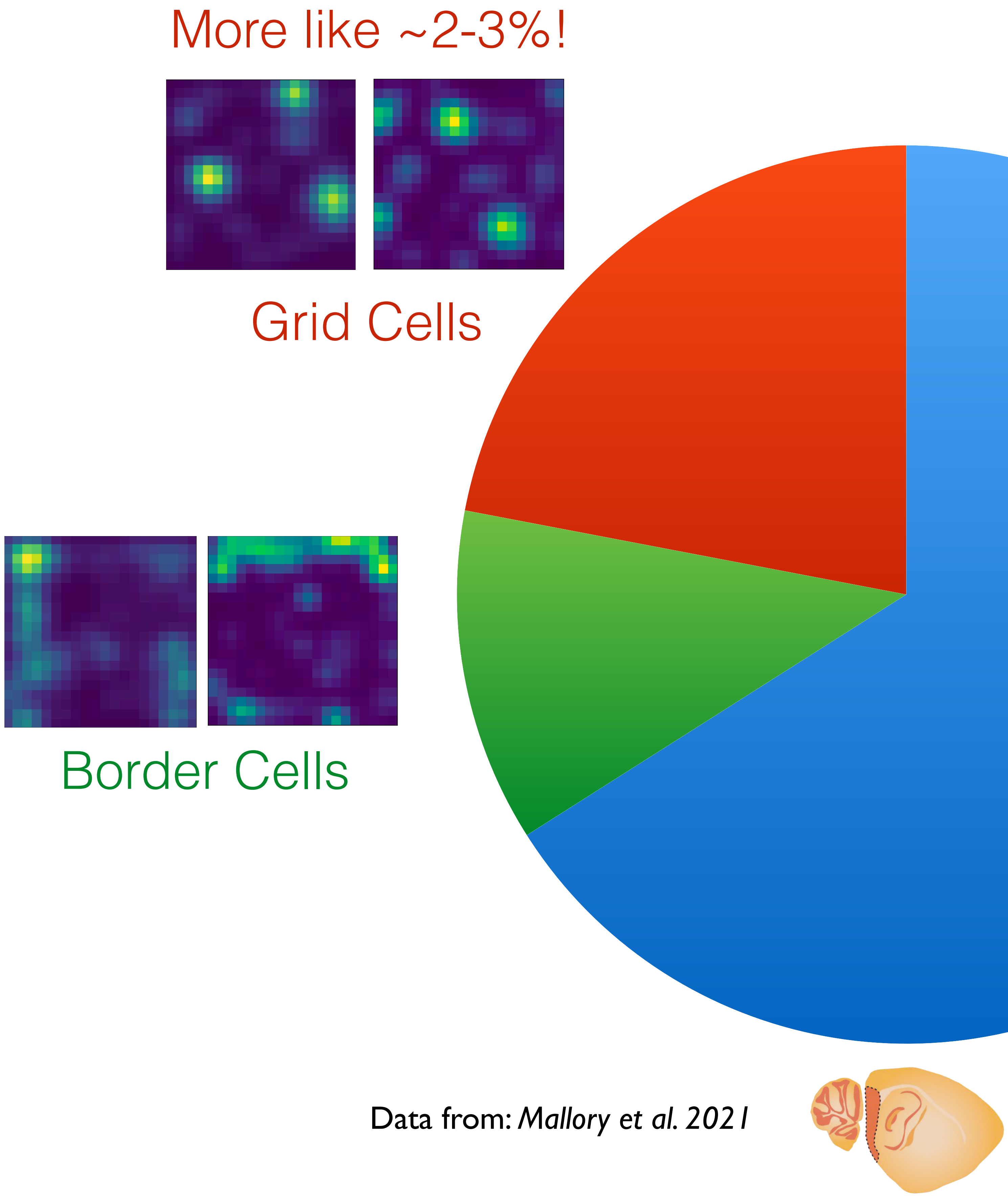


Data from: *Mallory et al. 2021*



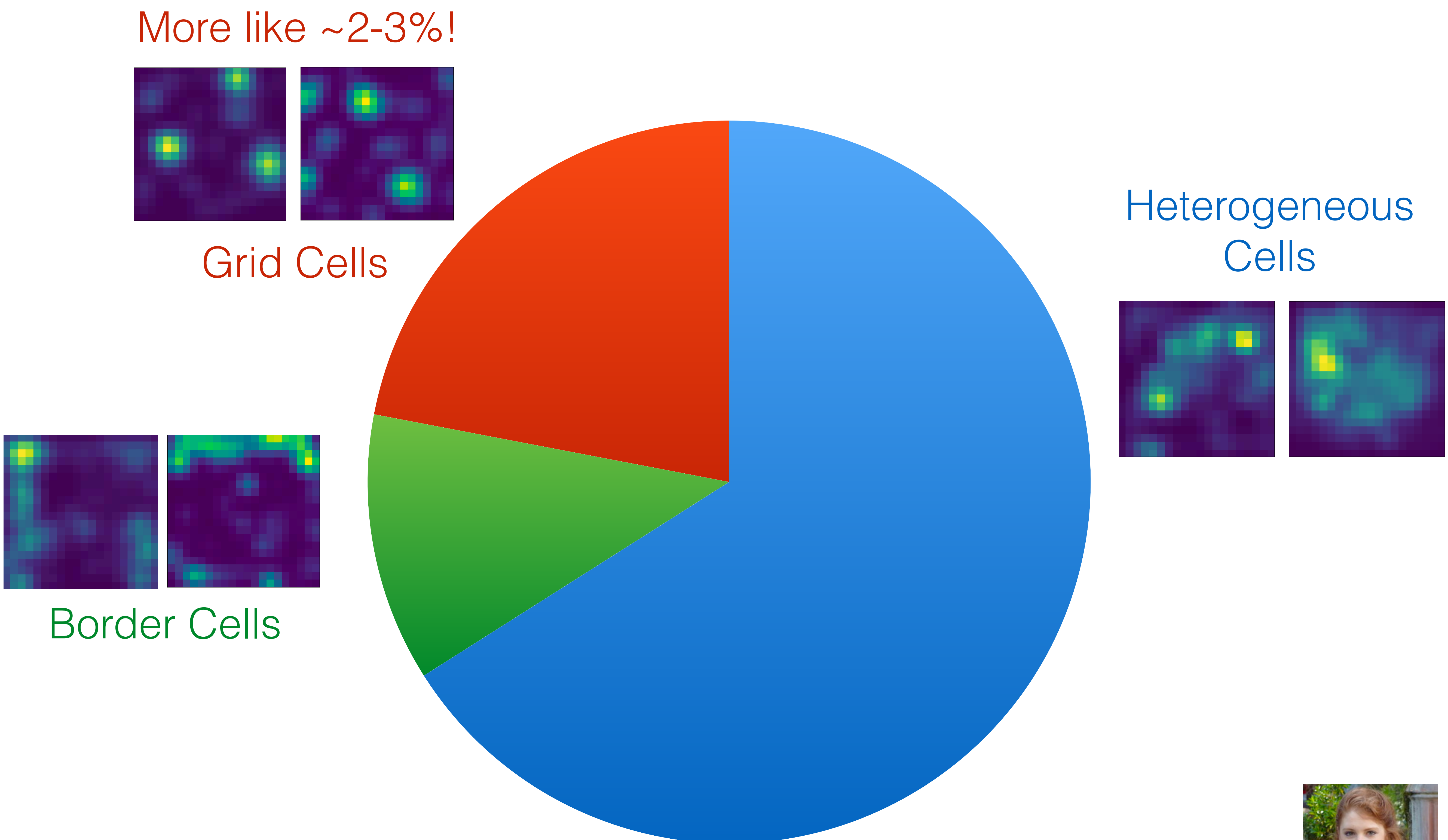
Caitlin Mallory

Accounting for heterogeneous code?

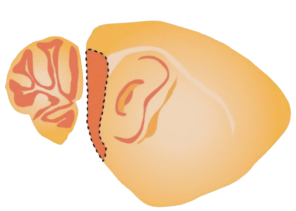



Caitlin Mallory

Accounting for heterogeneous code?

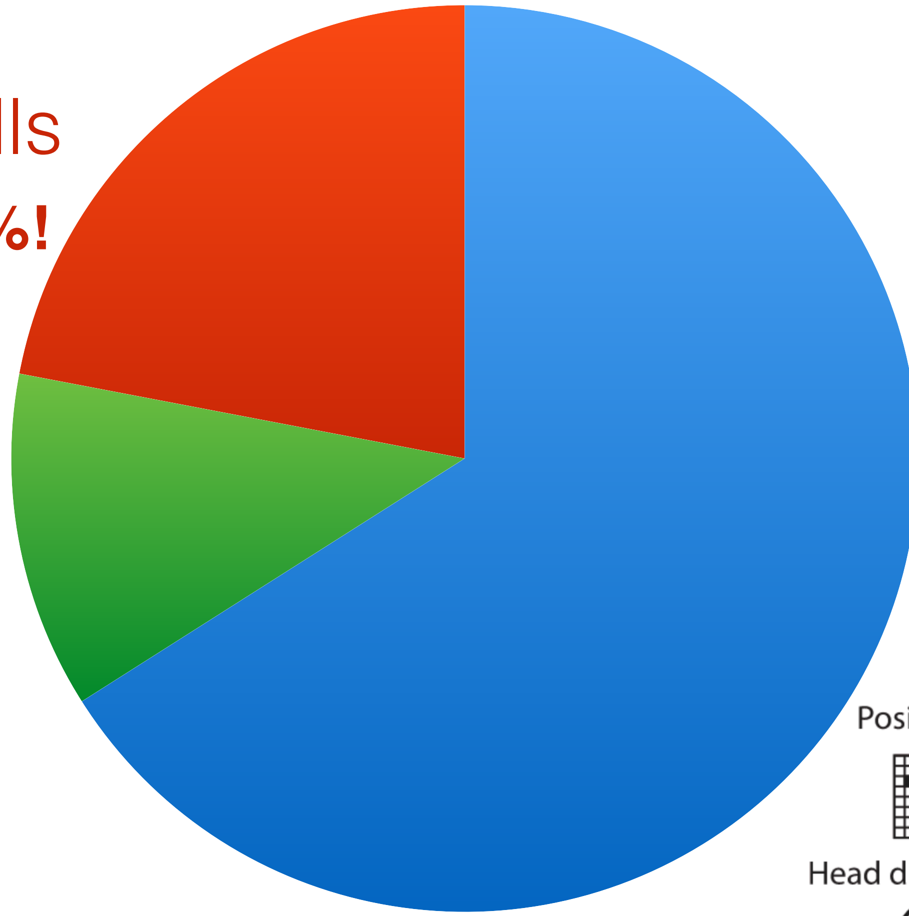


Data from: *Mallory et al. 2021*

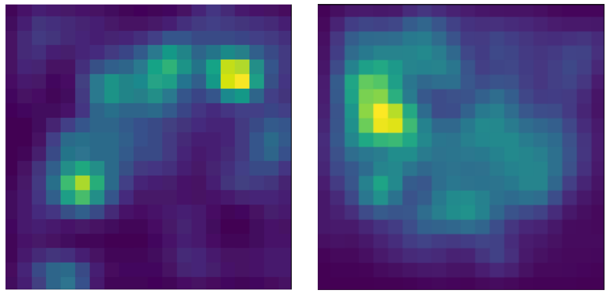


Caitlin Mallory

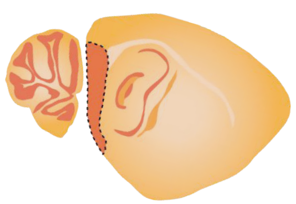
Accounting for heterogeneous code?

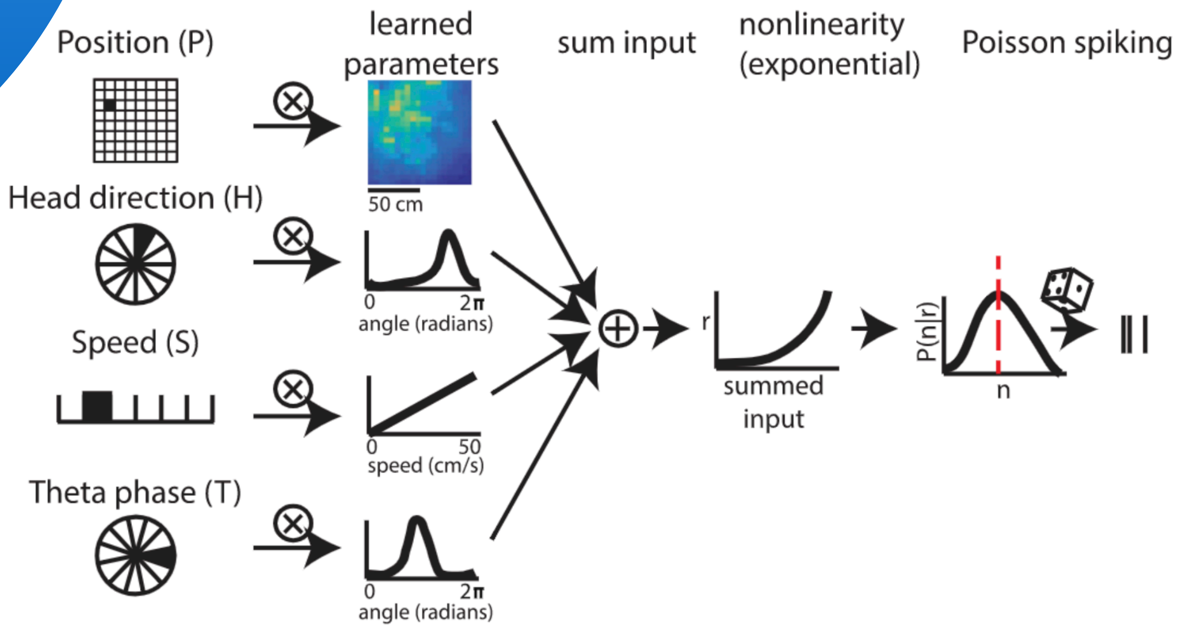


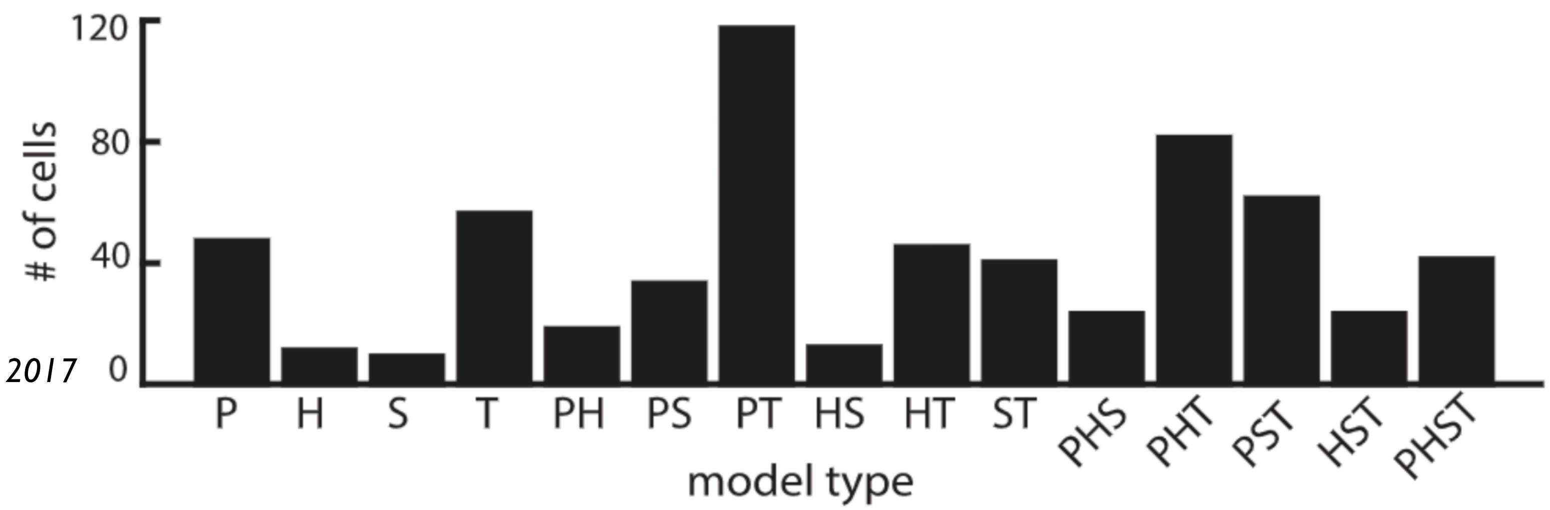
Heterogeneous Cells



Data from: *Mallory et al. 2021*







Hardcastle et al. 2017

Kiah Hardcastle

Surya Ganguli

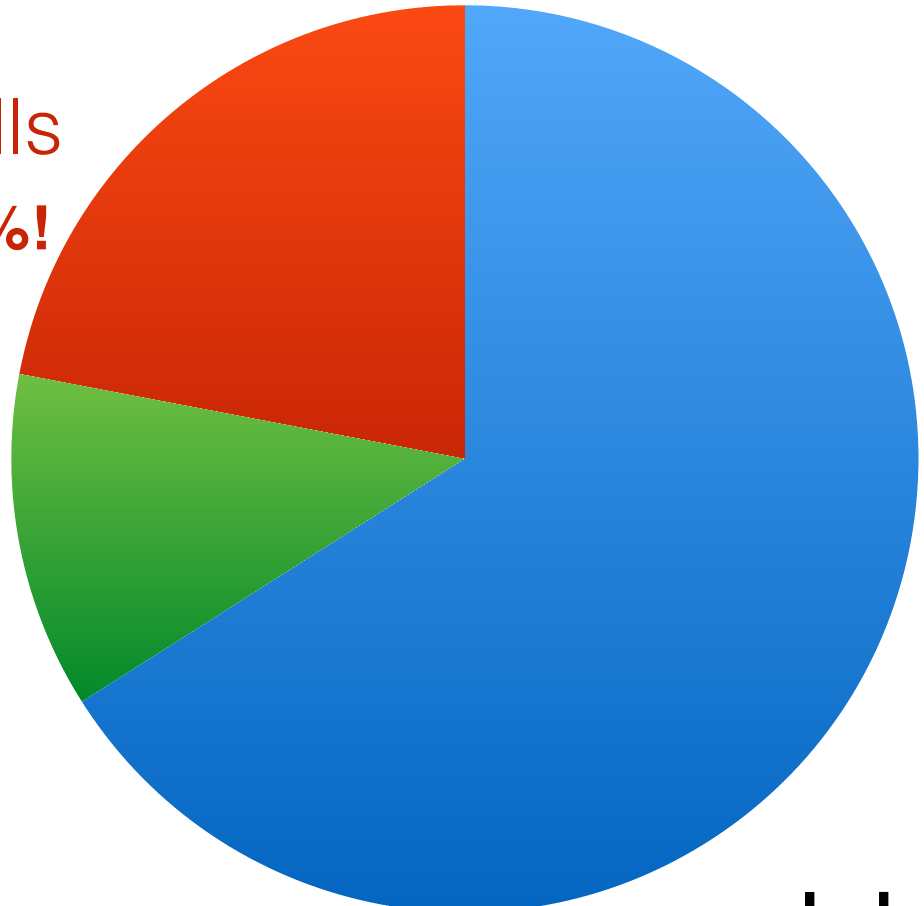
Lisa Giocomo

Accounting for heterogeneous code?

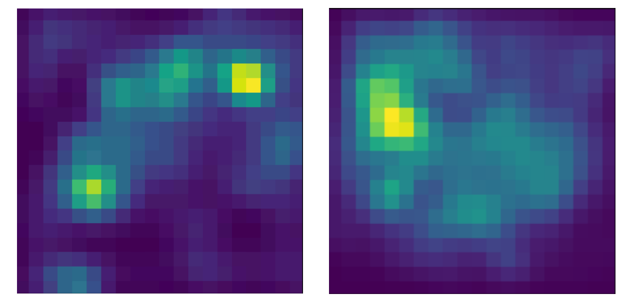
Grid Cells

More like ~2-3%!

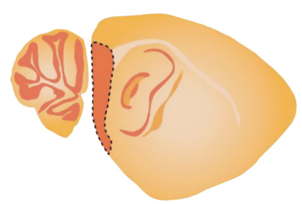
Border Cells



Heterogeneous Cells



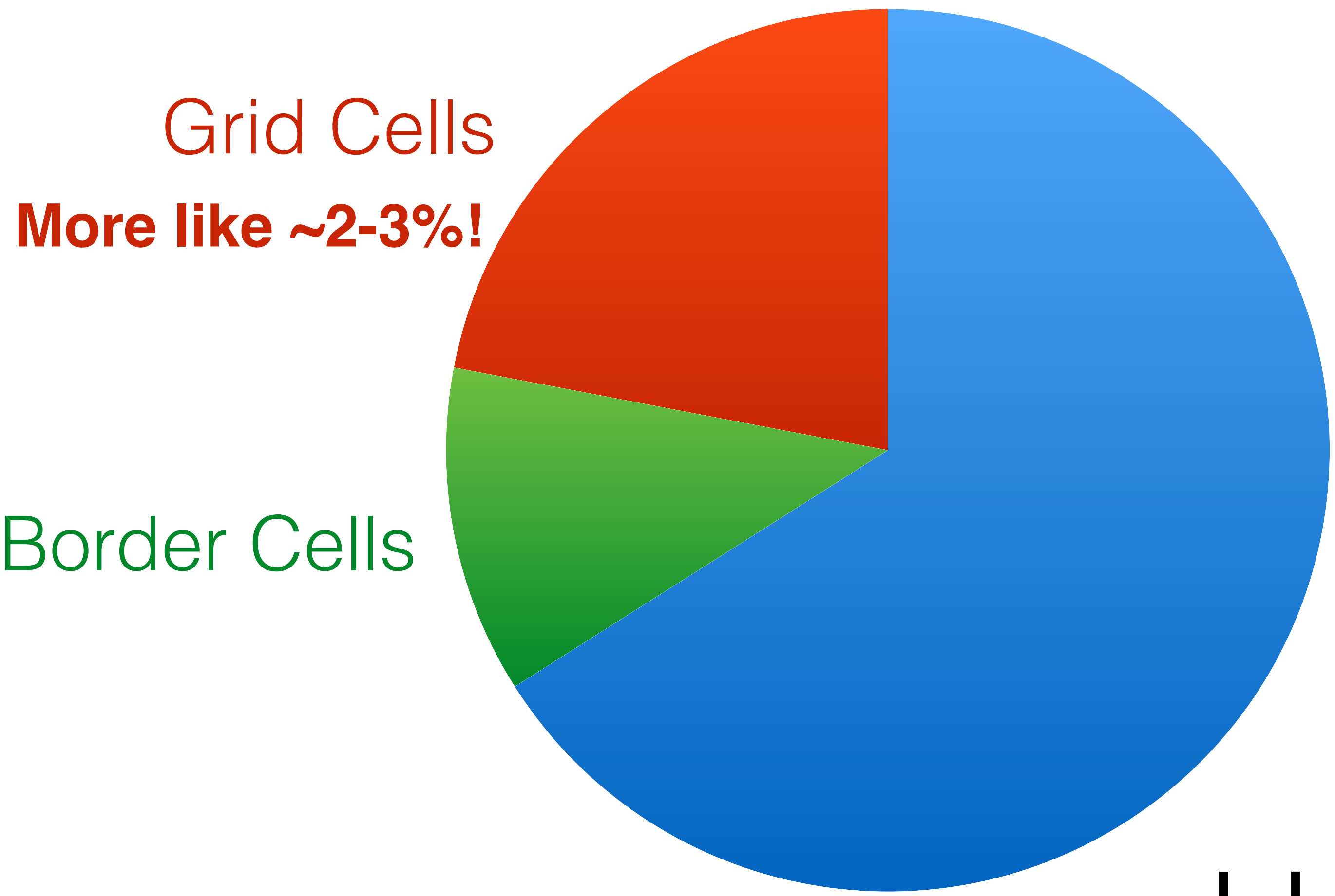
Data from: *Mallory et al. 2021*



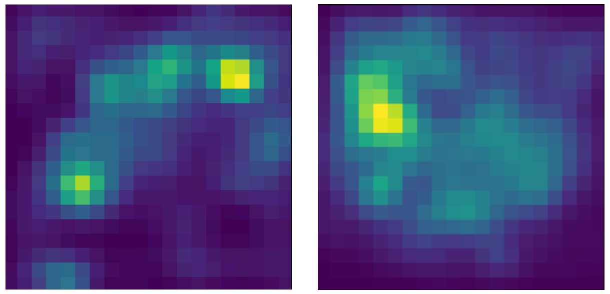
__Neurobiological Puzzle(s):__

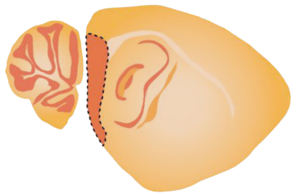
1. How might we characterize what these heterogeneous cells do?

Accounting for heterogeneous code?



Data from: *Mallory et al. 2021*



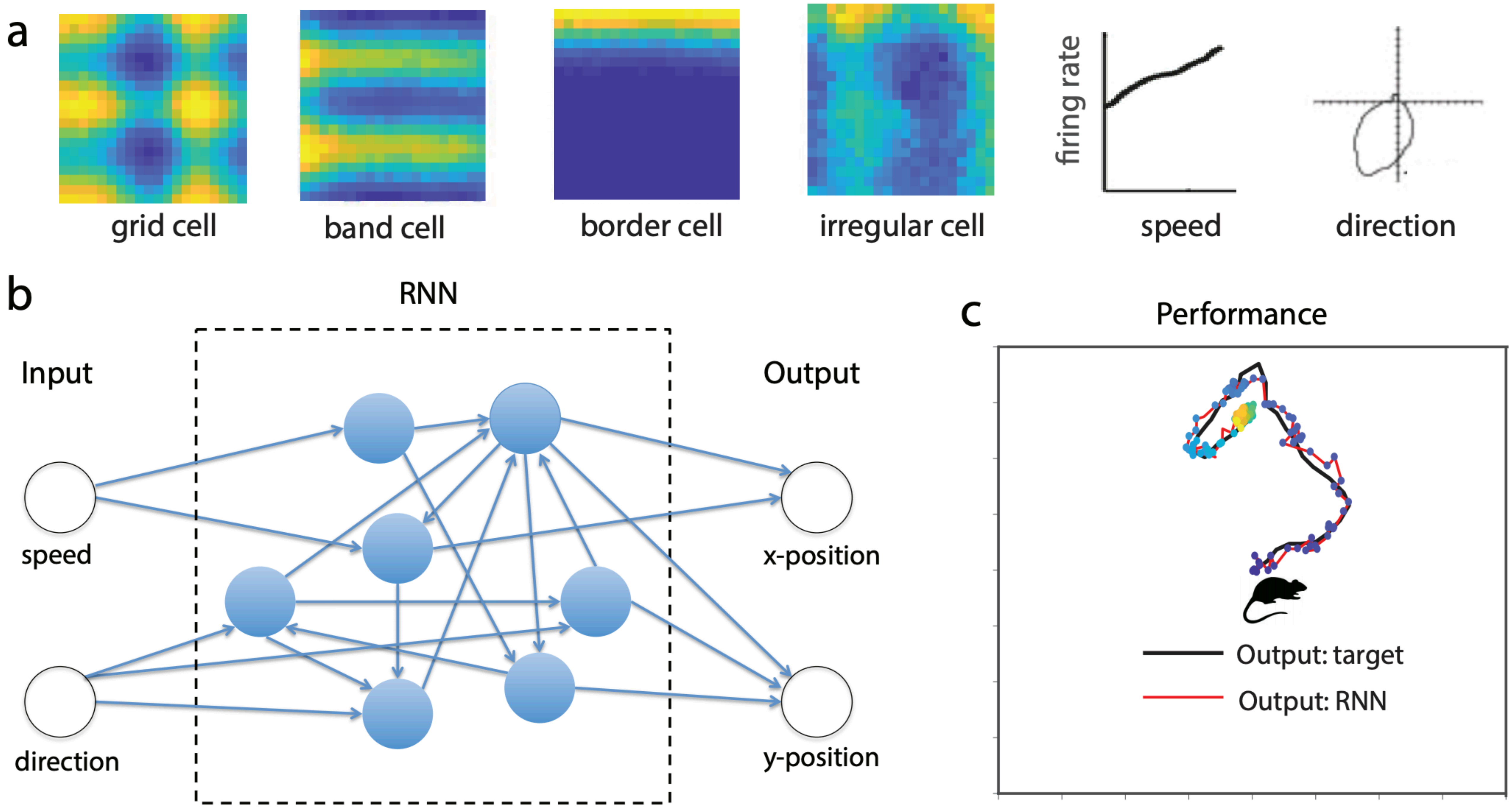


<u>Neurobiological Puzzle(s):</u>

1. How might we characterize what these heterogeneous cells do?

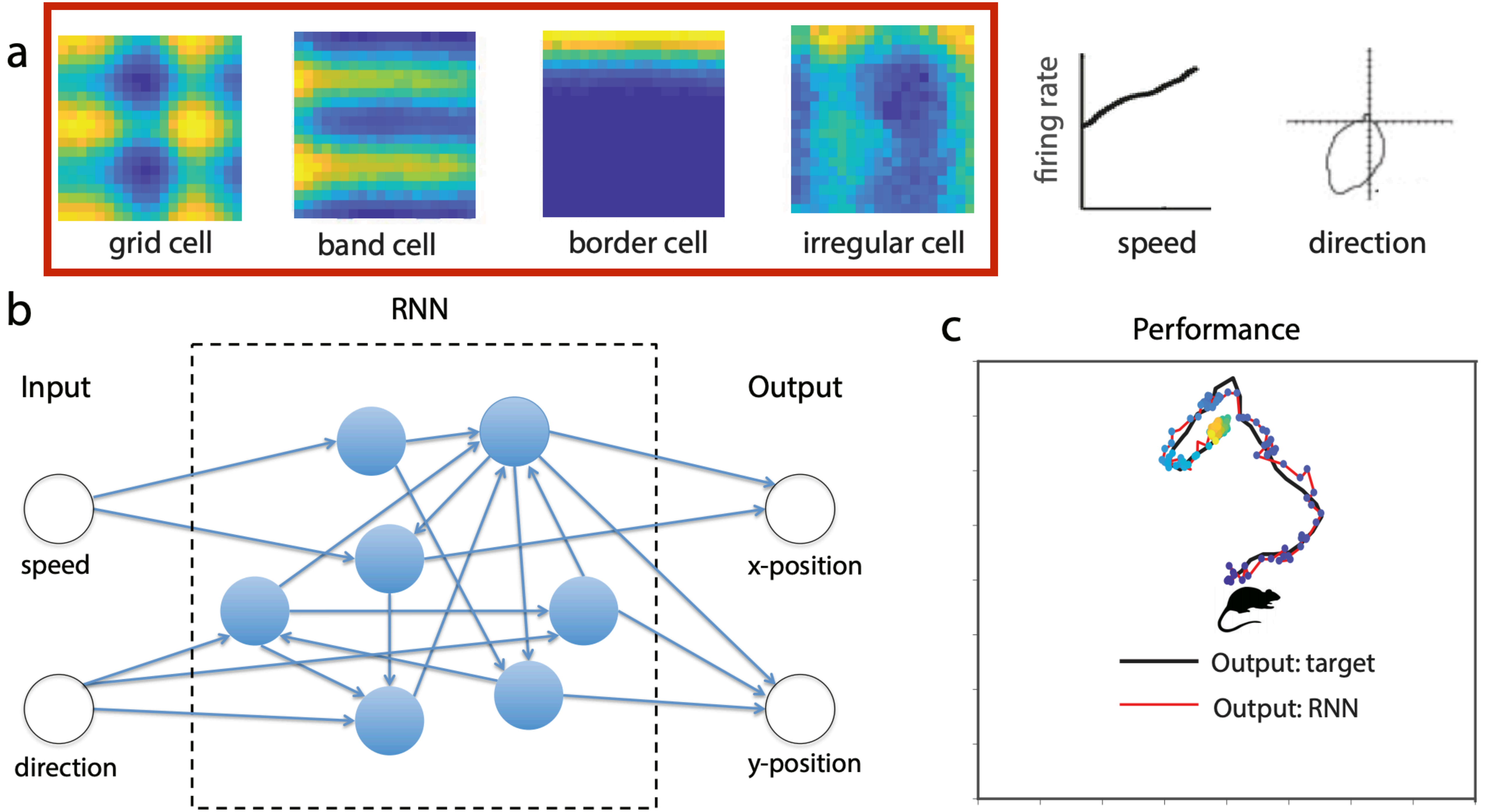
2. What functional role do these cells serve in the circuit, if any?

But more recently there are neural network models that "develop" these cells…



Cueva & Wei* 2018*

But more recently there are neural network models that "develop" these cells…

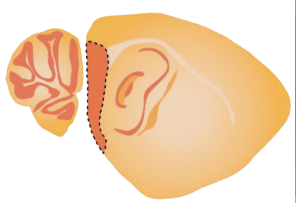


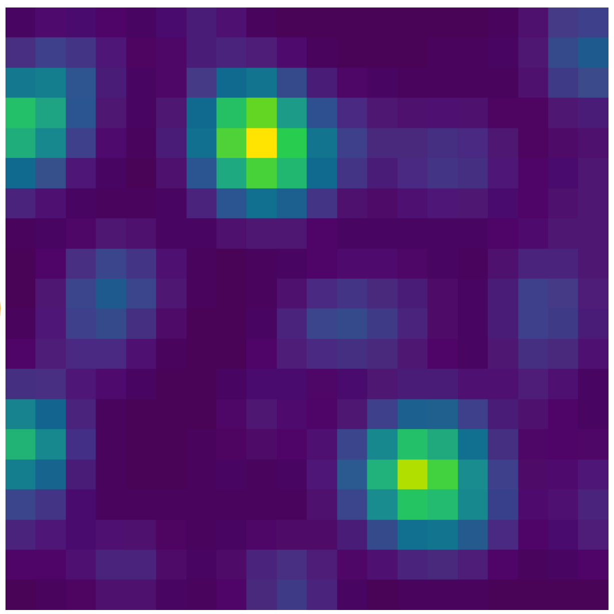
Cueva & Wei* 2018*

Goal-Driven Approach

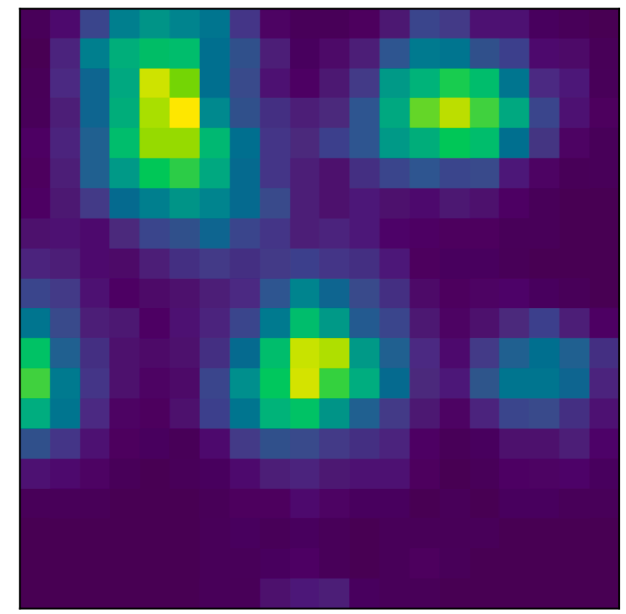
But are they a good ***quantitative*** model of these responses?

MEC Grid Cell

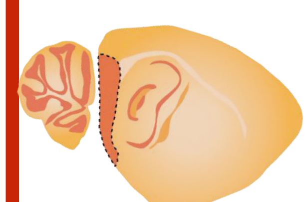


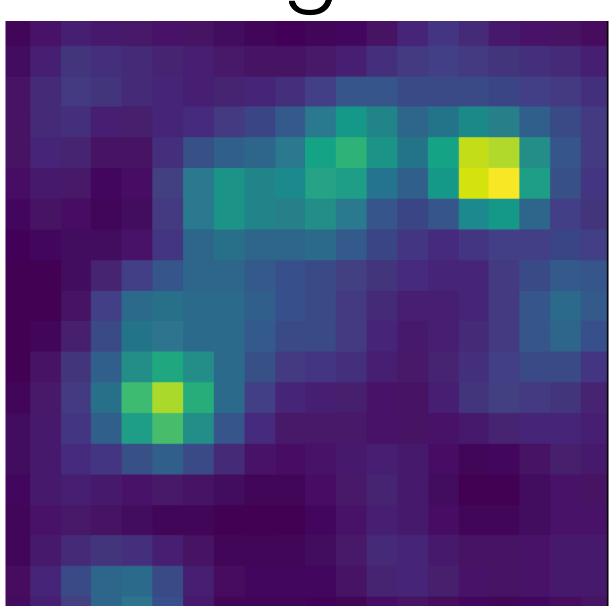


Model Grid Cell



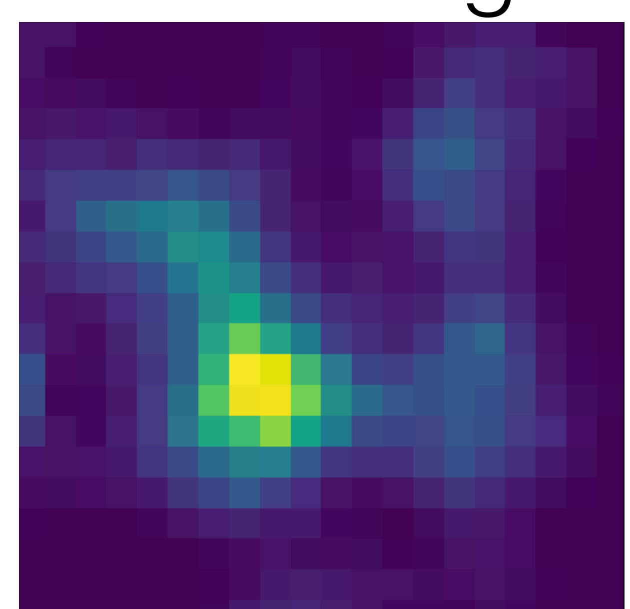
MEC Heterogeneous Cell





?

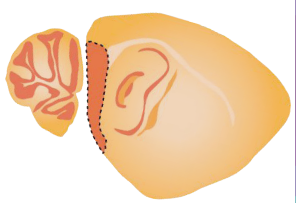
Model Heterogeneous Cell

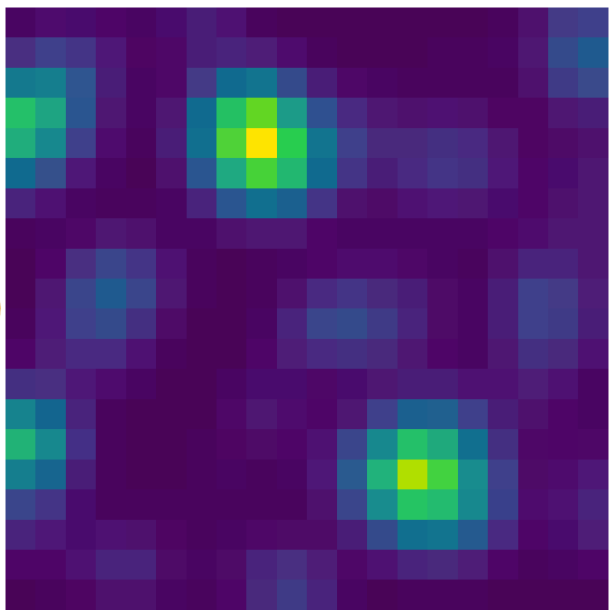


Goal-Driven Approach

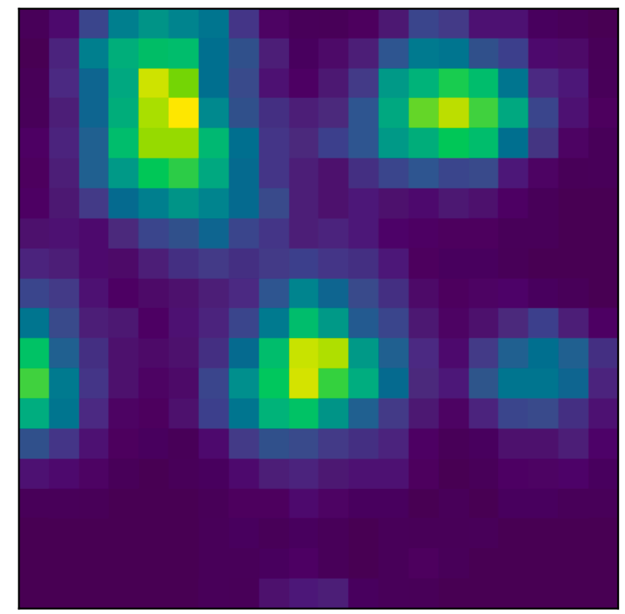
But are they a good ***quantitative*** model of these responses?

MEC Grid Cell

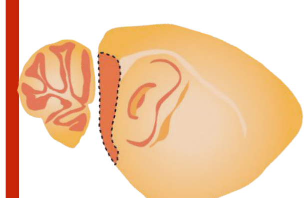


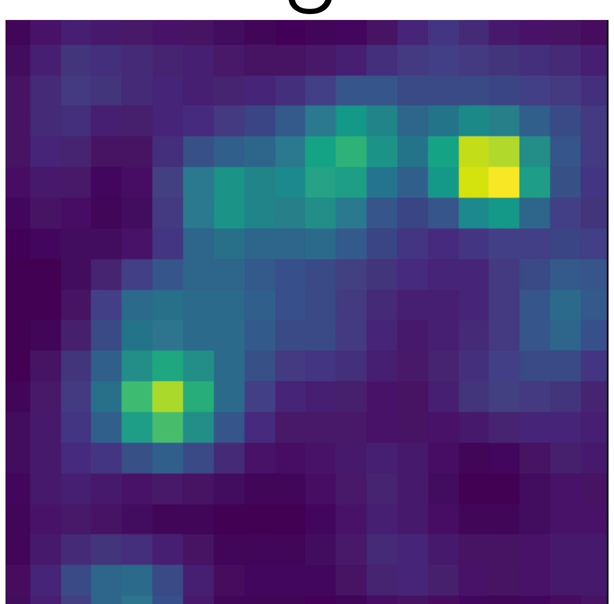


Model Grid Cell



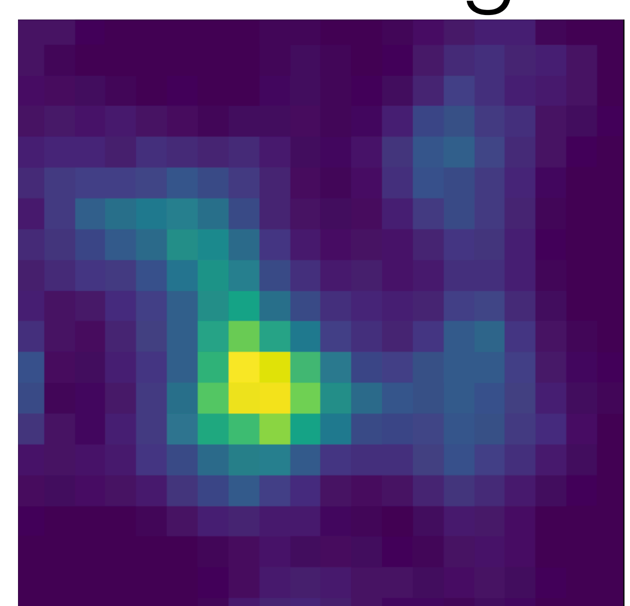
MEC Heterogeneous Cell





Not all models are equal!

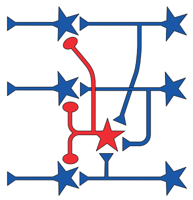
Model Heterogeneous Cell



Goal-Driven Approach

A *= architecture class*

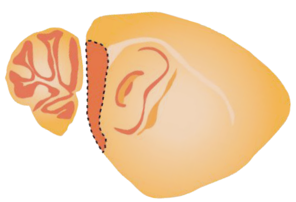
1. *"Circuit"*

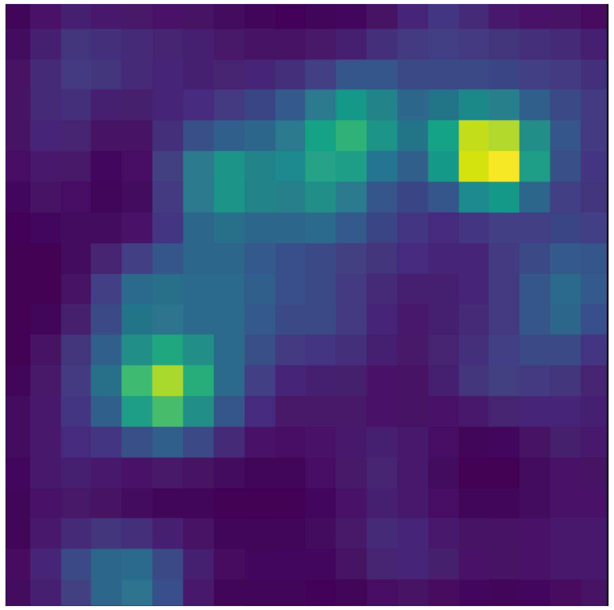


T *= task loss*

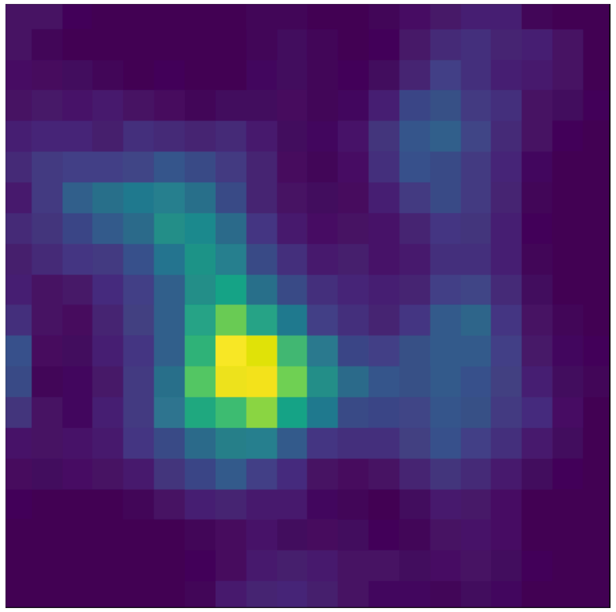
3. *"Ecological niche/behavior"*

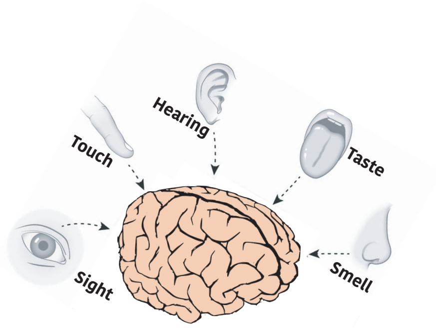
MEC Heterogeneous Cell





Model Heterogeneous Cell





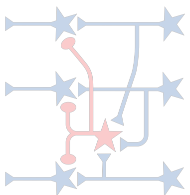
2. *"Environment"*

D *= data stream*

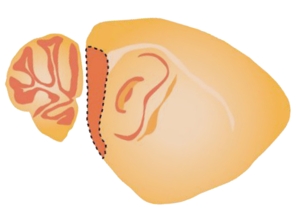
Goal-Driven Approach

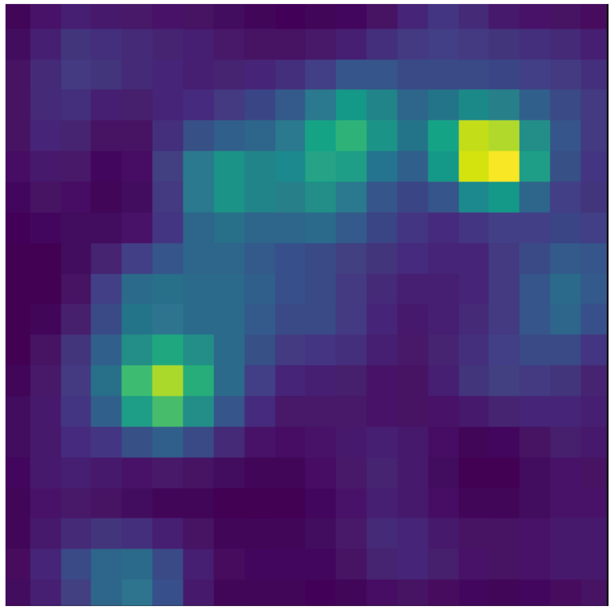
$\boldsymbol{T}$ = *task loss*

3. *"Ecological niche/behavior"*

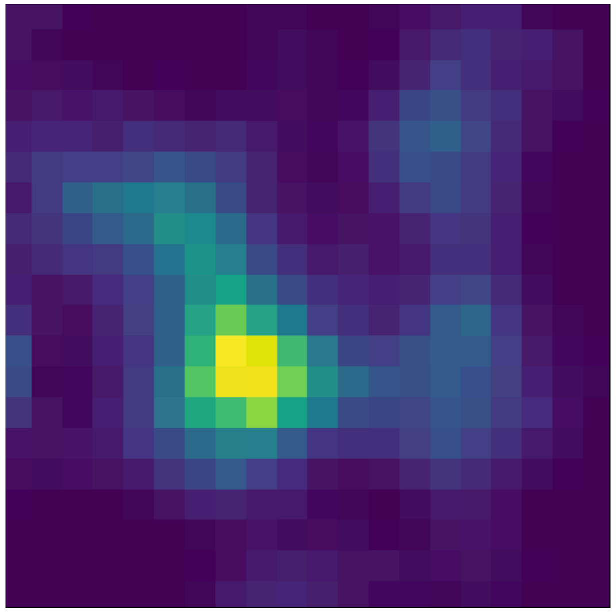


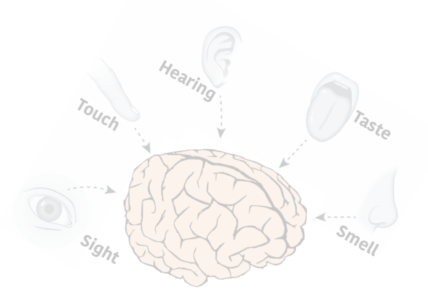
MEC Heterogeneous Cell



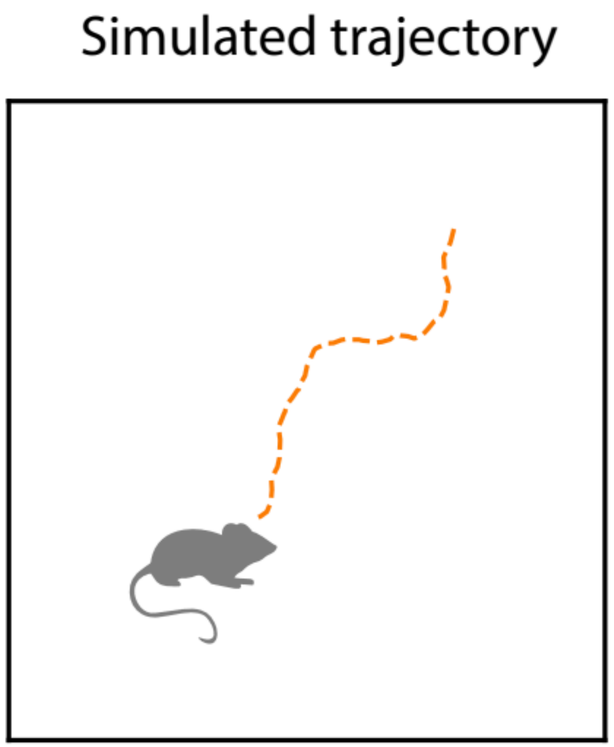


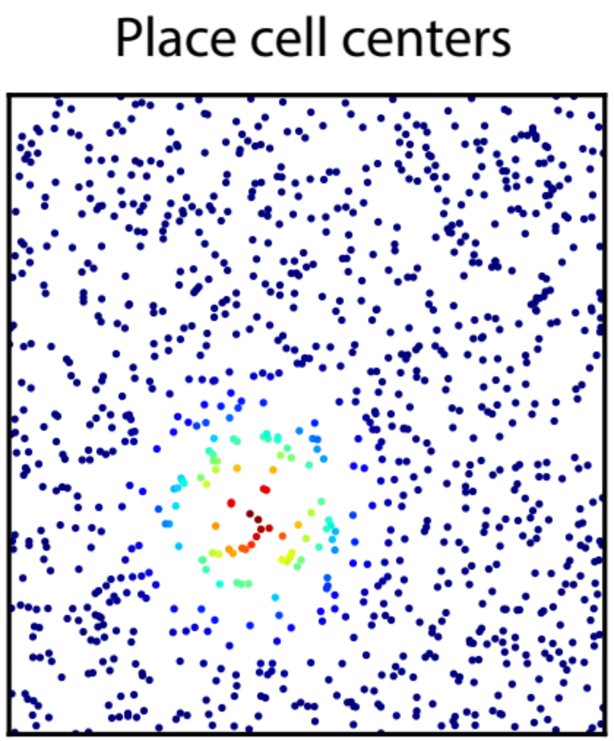
Model Heterogeneous Cell





A spectrum of tasks

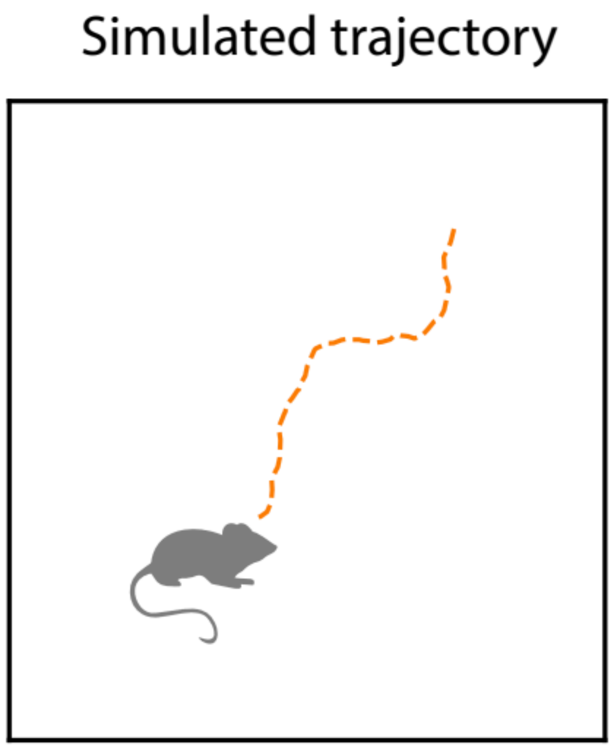


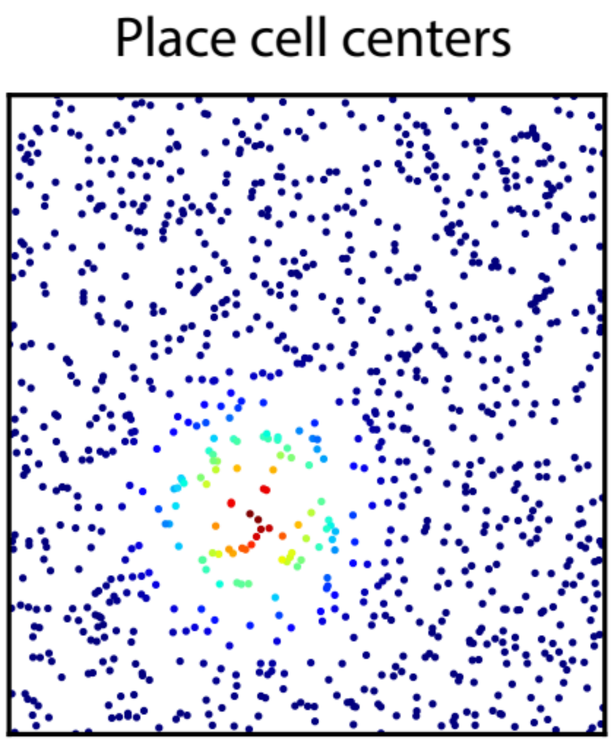


Banino, Barry* et al. 2018*

Sorscher, Mel* et al. 2019*

A spectrum of tasks





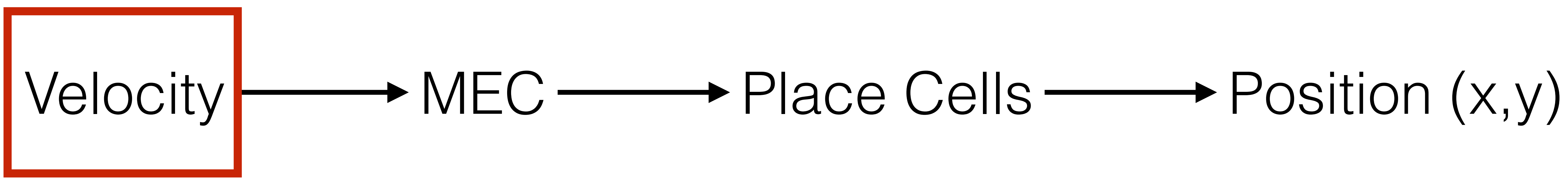
Banino, Barry* et al. 2018*

Sorscher, Mel* et al. 2019*

Velocity → MEC → Place Cells → Position (x,y)

A spectrum of tasks

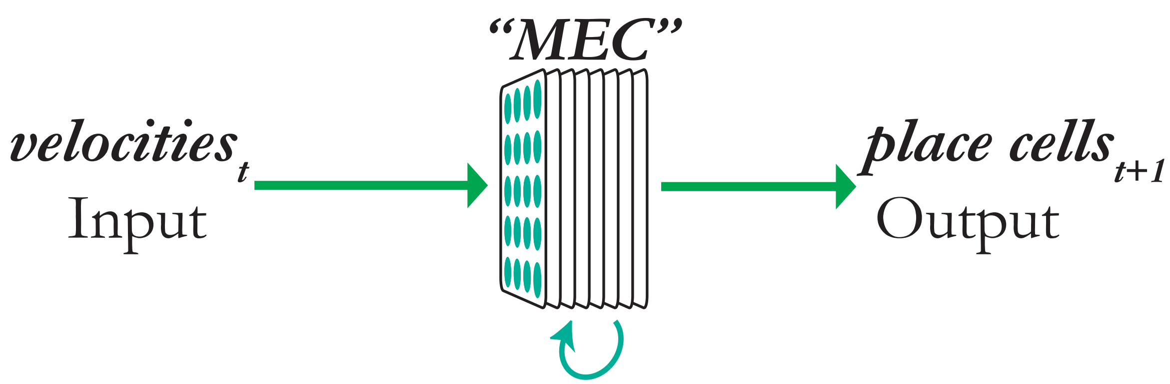
Simplest "model"

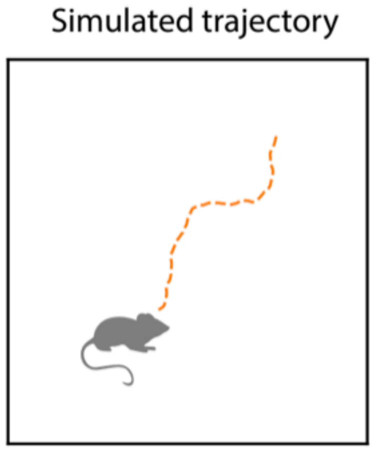


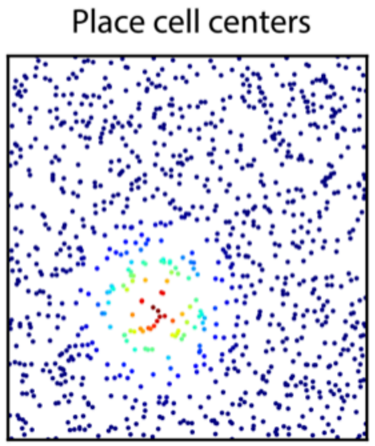
A spectrum of tasks

$$\mathcal{L}(\hat{p}, p) := -\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{N_p} p_i^t \log \hat{p}_i^t$$

Banino, Barry* et al. 2018*





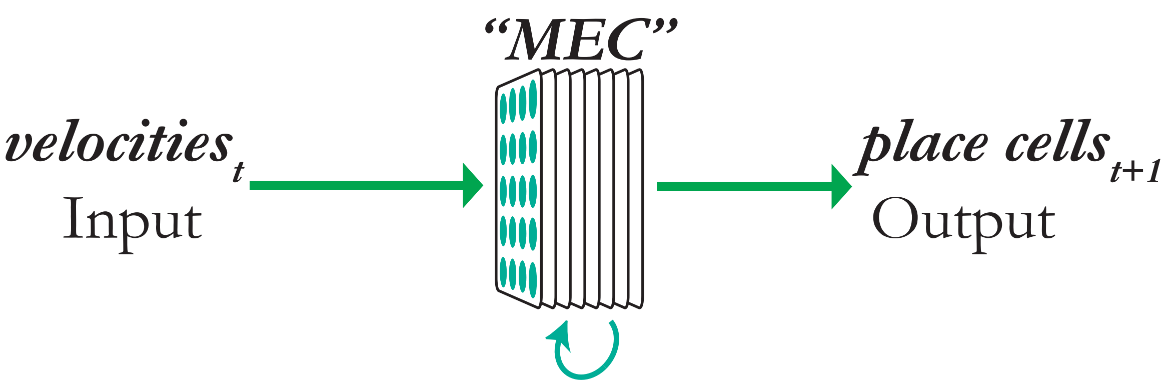


Velocity → MEC → Place Cells → Position (x,y)

A spectrum of tasks

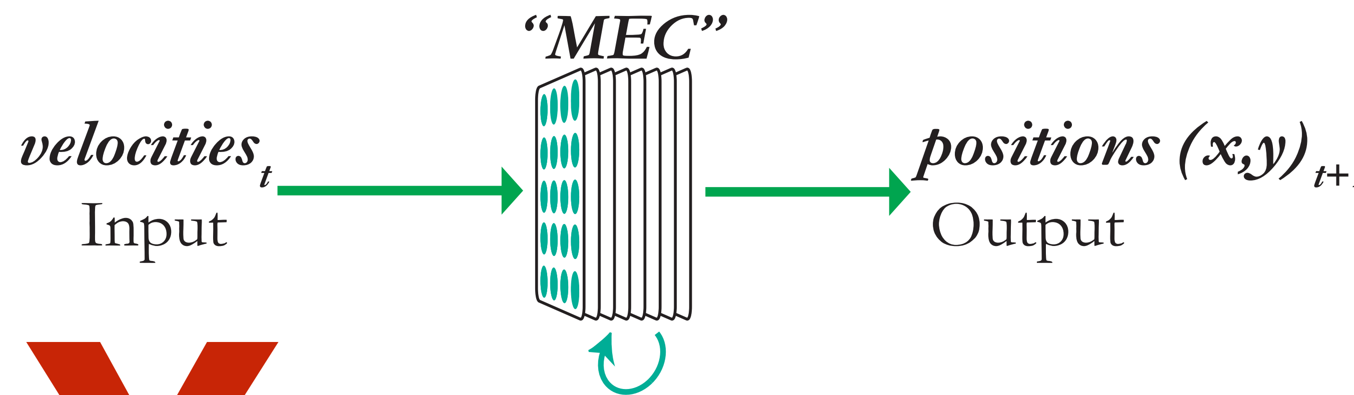
Banino, Barry* et al. 2018*

$$\mathcal{L}(\hat{p}, p) := -\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{N_p} p_i^t \log \hat{p}_i^t$$



Cueva & Wei* 2018*

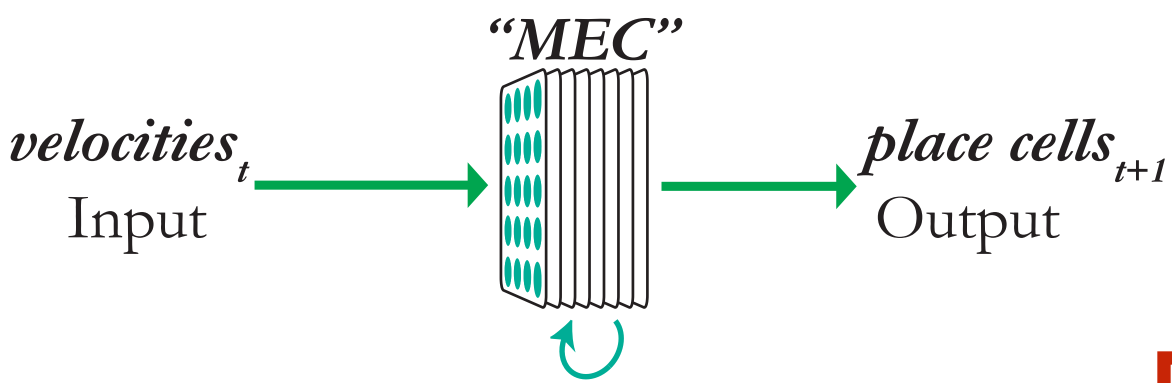
$$\mathcal{L}(\hat{p}, p) := \frac{1}{2}\frac{1}{T}\sum_{t=1}^{T}\left(\left(p_x^t - \hat{p}_x^t\right)^2 + \left(p_y^t - \hat{p}_y^t\right)^2\right)$$



A spectrum of tasks

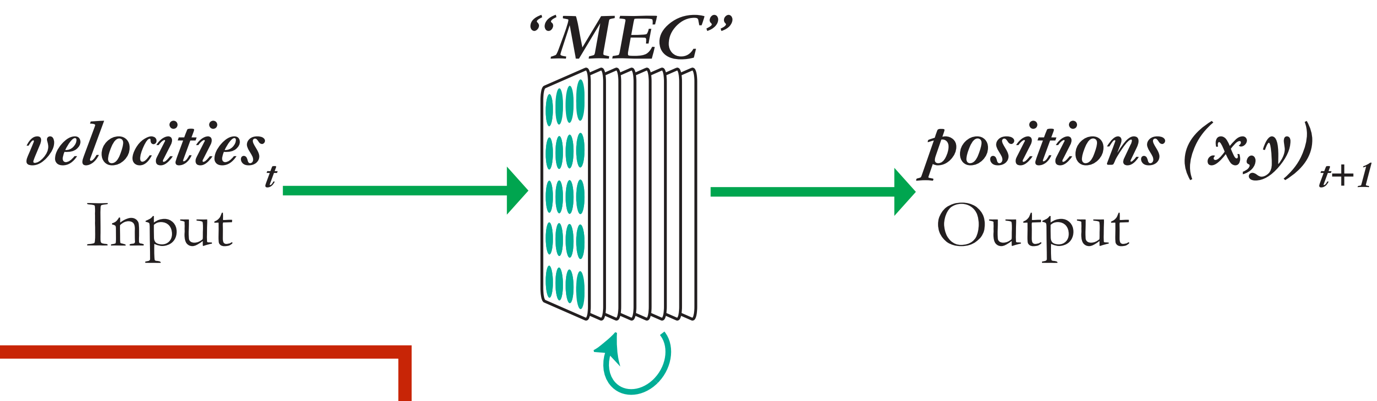
$$\mathcal{L}(\hat{p}, p) := -\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{N_p} p_i^t \log \hat{p}_i^t$$

Banino, Barry* et al. 2018*



Cueva & Wei* 2018*

$$\mathcal{L}(\hat{p}, p) := \frac{1}{2}\frac{1}{T}\sum_{t=1}^{T}\left(\left(p_x^t - \hat{p}_x^t\right)^2 + \left(p_y^t - \hat{p}_y^t\right)^2\right)$$



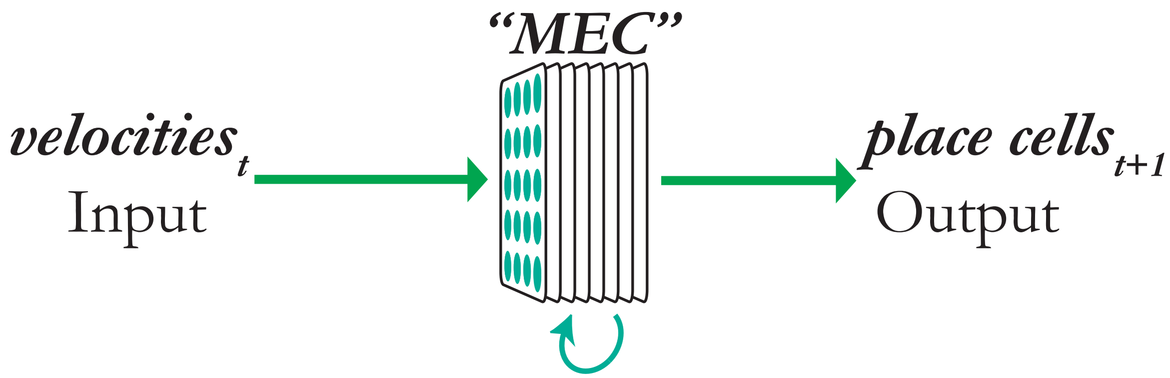
Velocity → MEC → **Place Cells** → Position (x,y)

Output-based models

A spectrum of tasks

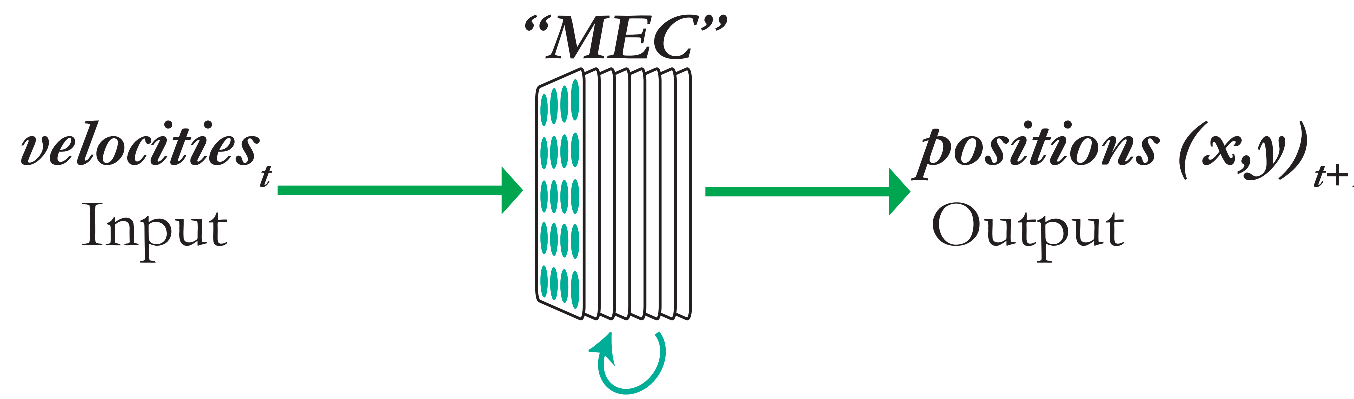
Banino, Barry* et al. 2018*

$$\mathcal{L}(\hat{p}, p) := -\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{N_p} p_i^t \log \hat{p}_i^t$$



Cueva & Wei* 2018*

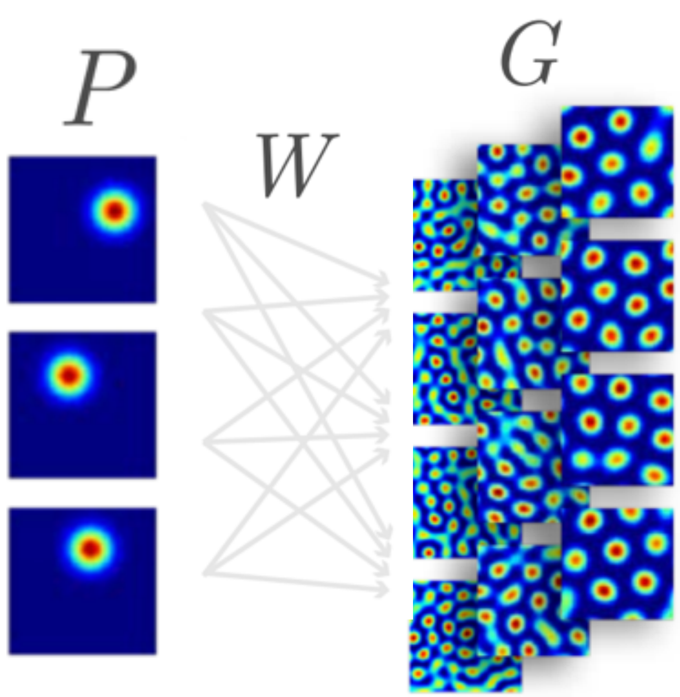
$$\mathcal{L}(\hat{p}, p) := \frac{1}{2}\frac{1}{T}\sum_{t=1}^{T}\left(\left(p_x^t - \hat{p}_x^t\right)^2 + \left(p_y^t - \hat{p}_y^t\right)^2\right)$$



Velocity → MEC → Place Cells → Position (x,y)

NMF

(Place Cell Input)

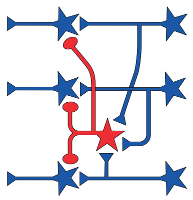


Dordek et al. 2016

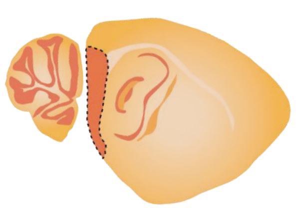
Goal-Driven Approach

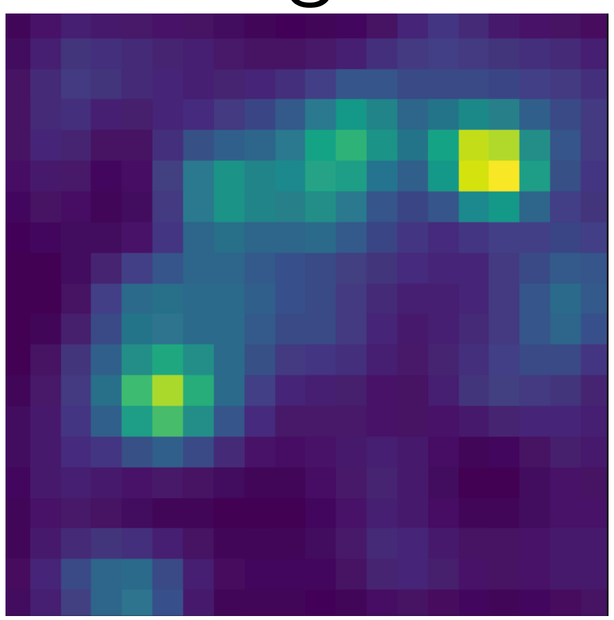
A = *architecture class*

1. *"Circuit"*

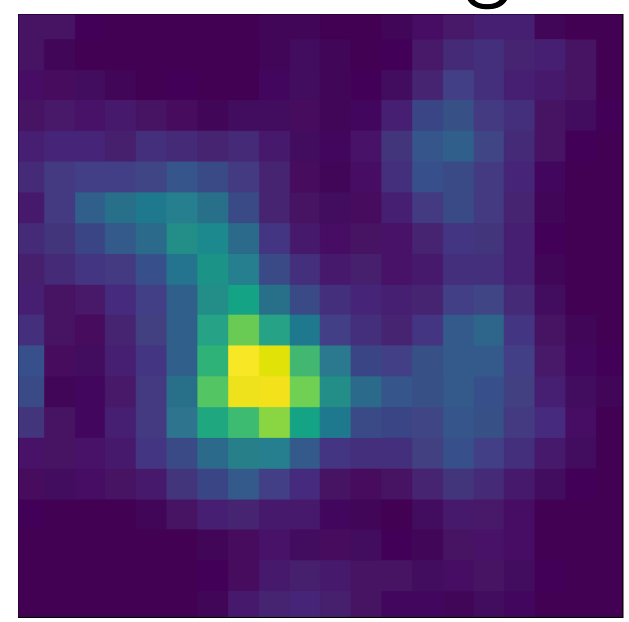


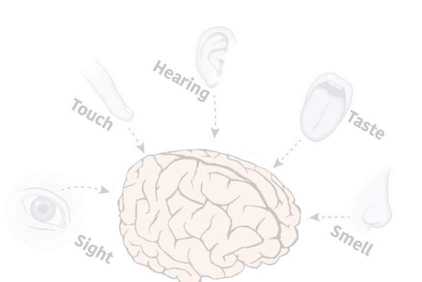
MEC Heterogeneous Cell



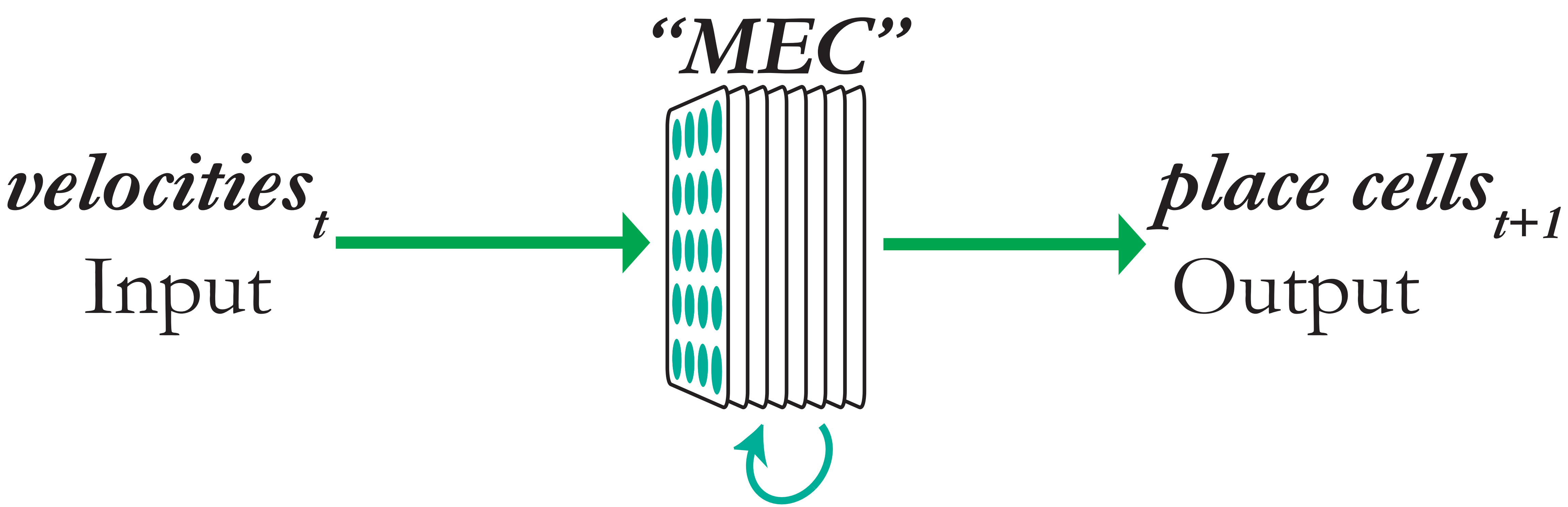


Model Heterogeneous Cell

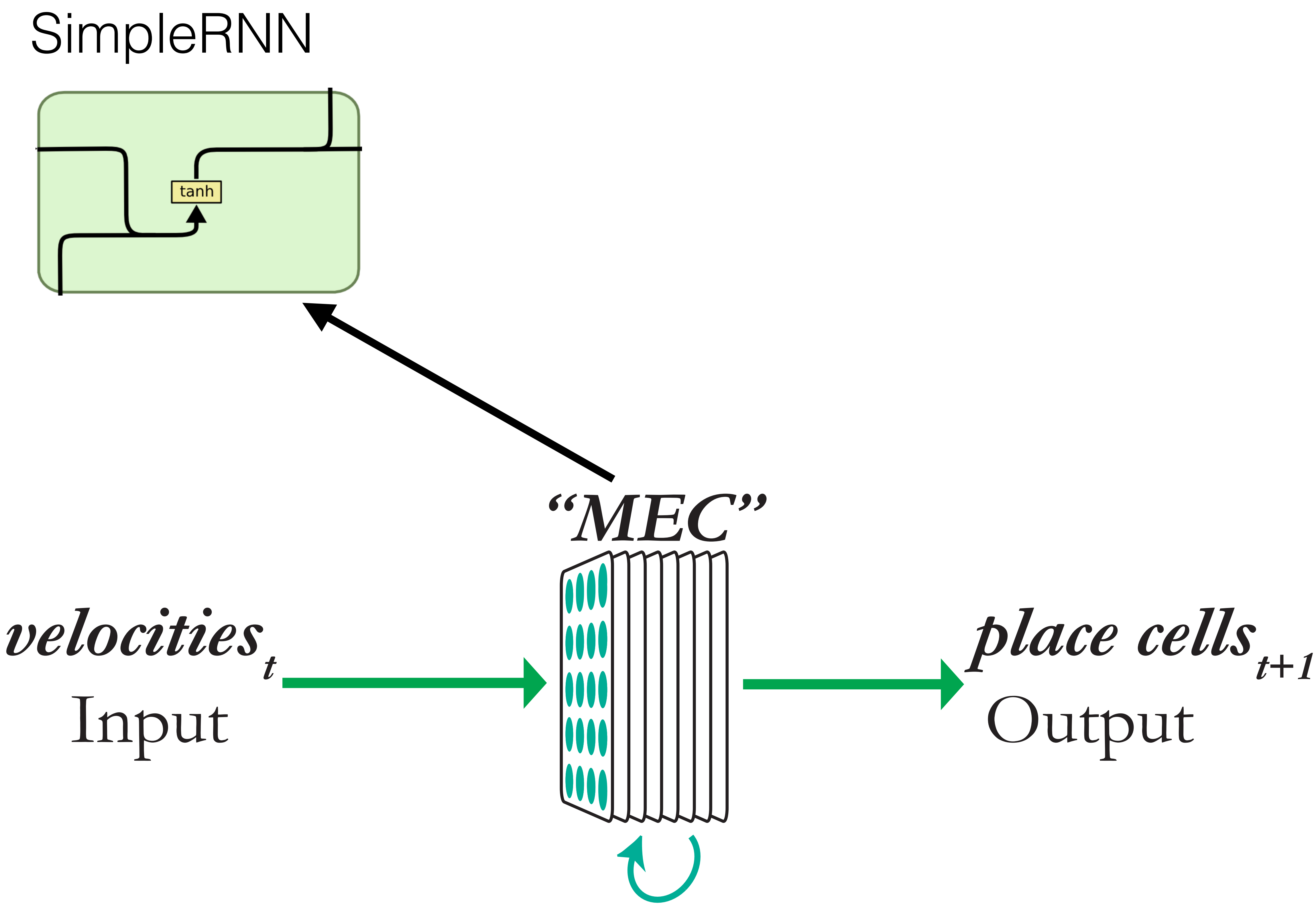




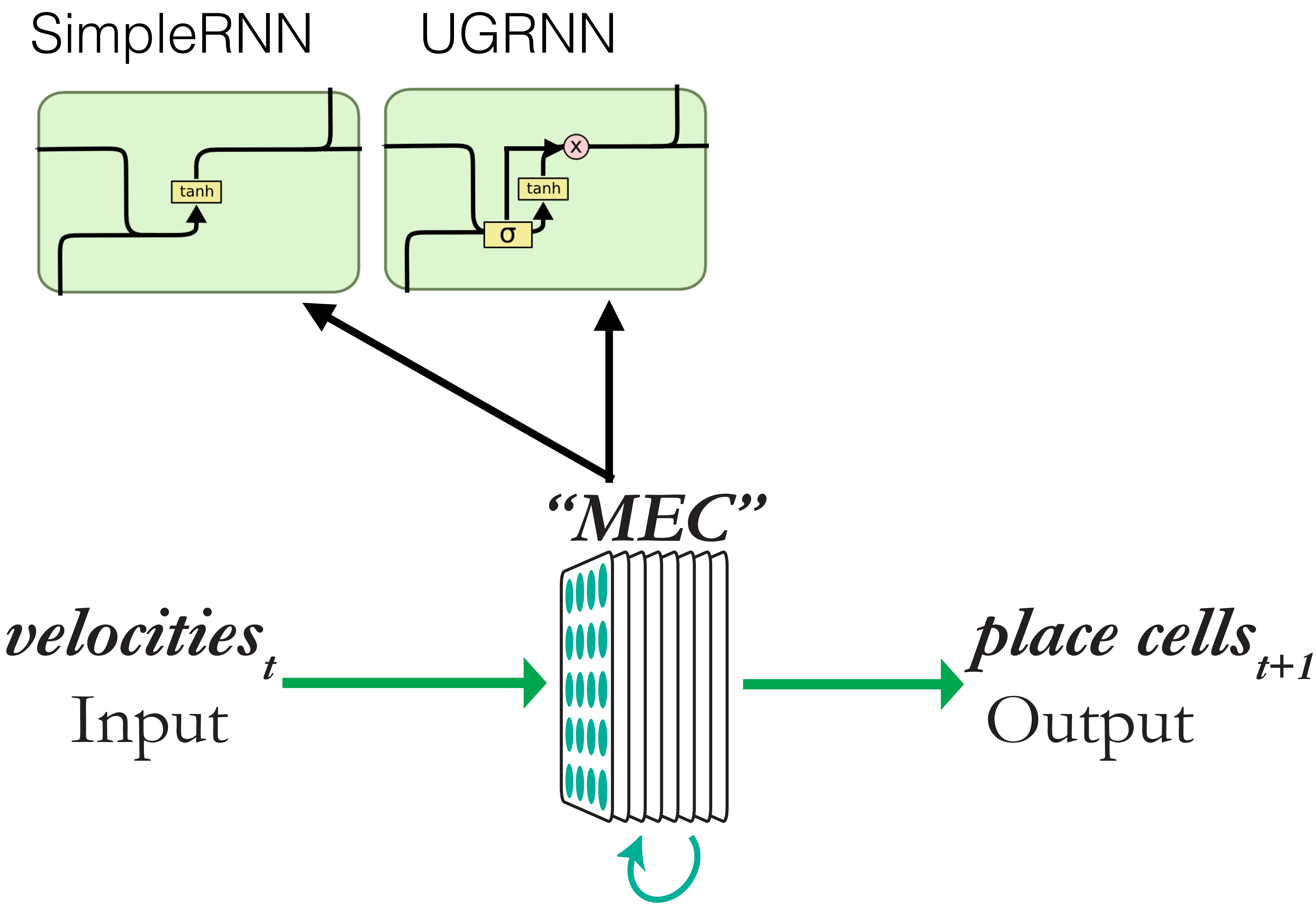
A spectrum of circuits



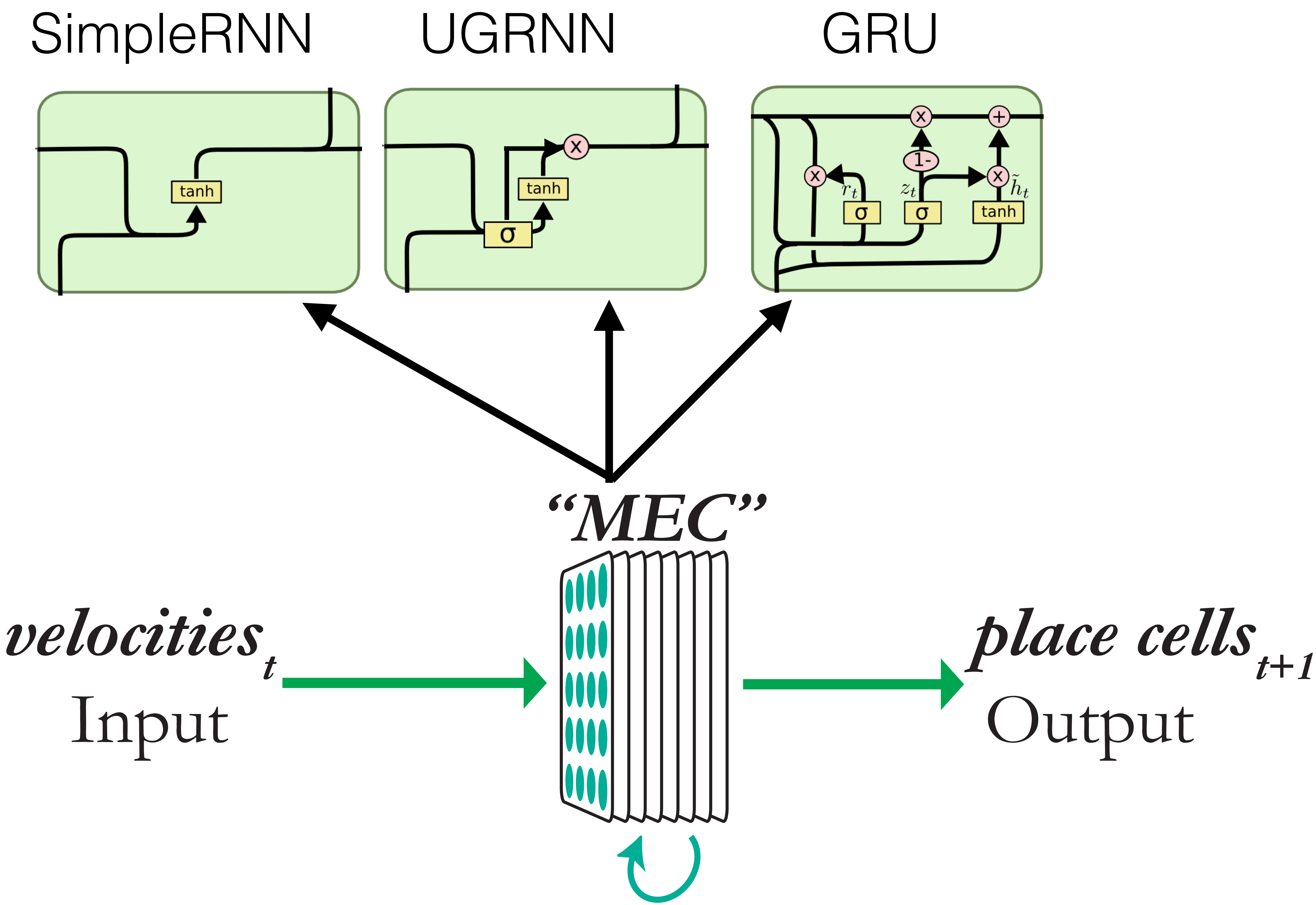
A spectrum of circuits



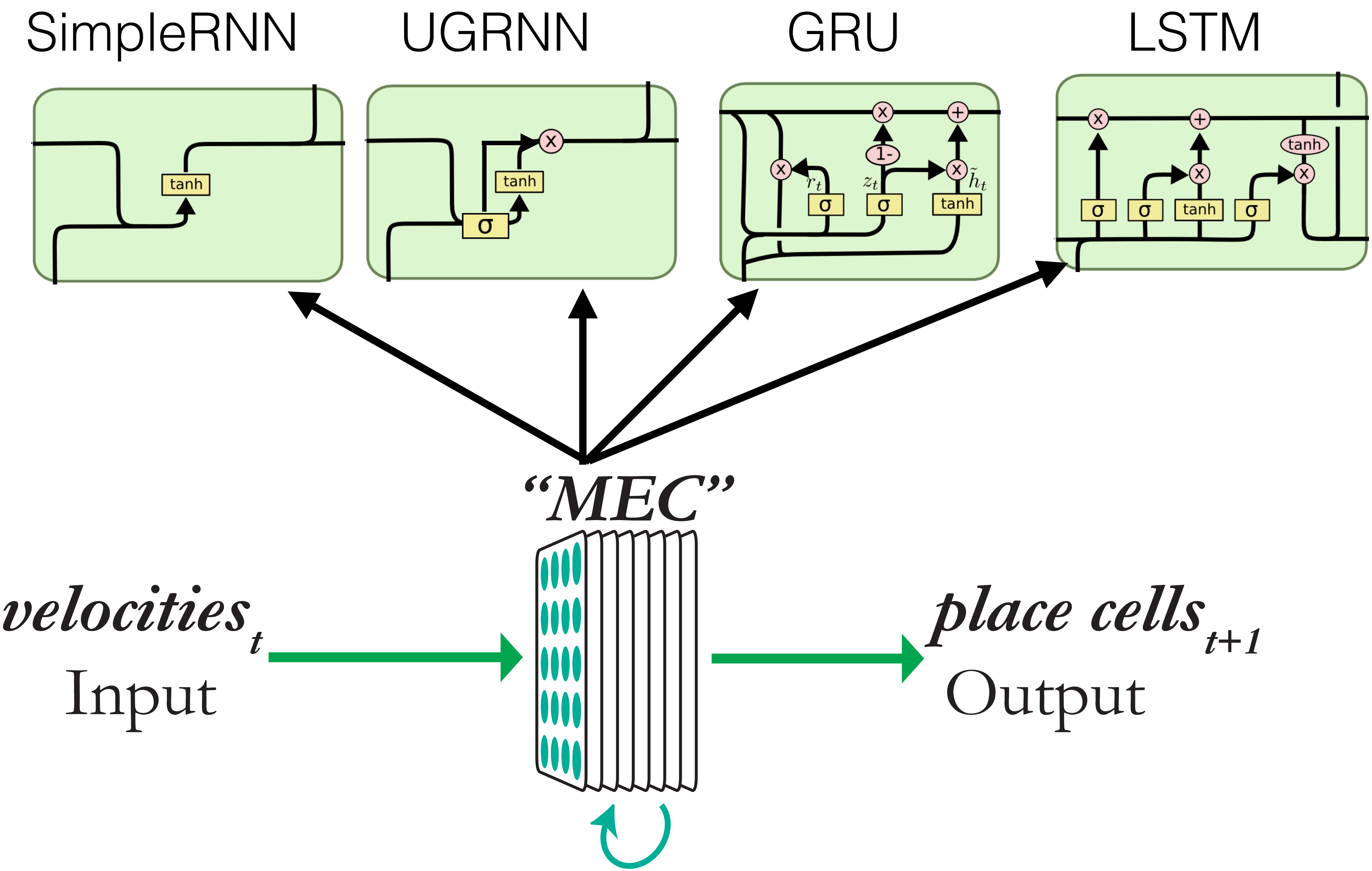
A spectrum of circuits — learnable modulation (“gating”)



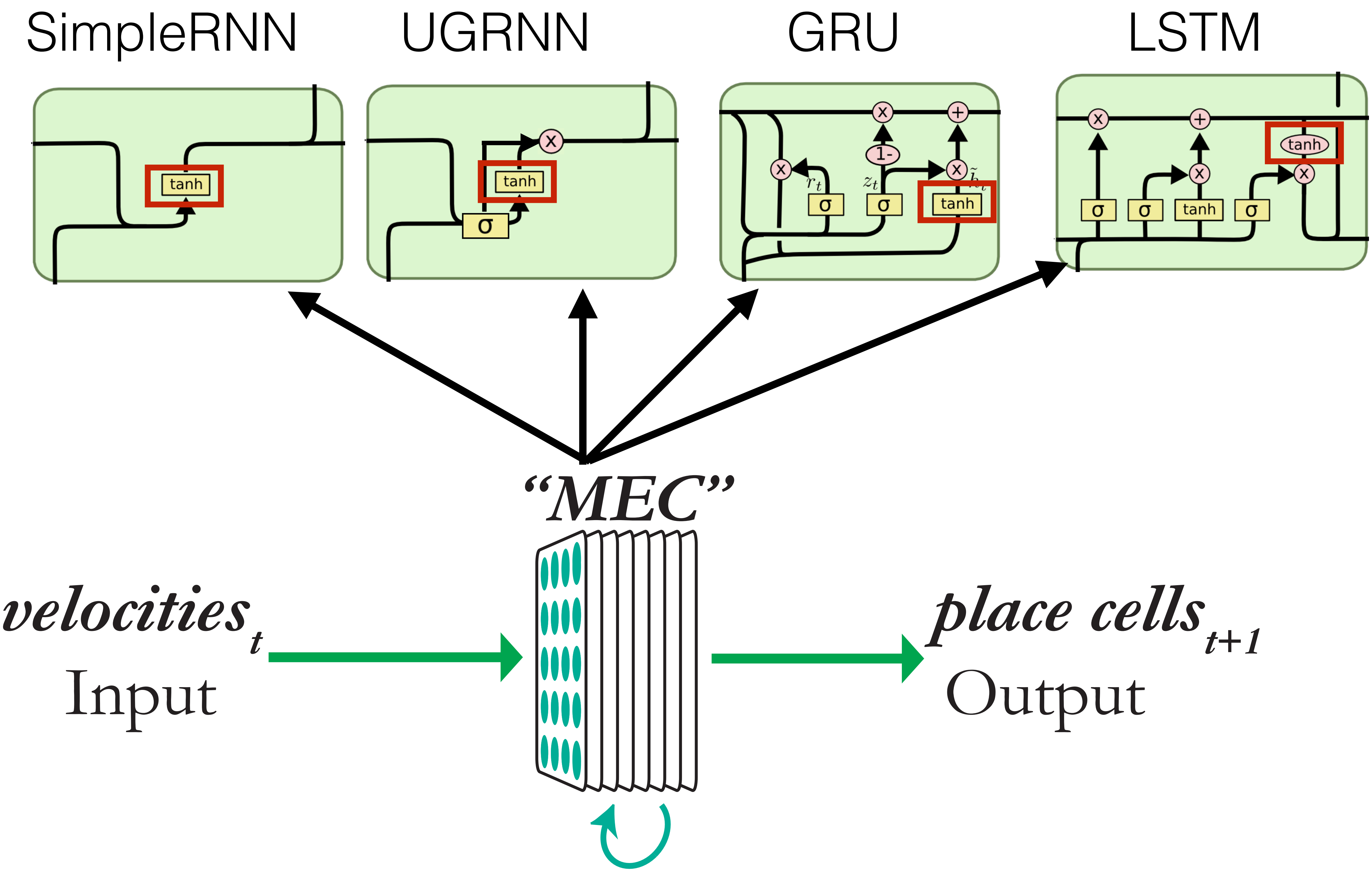
A spectrum of circuits — learnable modulation (“gating”)



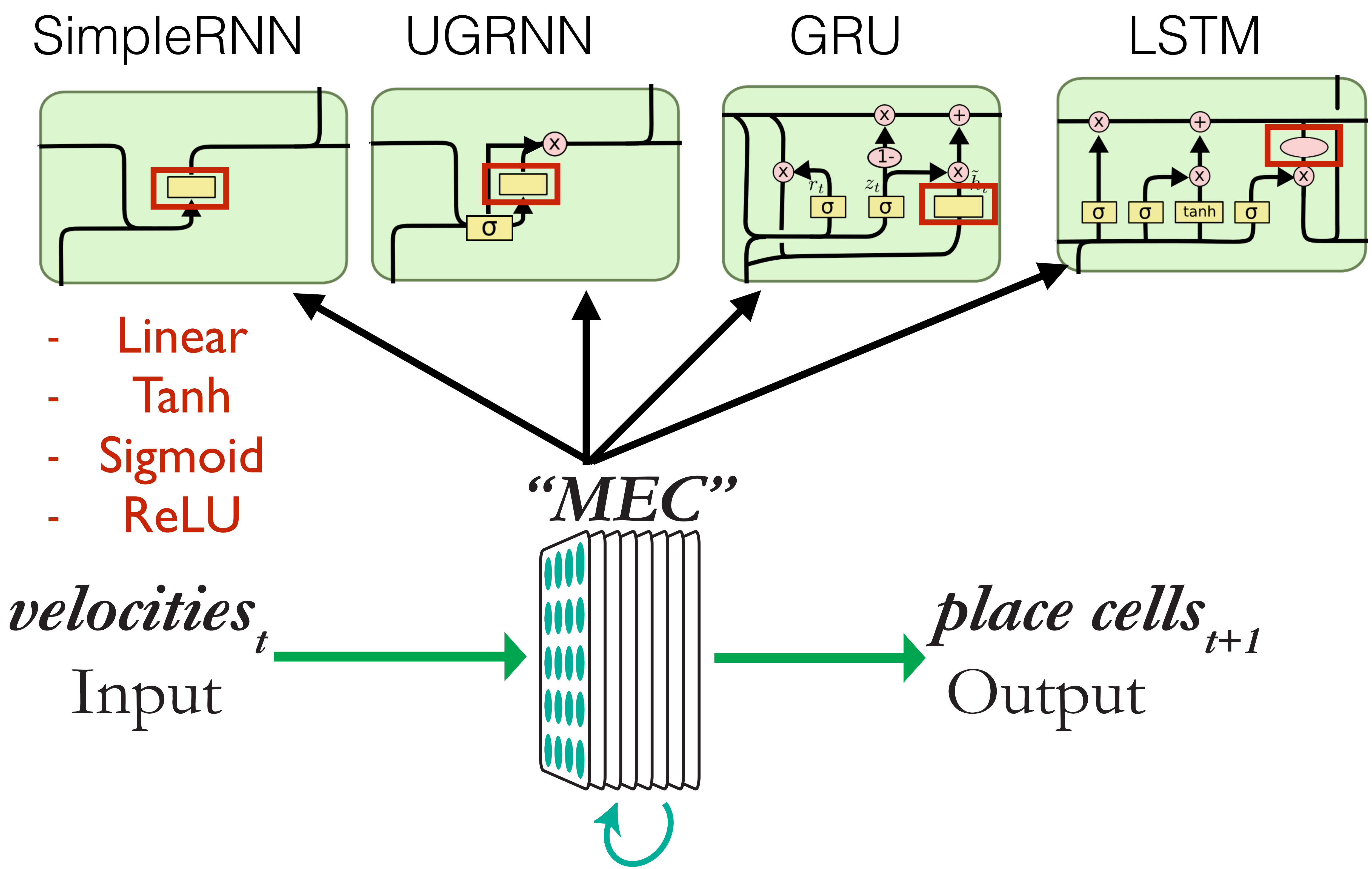
A spectrum of circuits — learnable modulation ("gating")



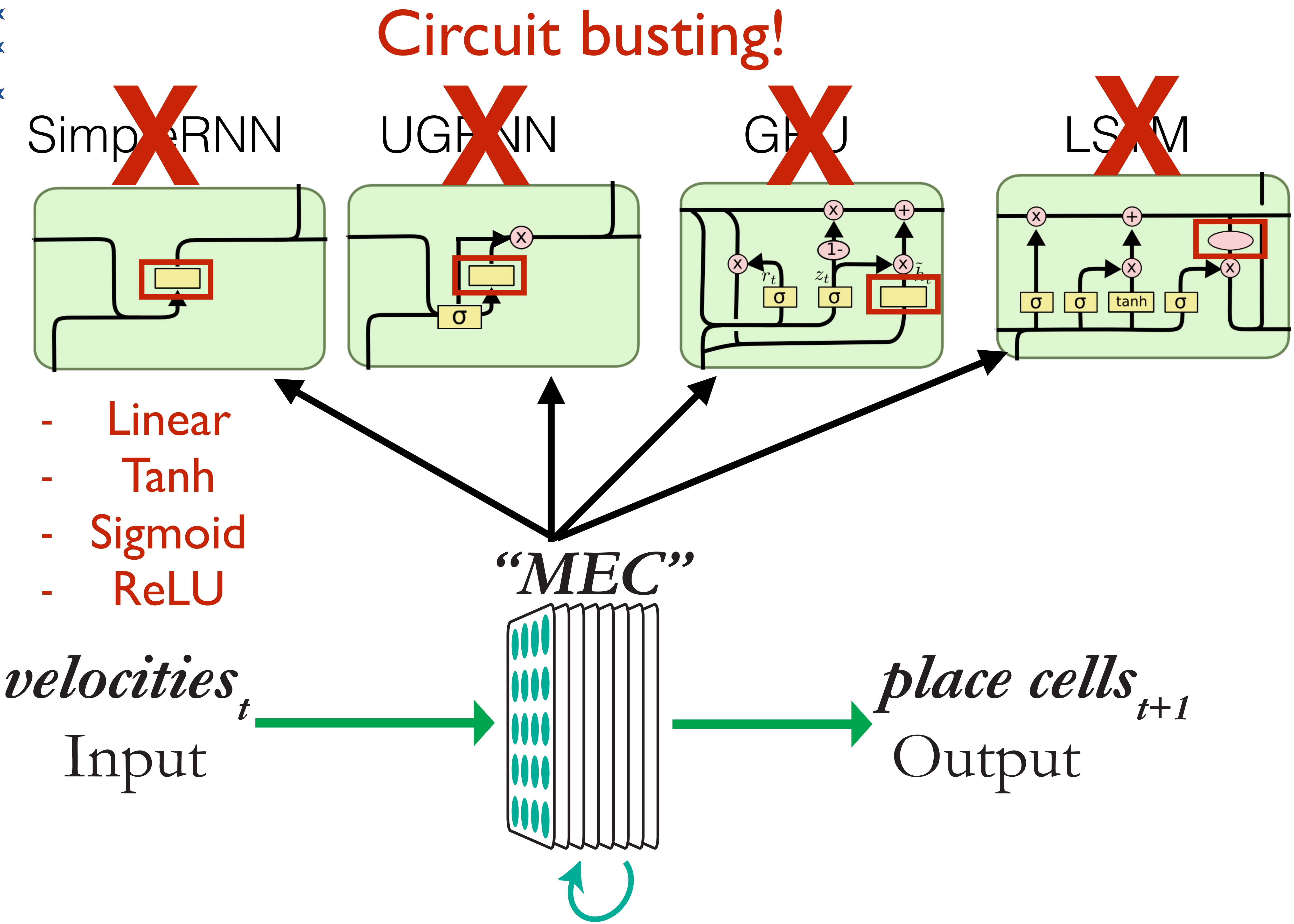
A spectrum of circuits — output nonlinearity



A spectrum of circuits — output nonlinearity

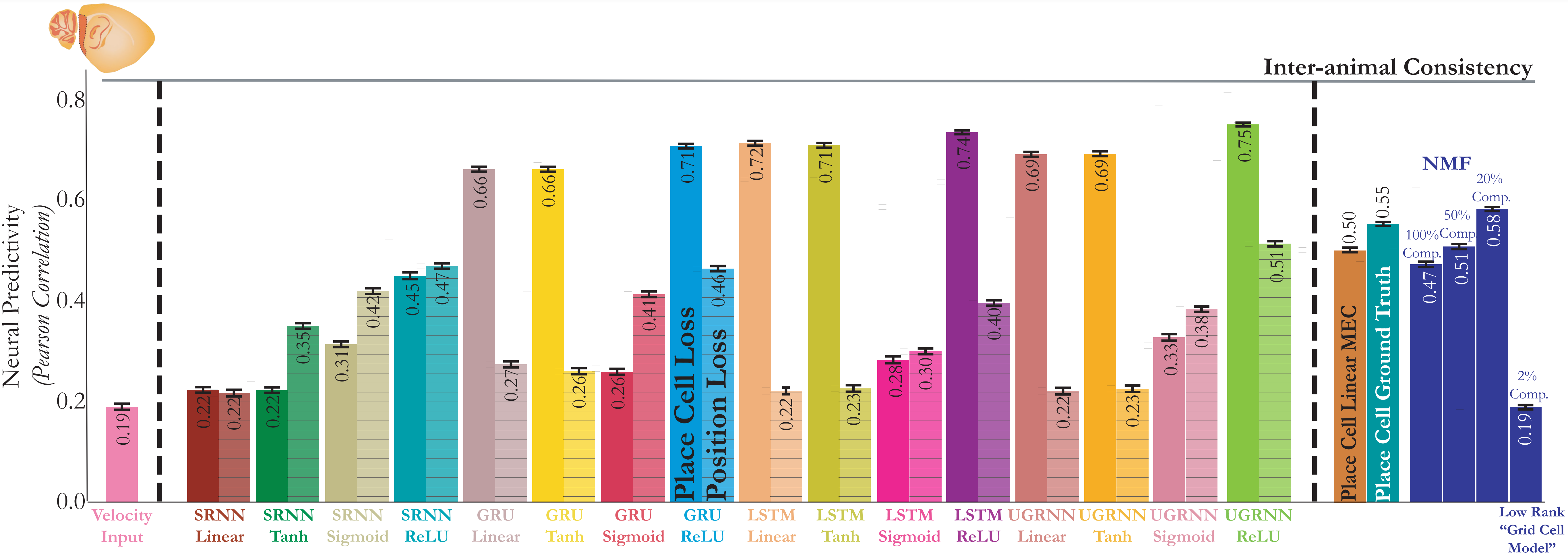


A spectrum of circuits — output nonlinearity



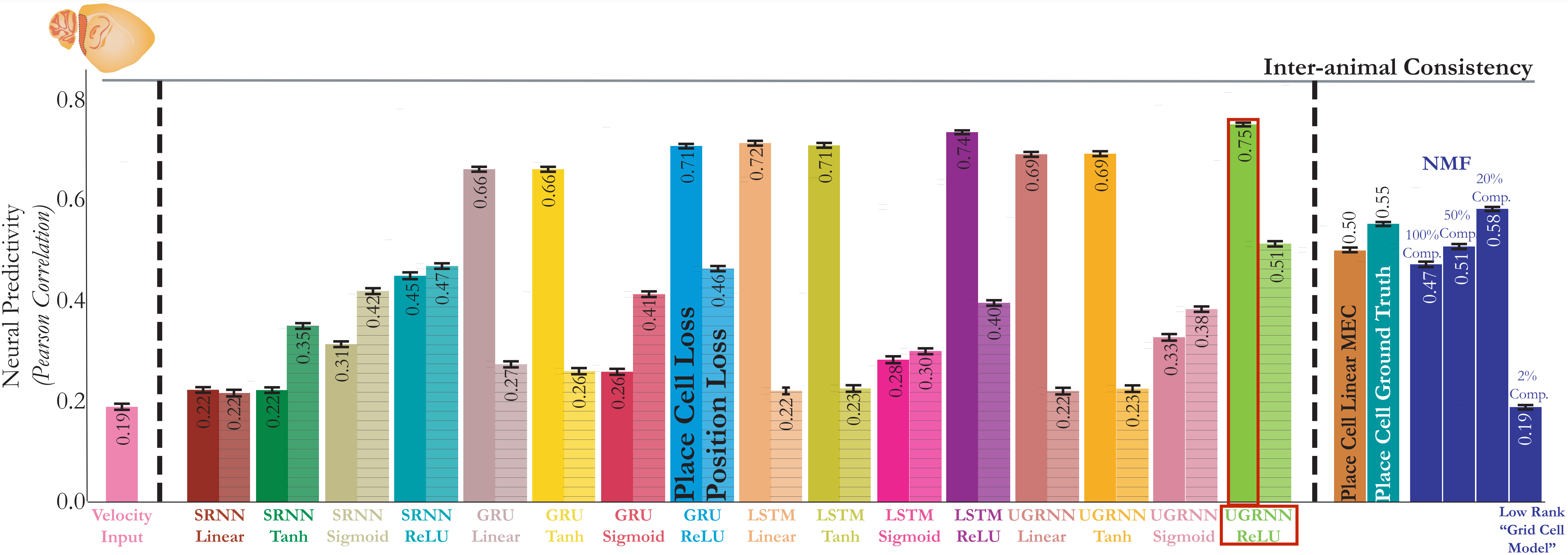
- Linear
- Tanh
- Sigmoid
- ReLU

Benchmarking models with the same transform as between animals

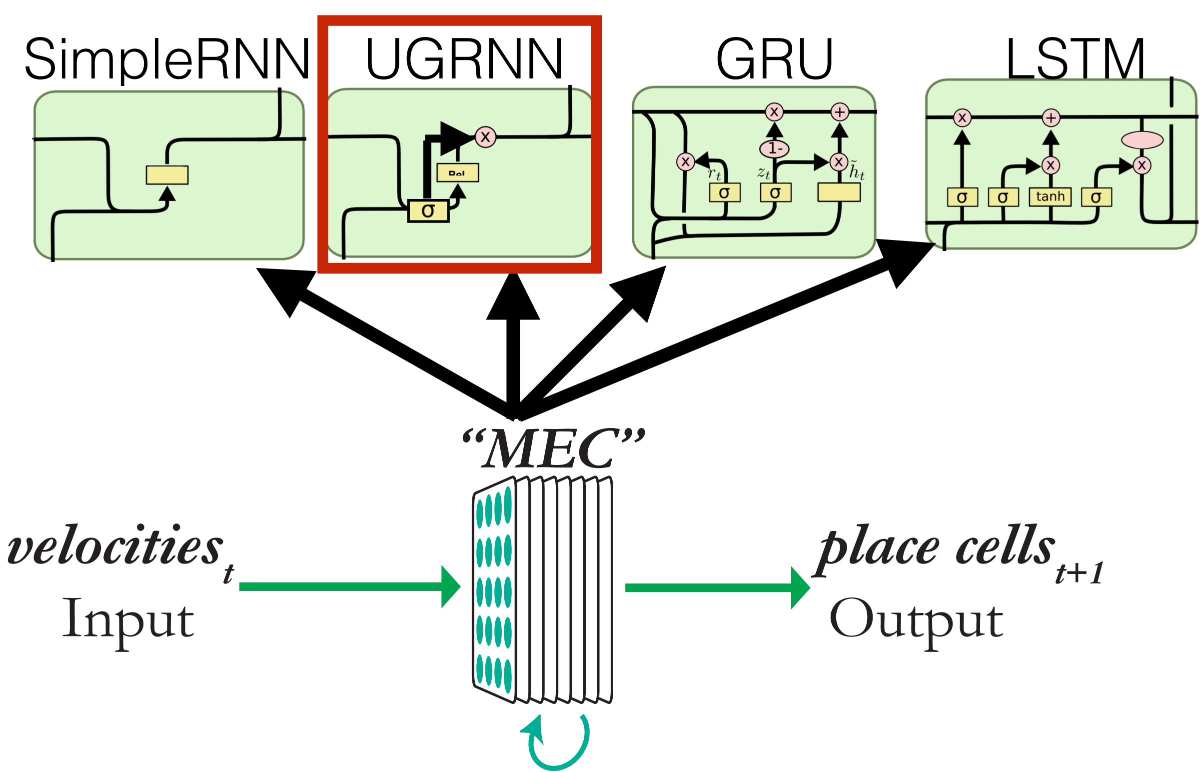


Caitlin Mallory

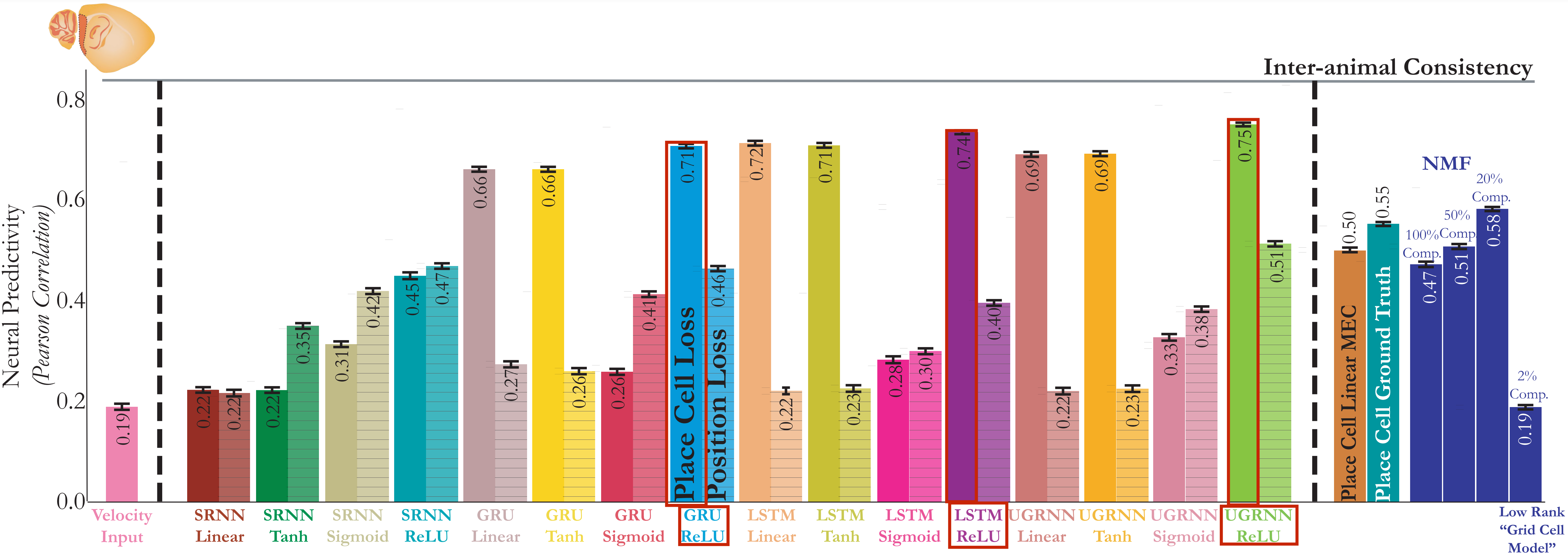
Task-optimized navigational models best predict the *entire* MEC population



Best task-optimized models explain almost all of the neural variability

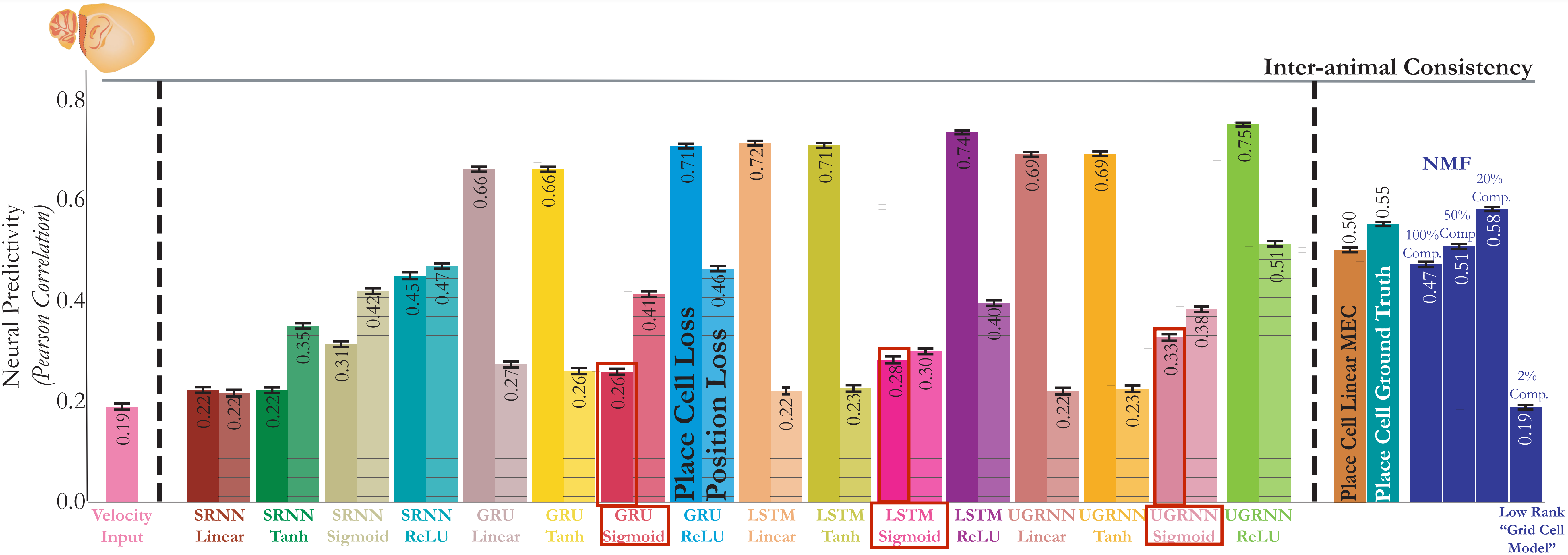


Nonlinearity type affects generalization



Nonnegativity constraint + gating aids in generalization across environments

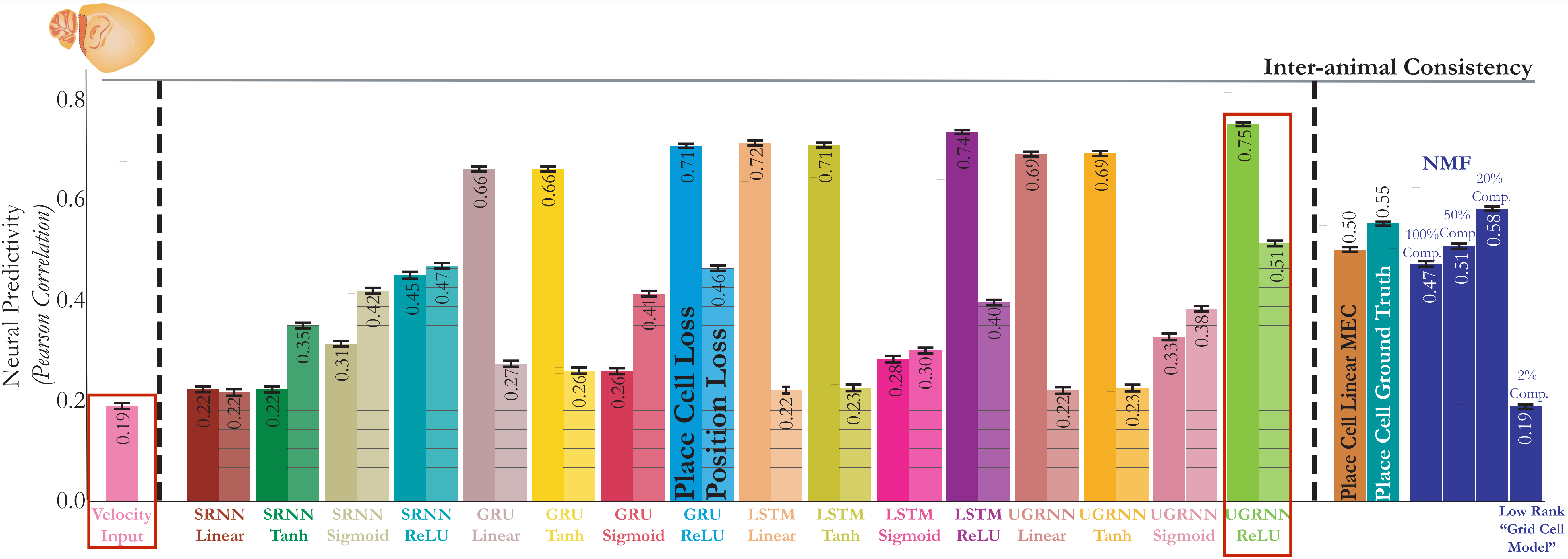
Nonlinearity type affects generalization



Nonnegativity constraint + gating aids in generalization across environments

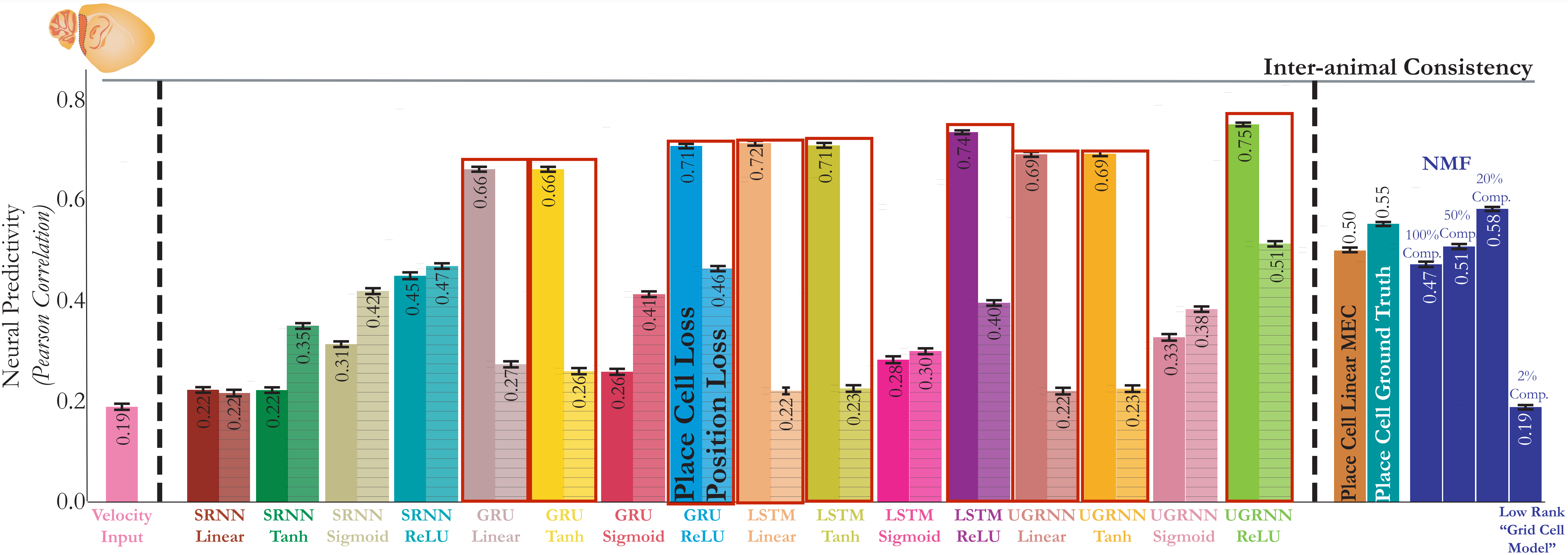
But this nonnegativity constraint must *not* saturate either!

Model input is a poor predictor of population



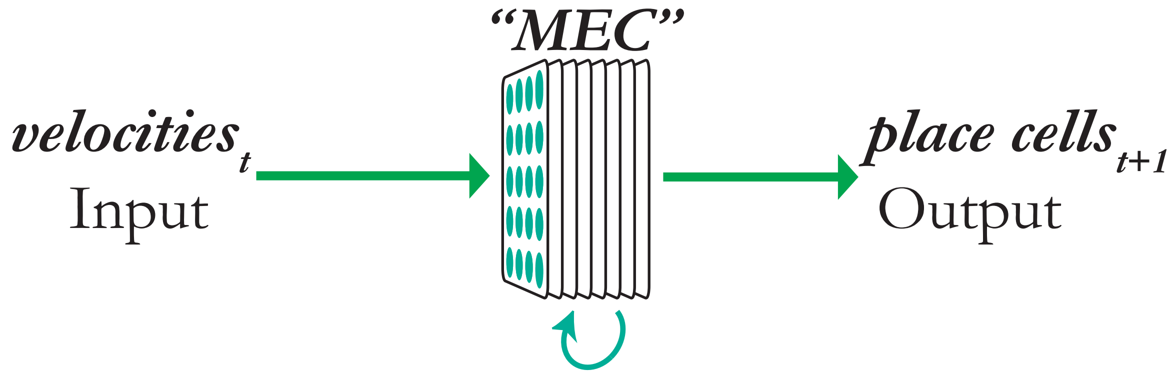
Models add a lot of predictive power to their inputs

Direct path integration *fails* to generalize

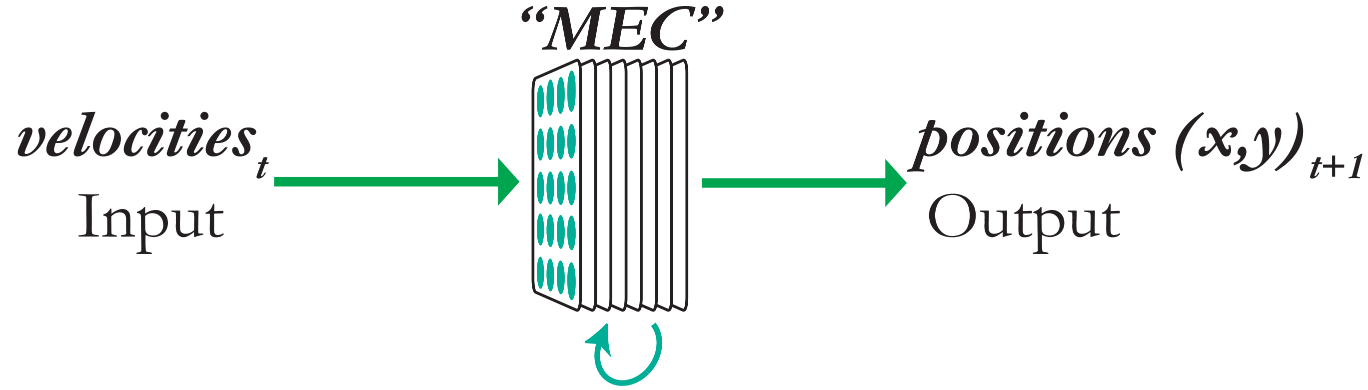


Output place cell supervision provides better generalization over direct supervision of position (path integration)

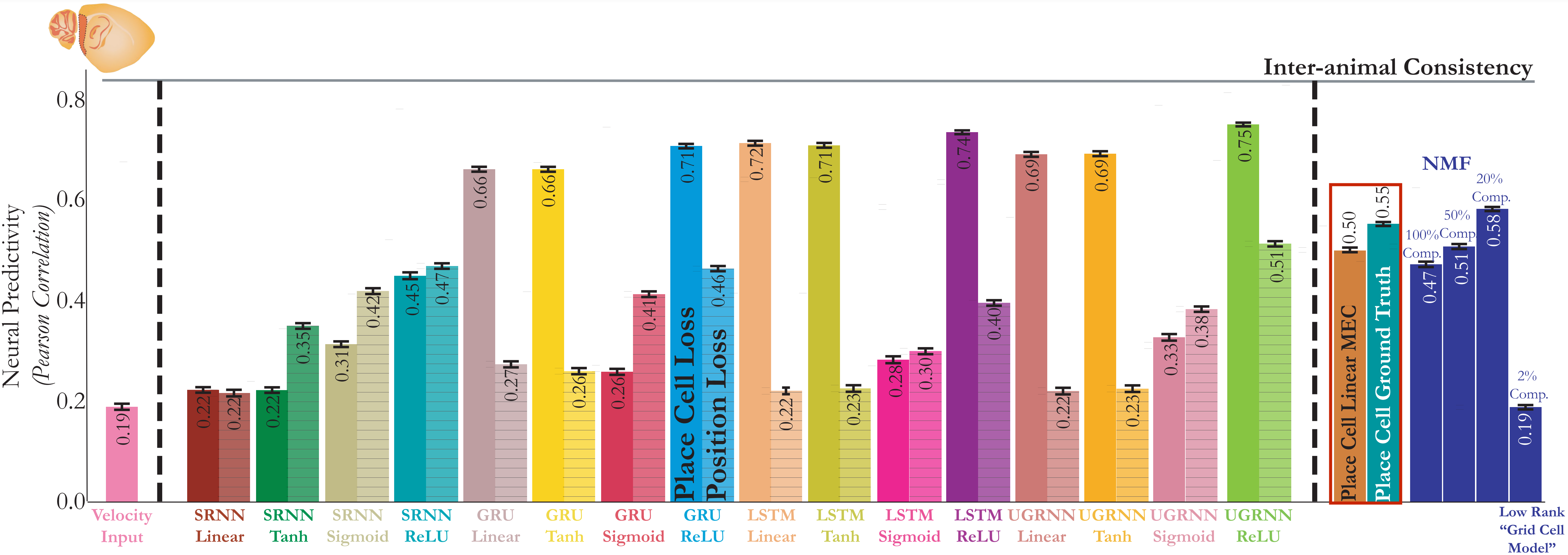
$$\mathcal{L}(\hat{p}, p) := -\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{N_p} p_i^t \log \hat{p}_i^t$$



$$\mathcal{L}(\hat{p}, p) := \frac{1}{2}\frac{1}{T}\sum_{t=1}^{T}\left(\left(p_x^t - \hat{p}_x^t\right)^2 + \left(p_y^t - \hat{p}_y^t\right)^2\right)$$

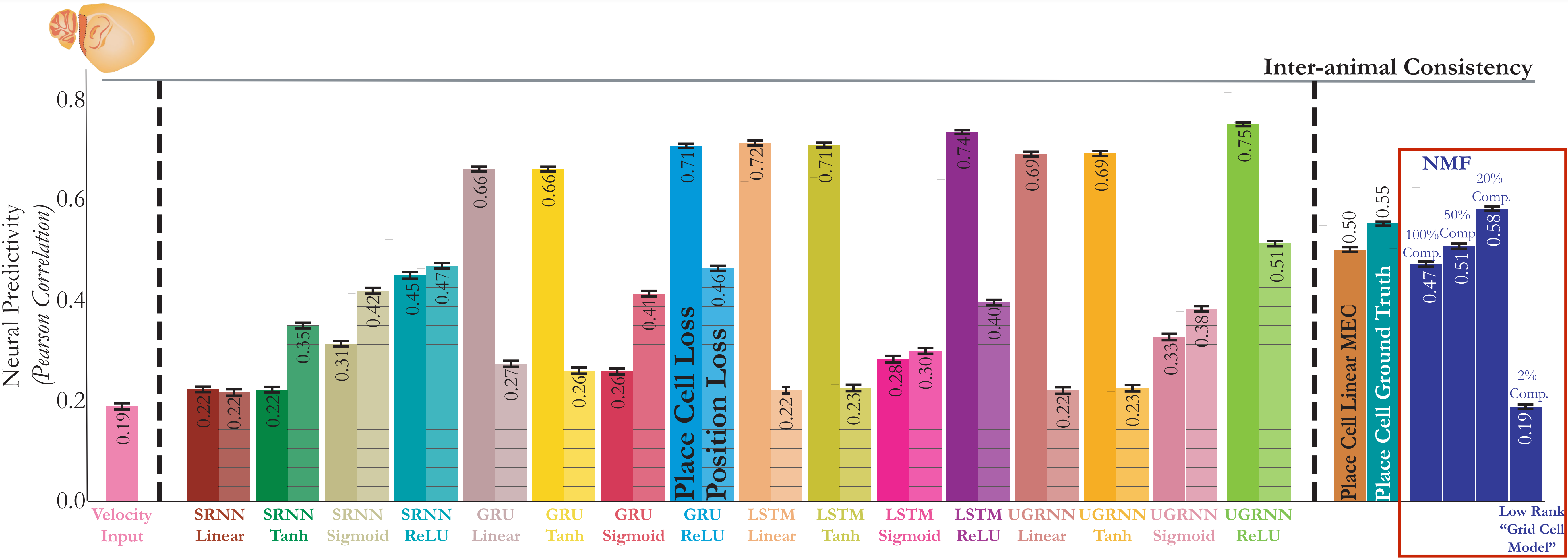


Place cells alone are a poor predictor



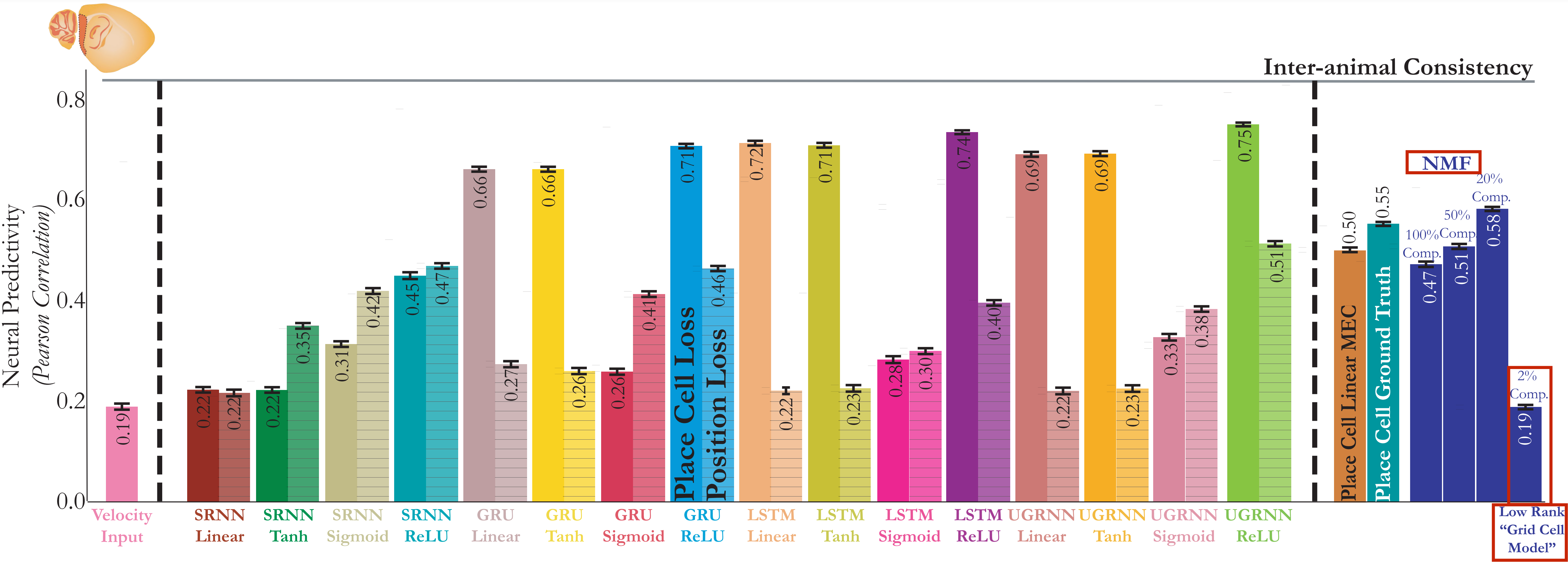
But place cells alone are *not* a good predictor of MEC (good!)
You actually need to integrate them!

NMF is also a poor predictor



Dimensionality reduction on place cells is *not* a good predictor of MEC either

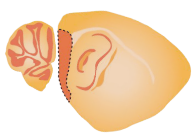
Grid cell oriented NMF is a poor predictor

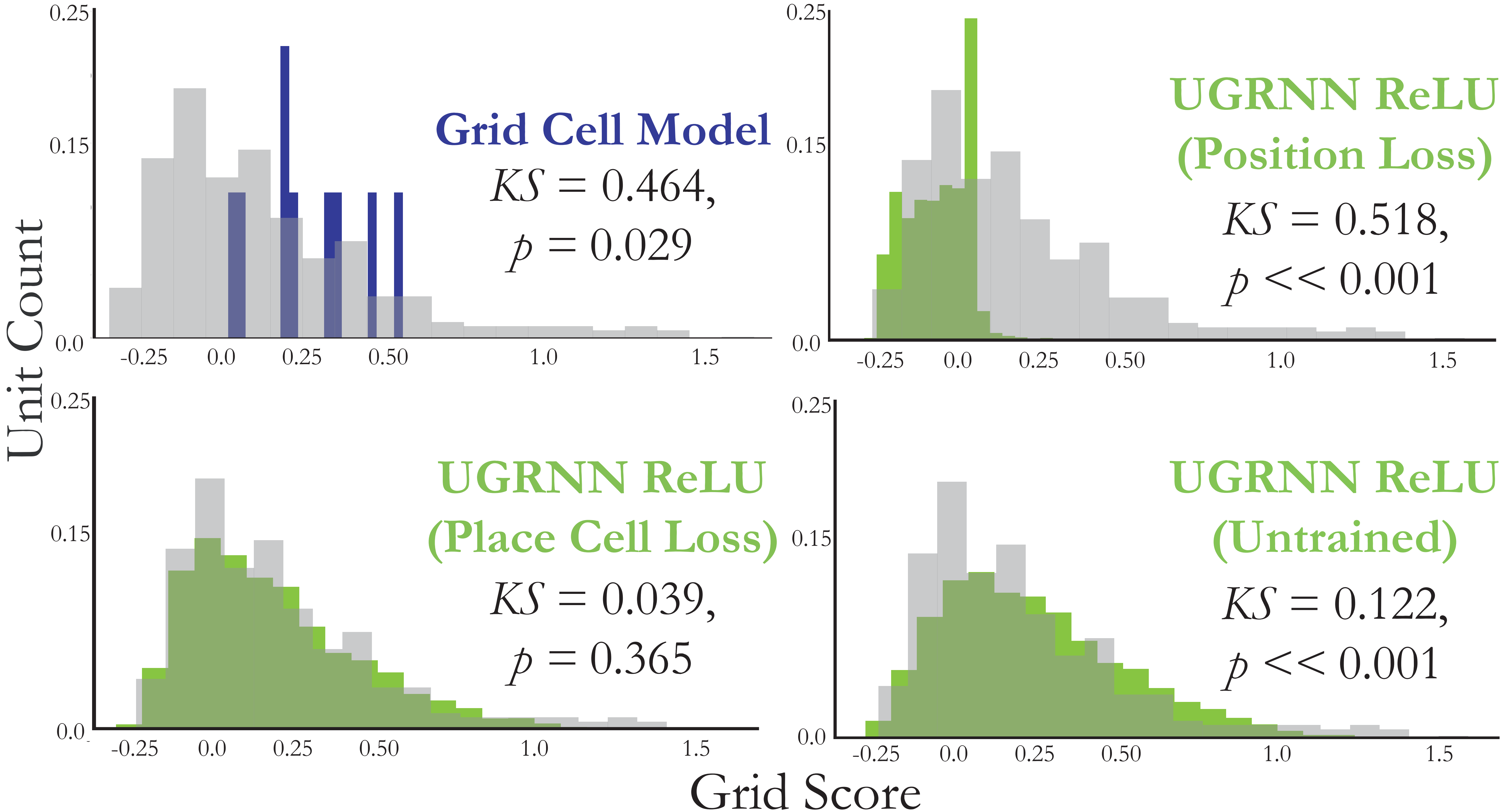


Grid cell oriented model is an especially *poor* predictor!

Task-optimized navigational models best predict the *entire* MEC population

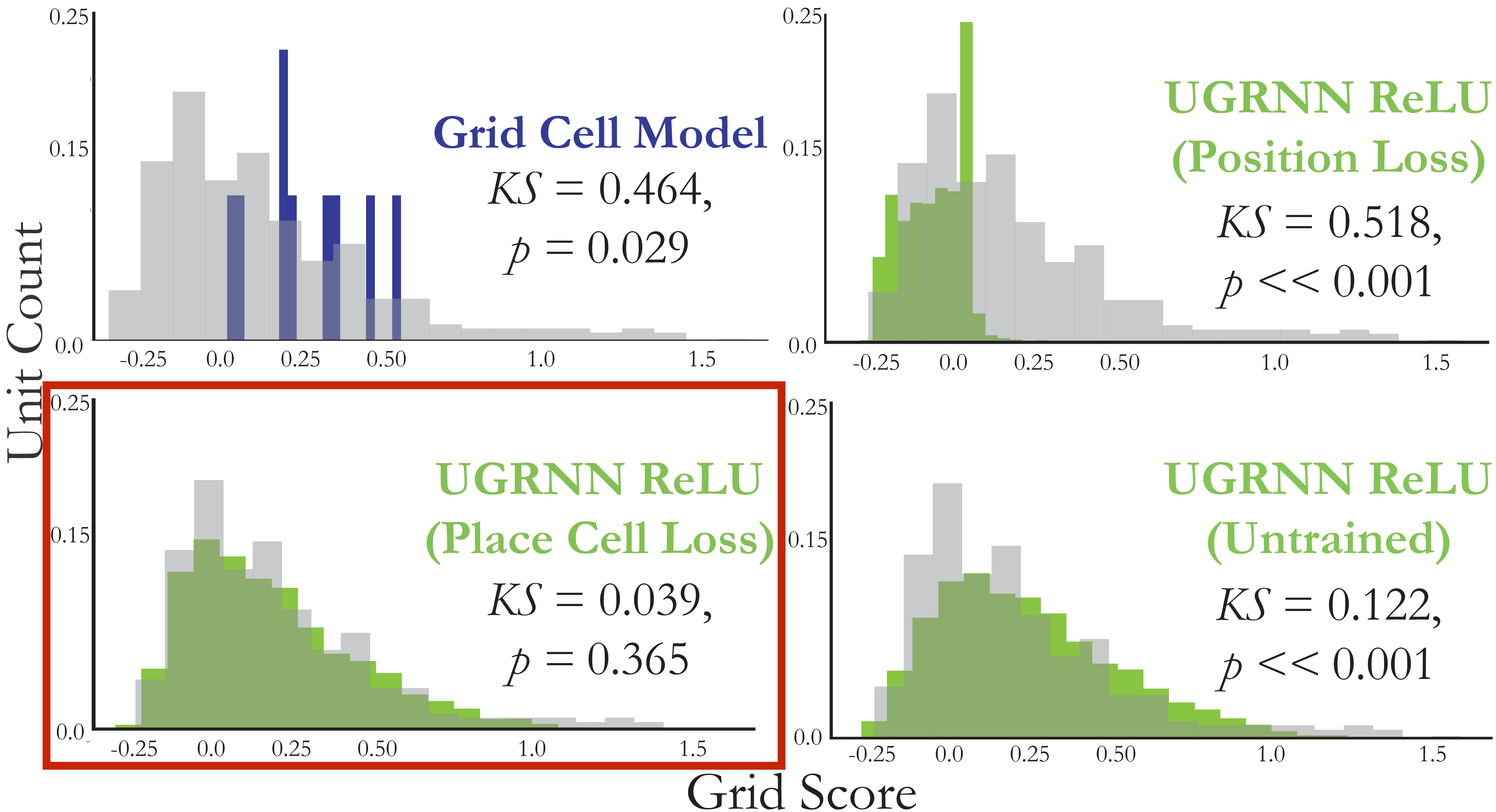
Grid score distribution does not require any parameter fitting





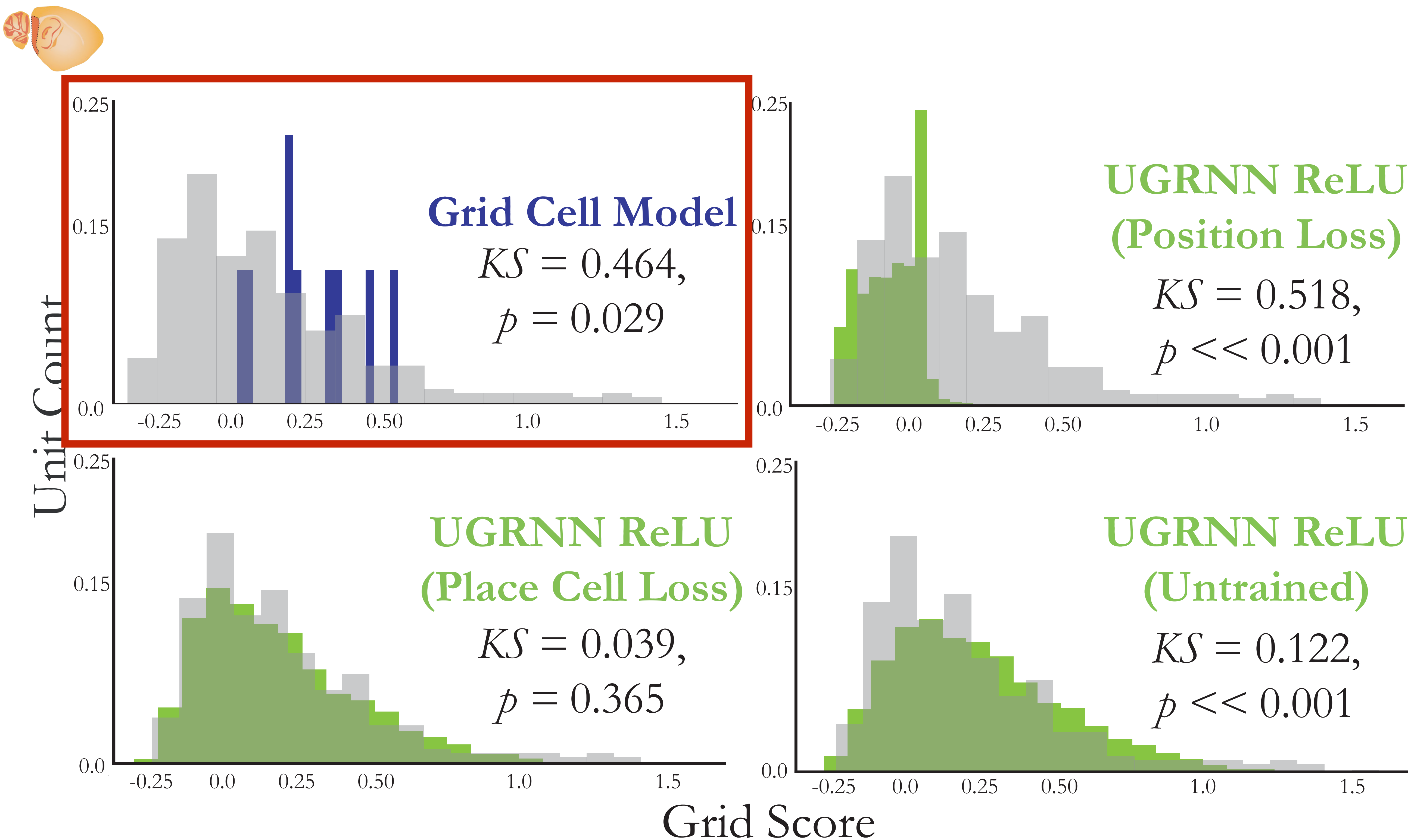
Task-optimized navigational models best predict the *entire* MEC population

Best model class in terms of neural predictivity also matches grid score distribution in its own synthetic population



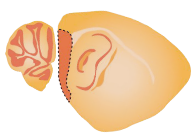
Task-optimized navigational models best predict the *entire* MEC population

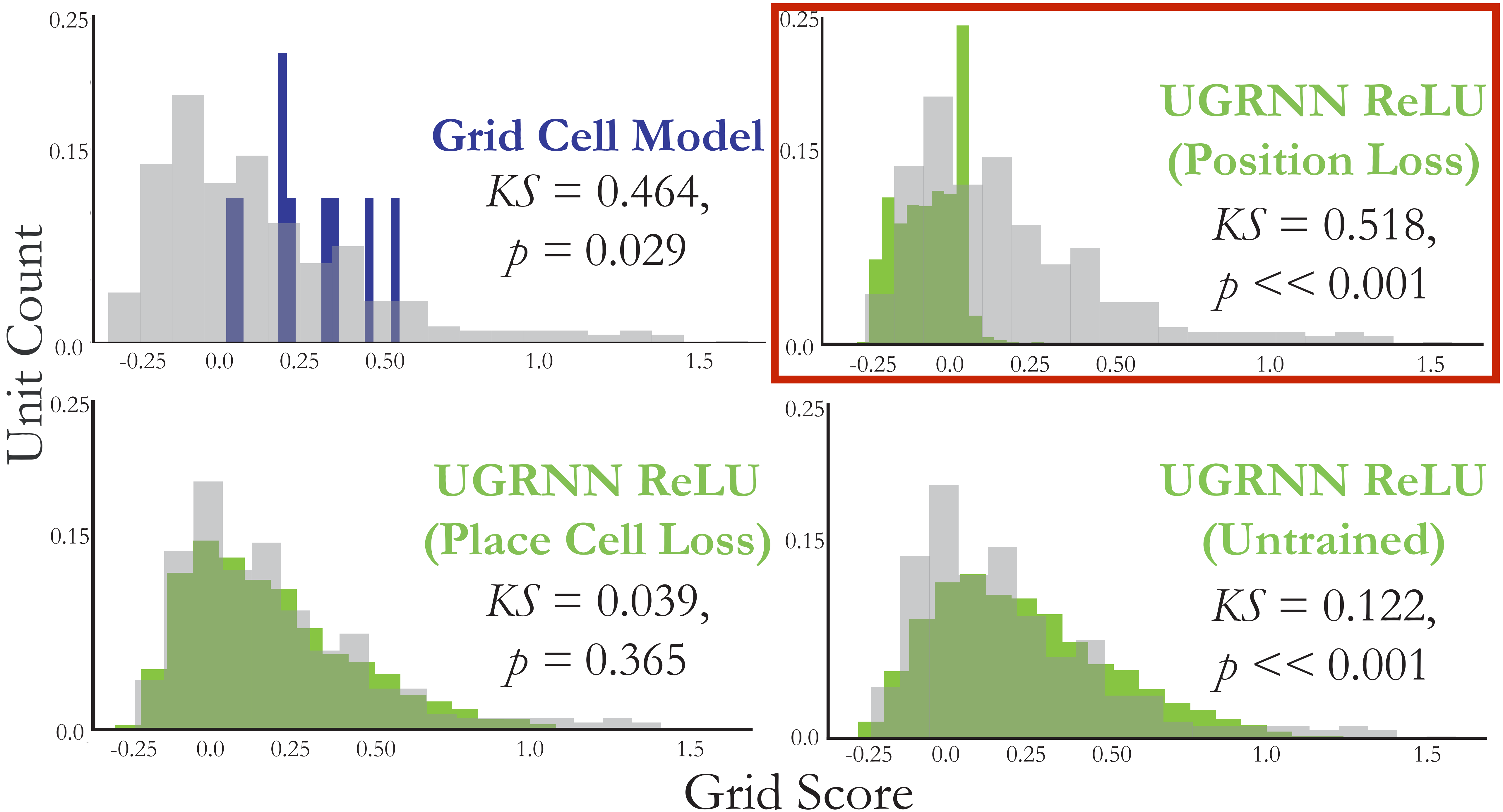
Low-rank model is too biased towards grid-like units



Task-optimized navigational models best predict the *entire* MEC population

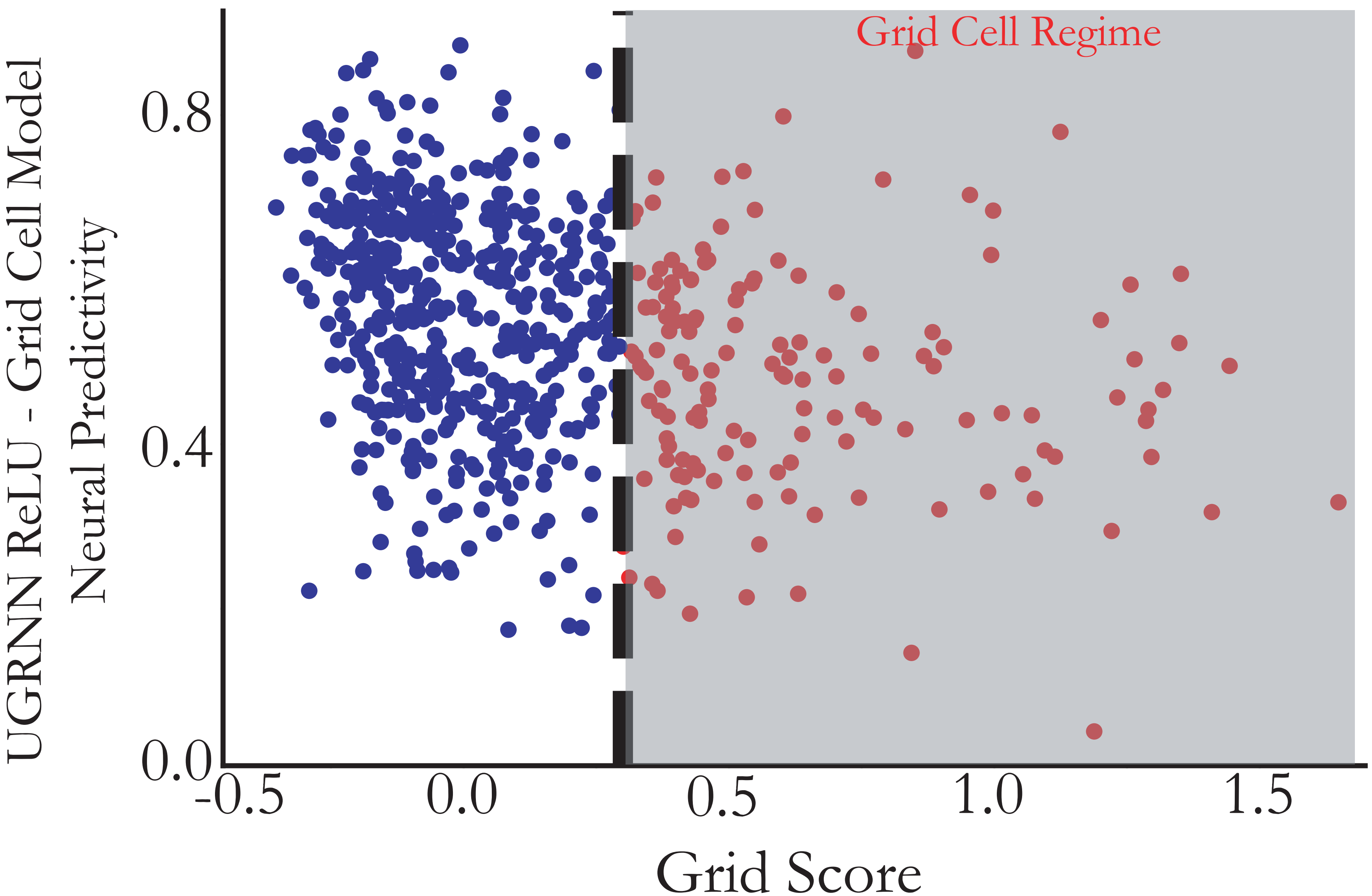
Without place cell integration, the model is too biased towards *non* grid-like units

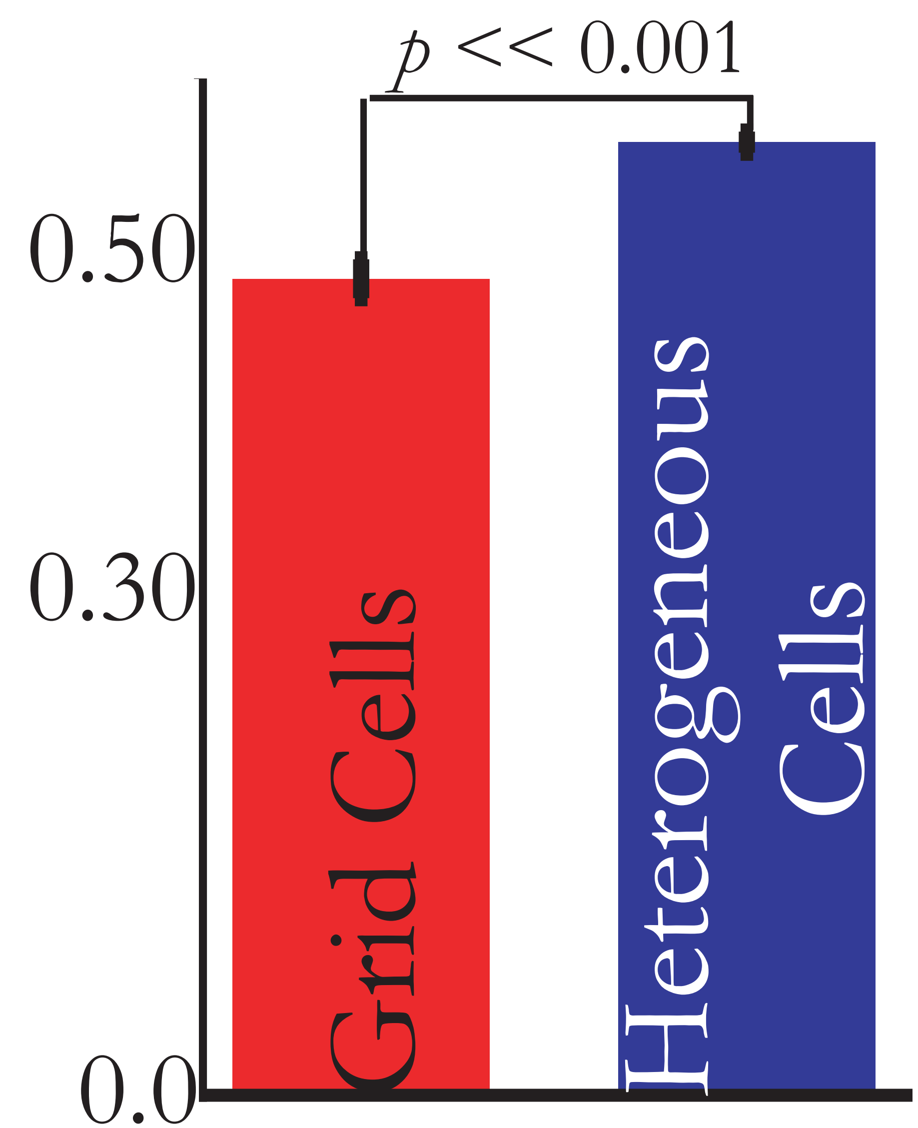




Neural network model better predicts heterogeneous cells

Neural network models are differentially better at heterogeneous cells than NMF

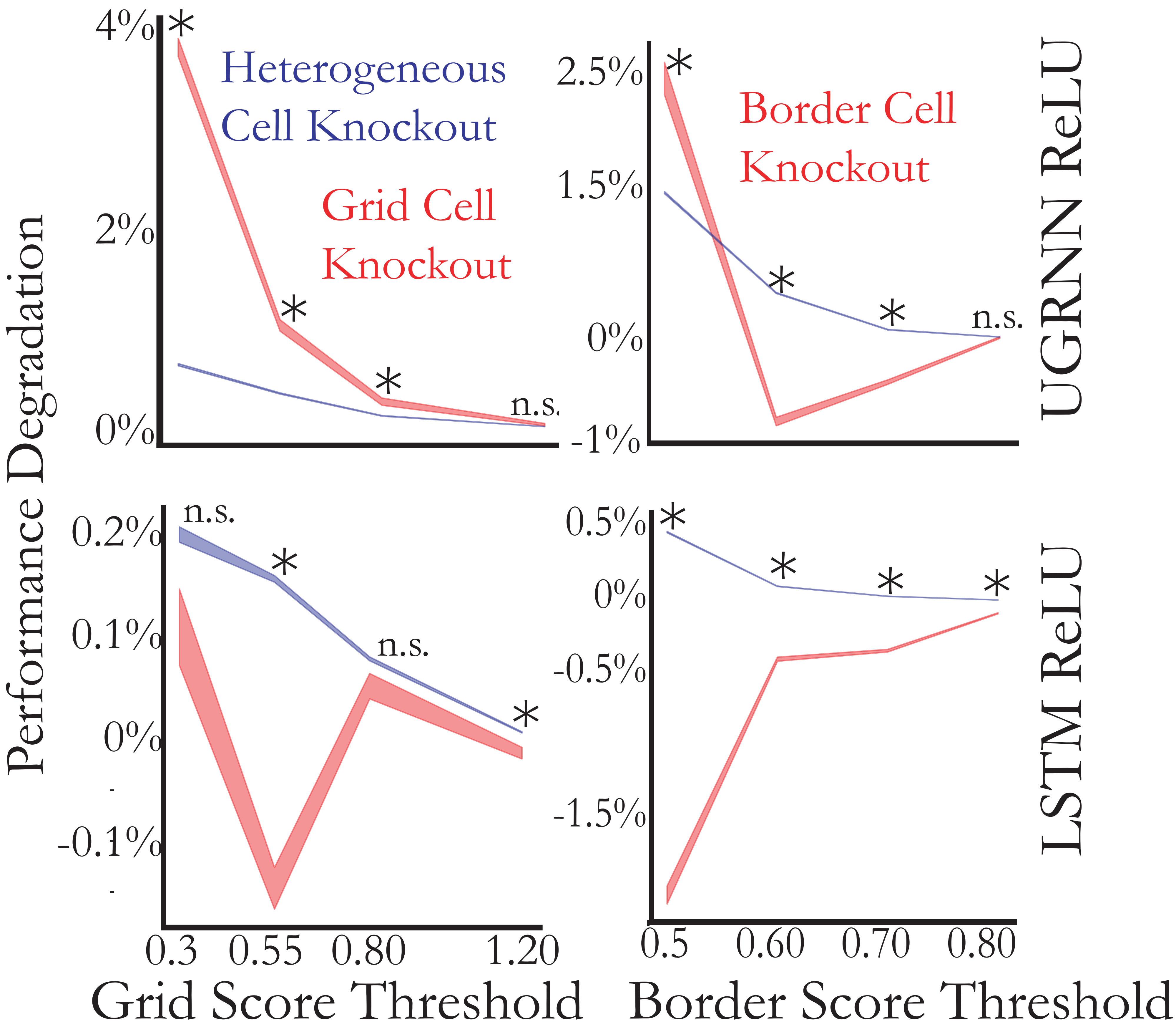




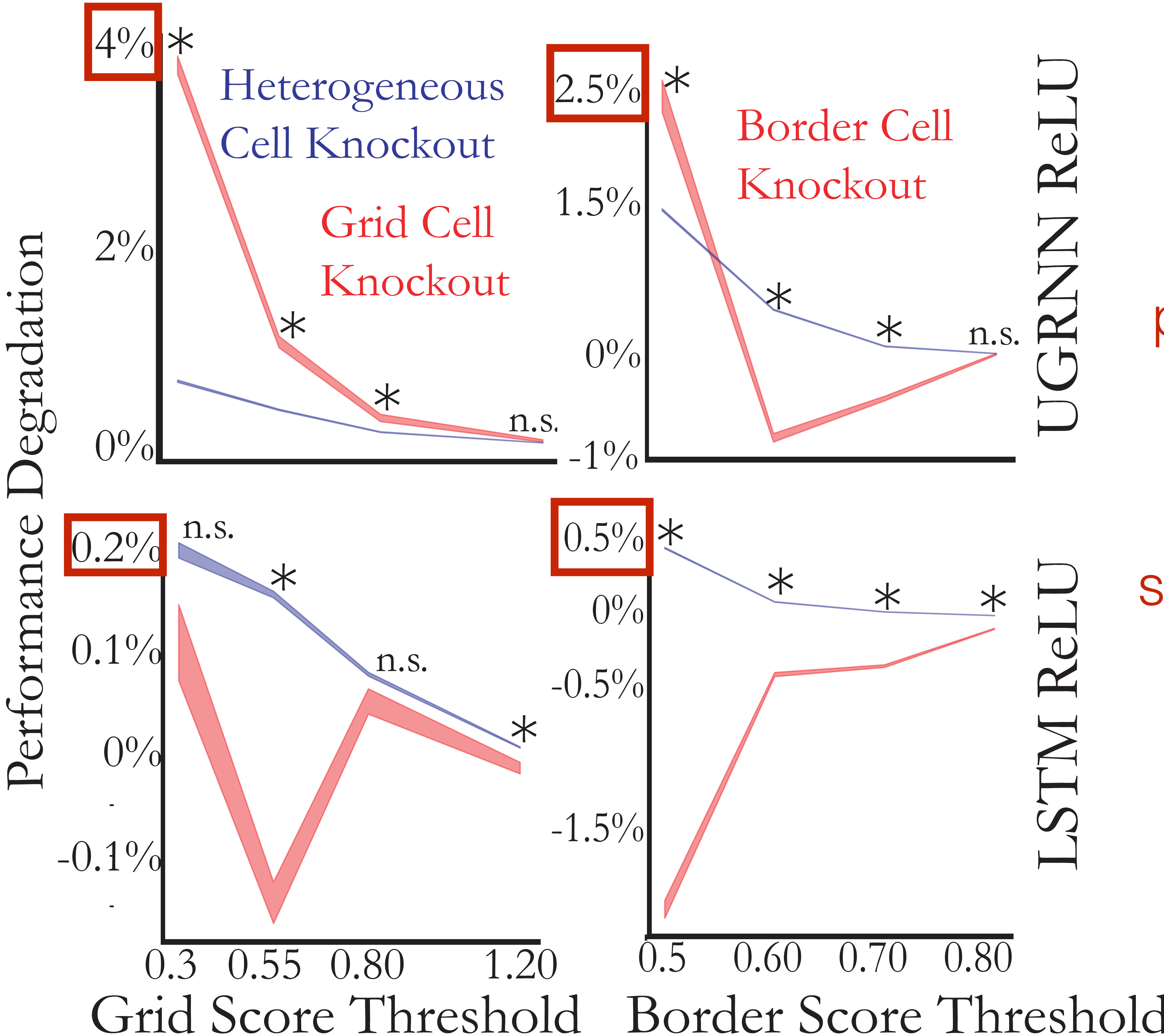
Knockout experiments

Given that we have a model that exhibits close similarity to MEC, we can use it to generate predictions for experiments that are very difficult to do

Knockout experiments

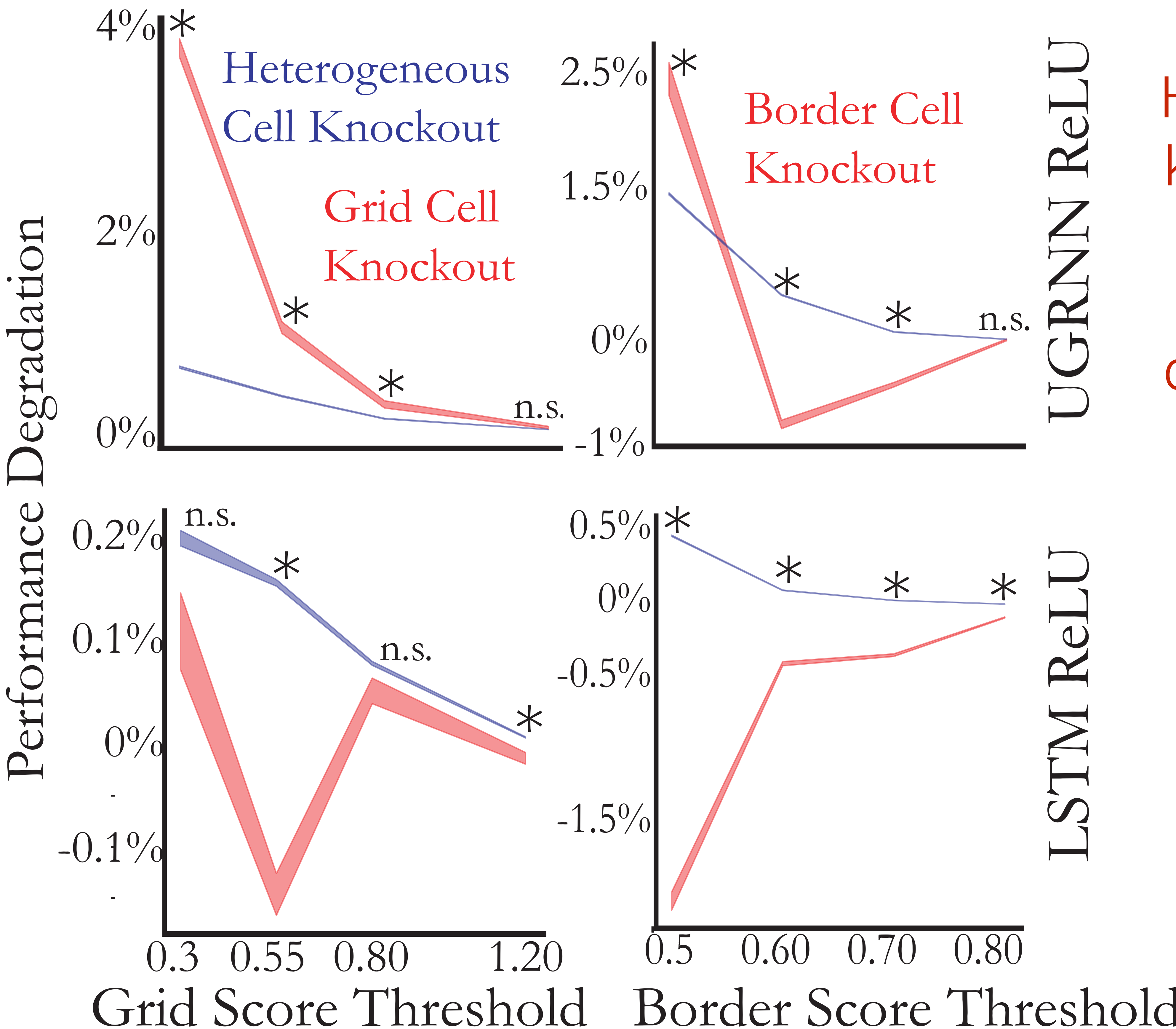


Networks are robust to knockouts



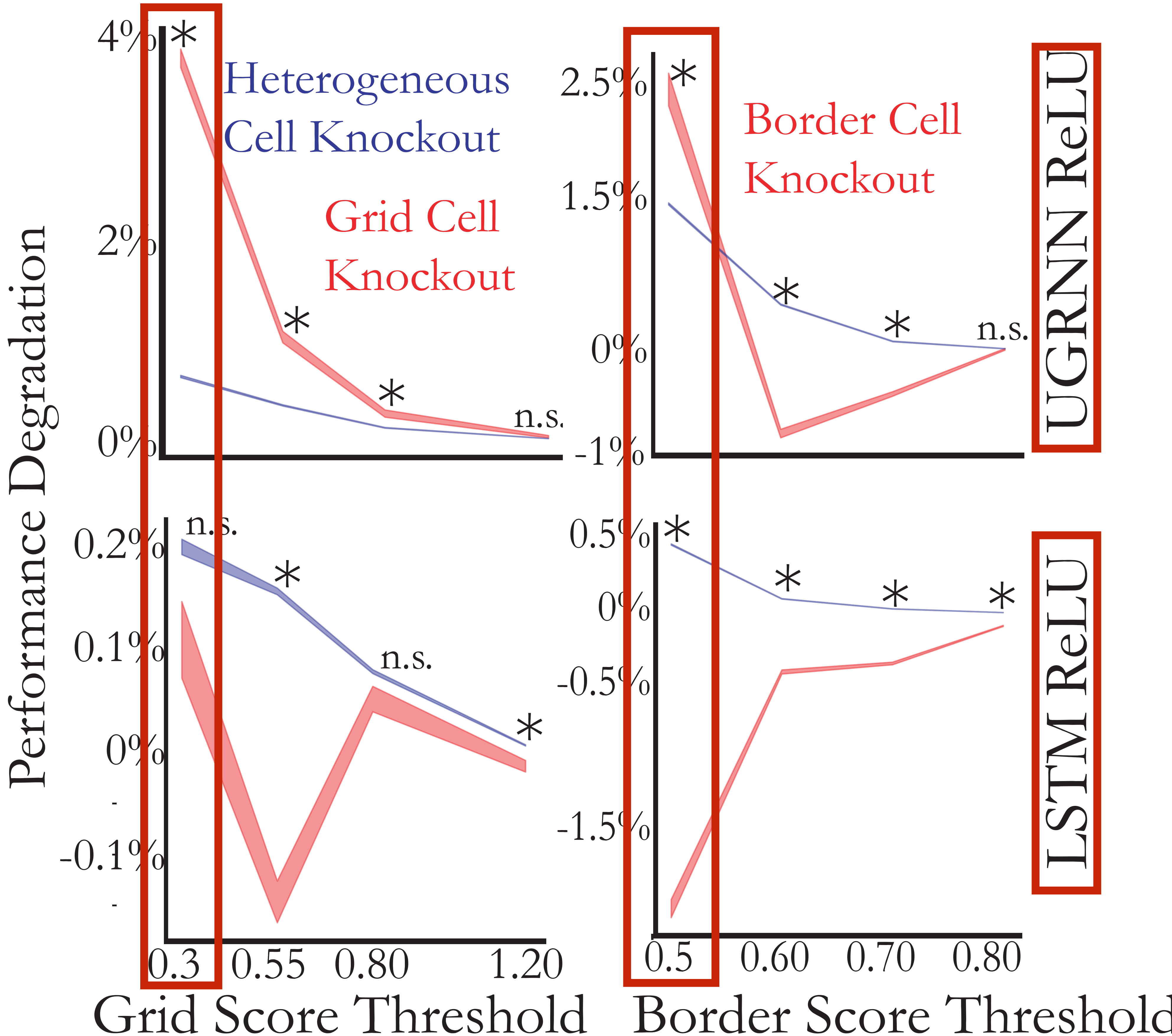
Network performance is robust to knockouts on the order of several hundred units

Heterogeneous cells are relevant to navigation



Heterogeneous knockout gives similar performance degradation as cell type specific knockout, especially as threshold increases

Differences in gating architecture

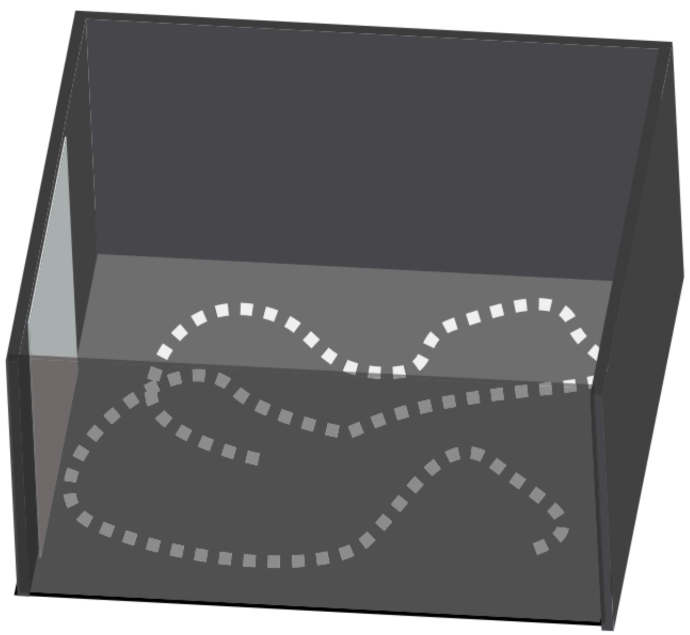


At the lowest threshold of cell type specificity, different gating architectures give somewhat different predictions, which may be useful to gather evidence for in future experiments

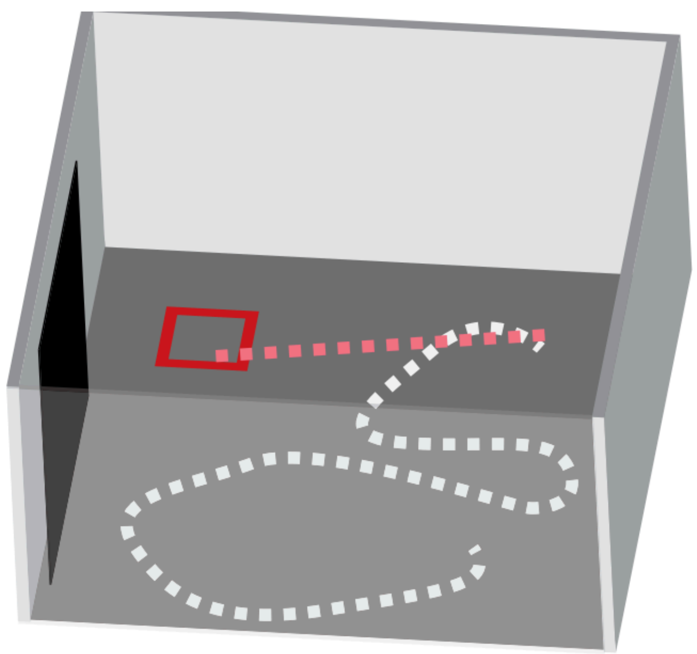
Remapping in the presence of reward

Remembered reward locations restructure entorhinal spatial maps

William N. Butler*, Kiah Hardcastle*, Lisa M. Giocomo†



free foraging (ENV1)

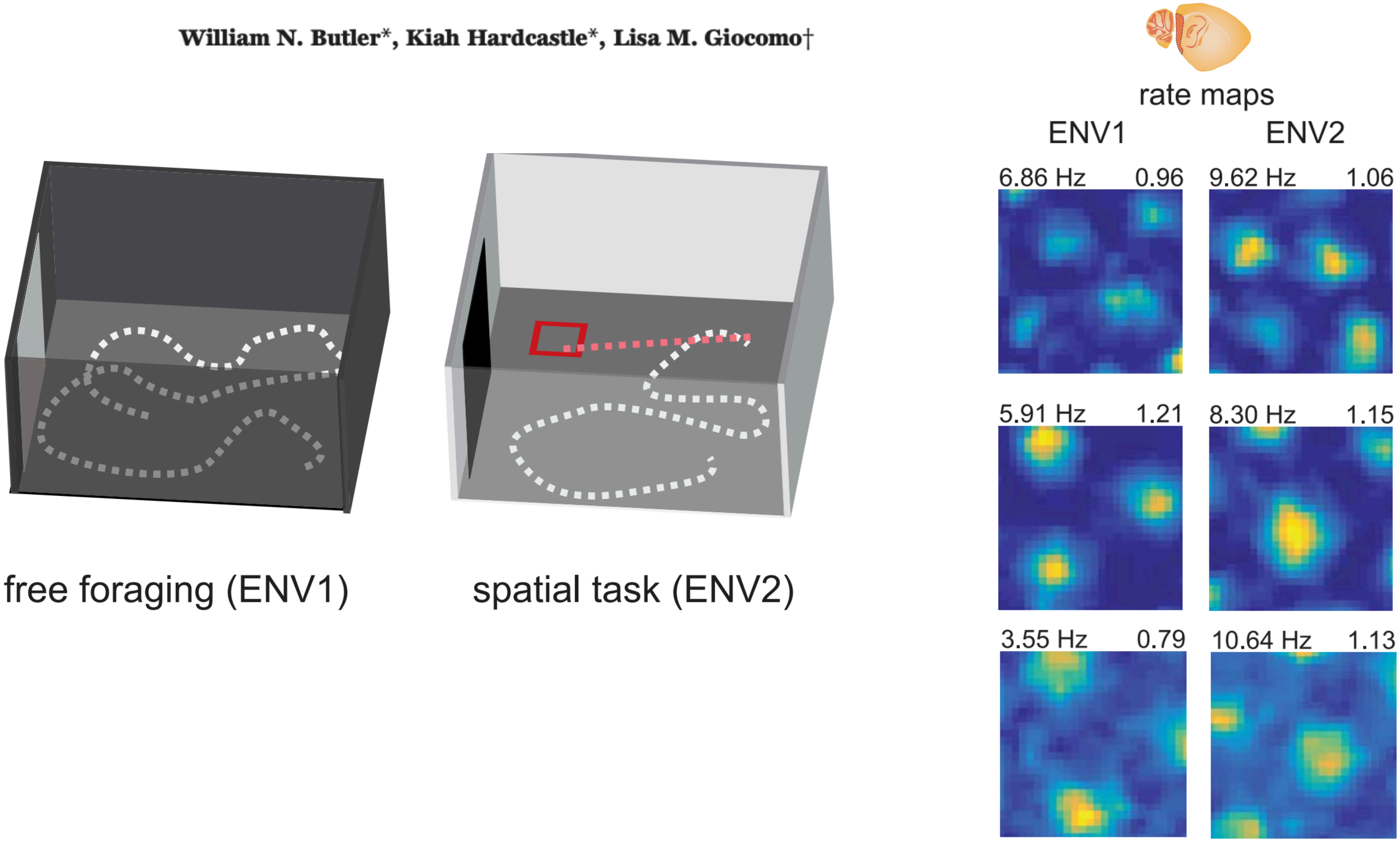


spatial task (ENV2)

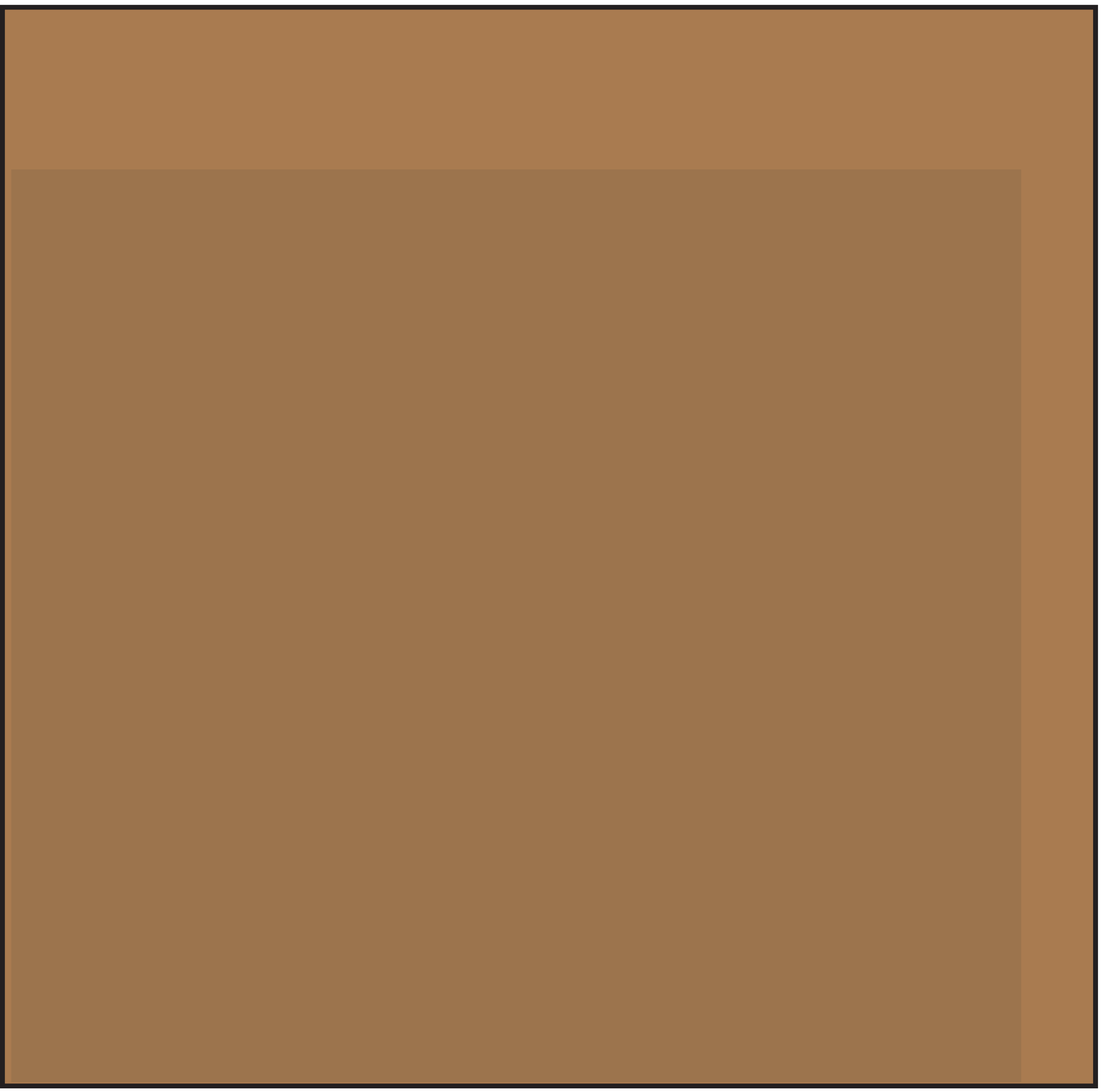
Remapping in the presence of reward

Remembered reward locations restructure entorhinal spatial maps

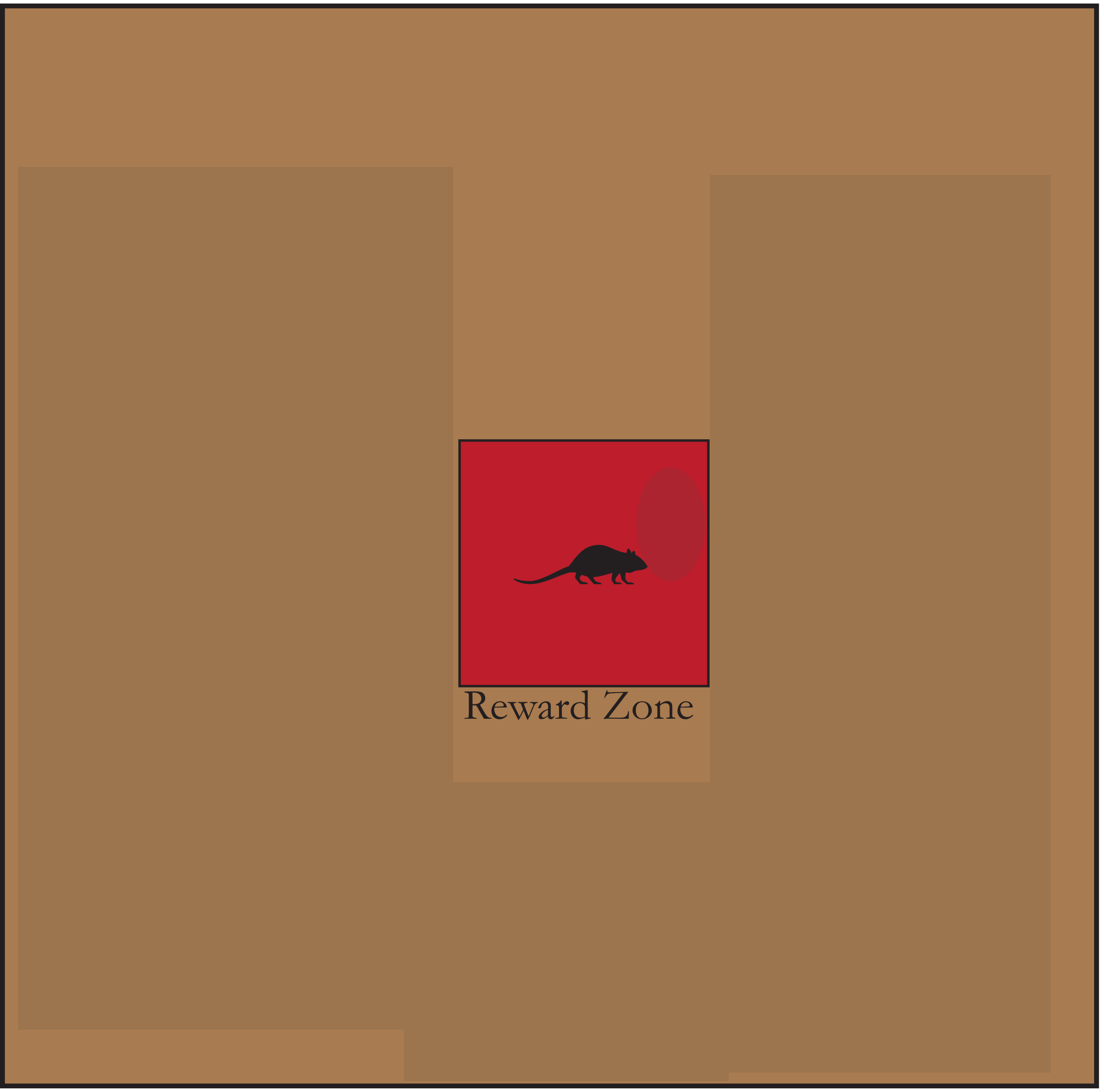
William N. Butler*, Kiah Hardcastle*, Lisa M. Giocomo†



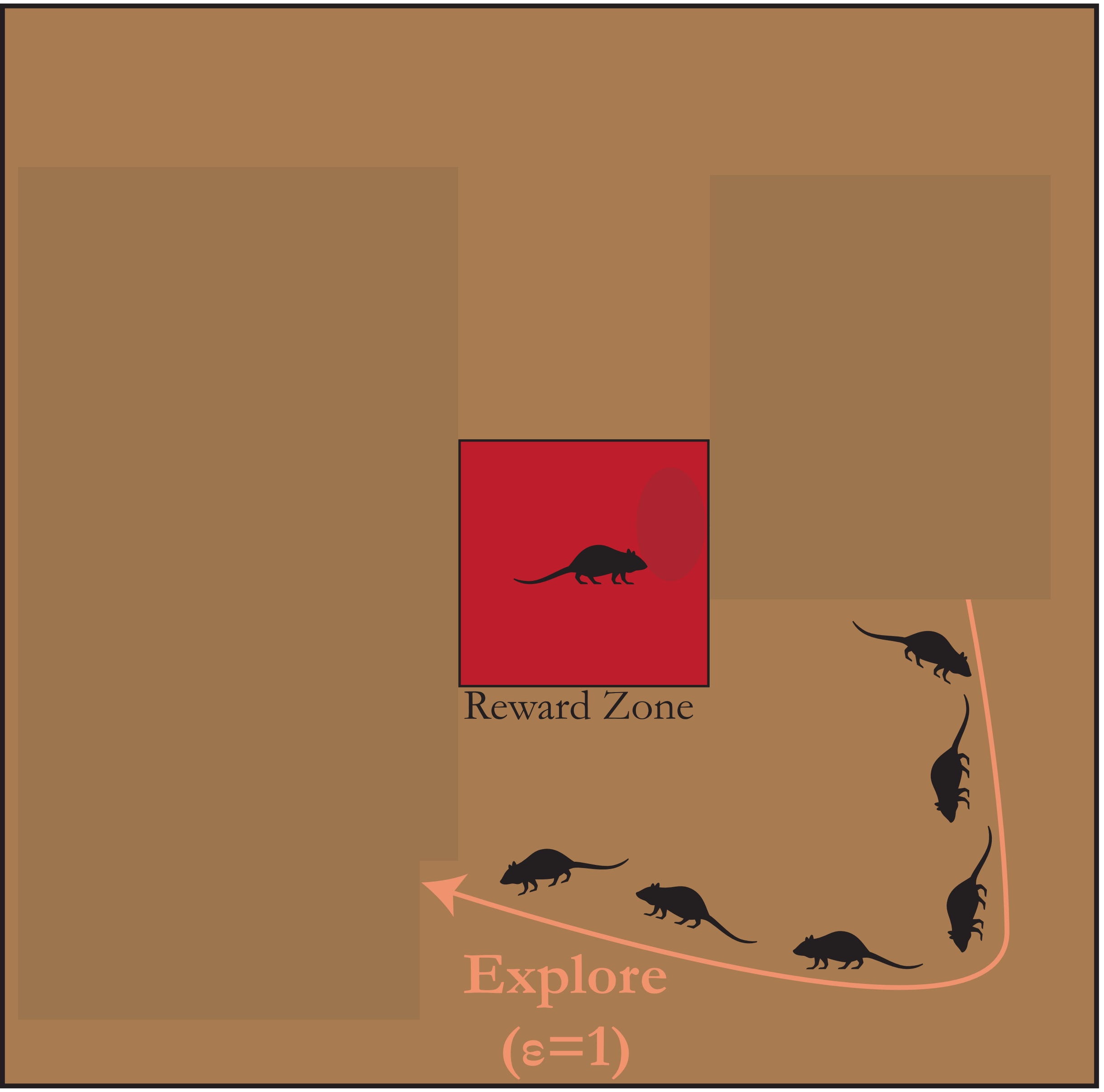
Modeling rewards



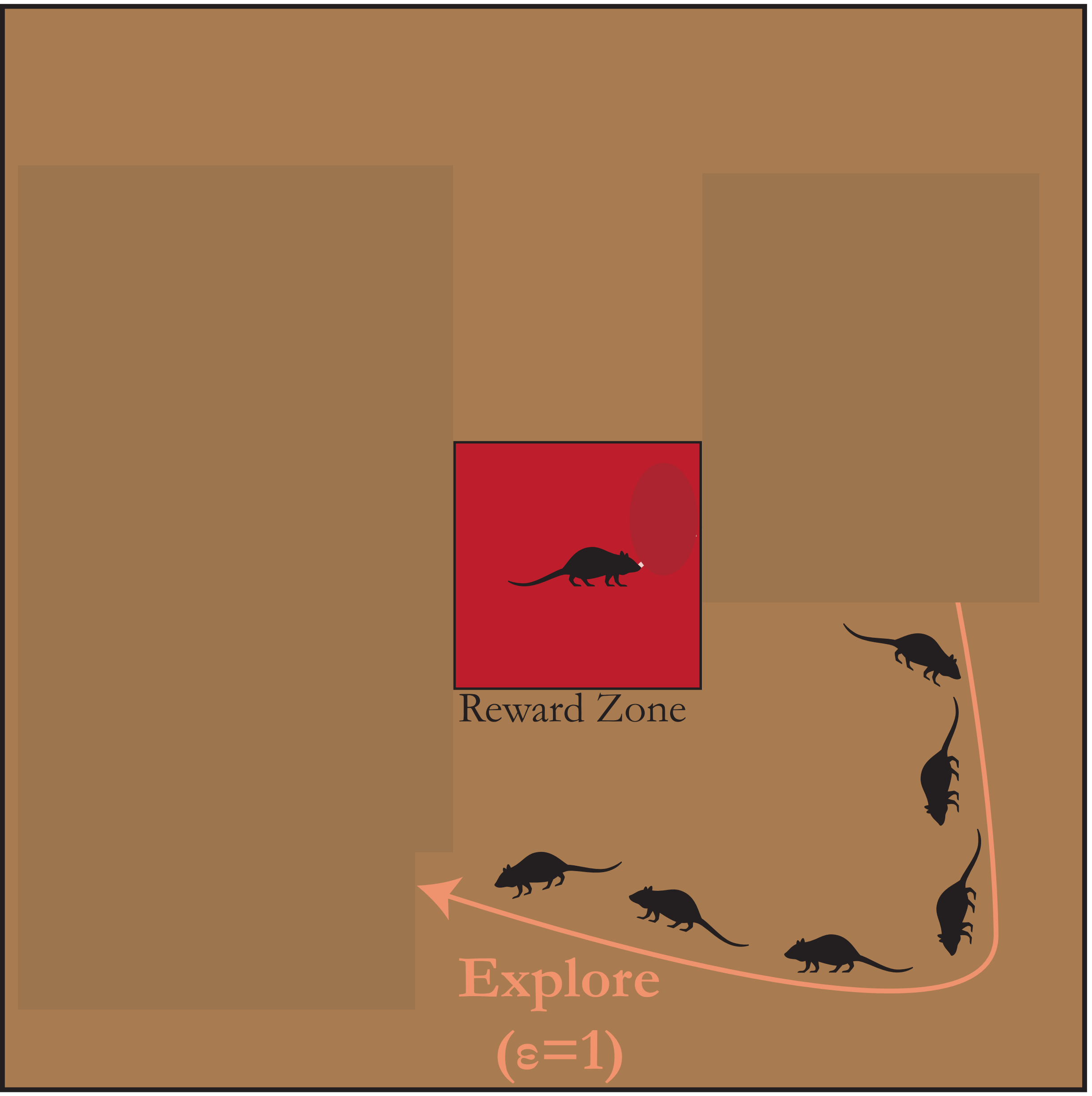
Modeling rewards

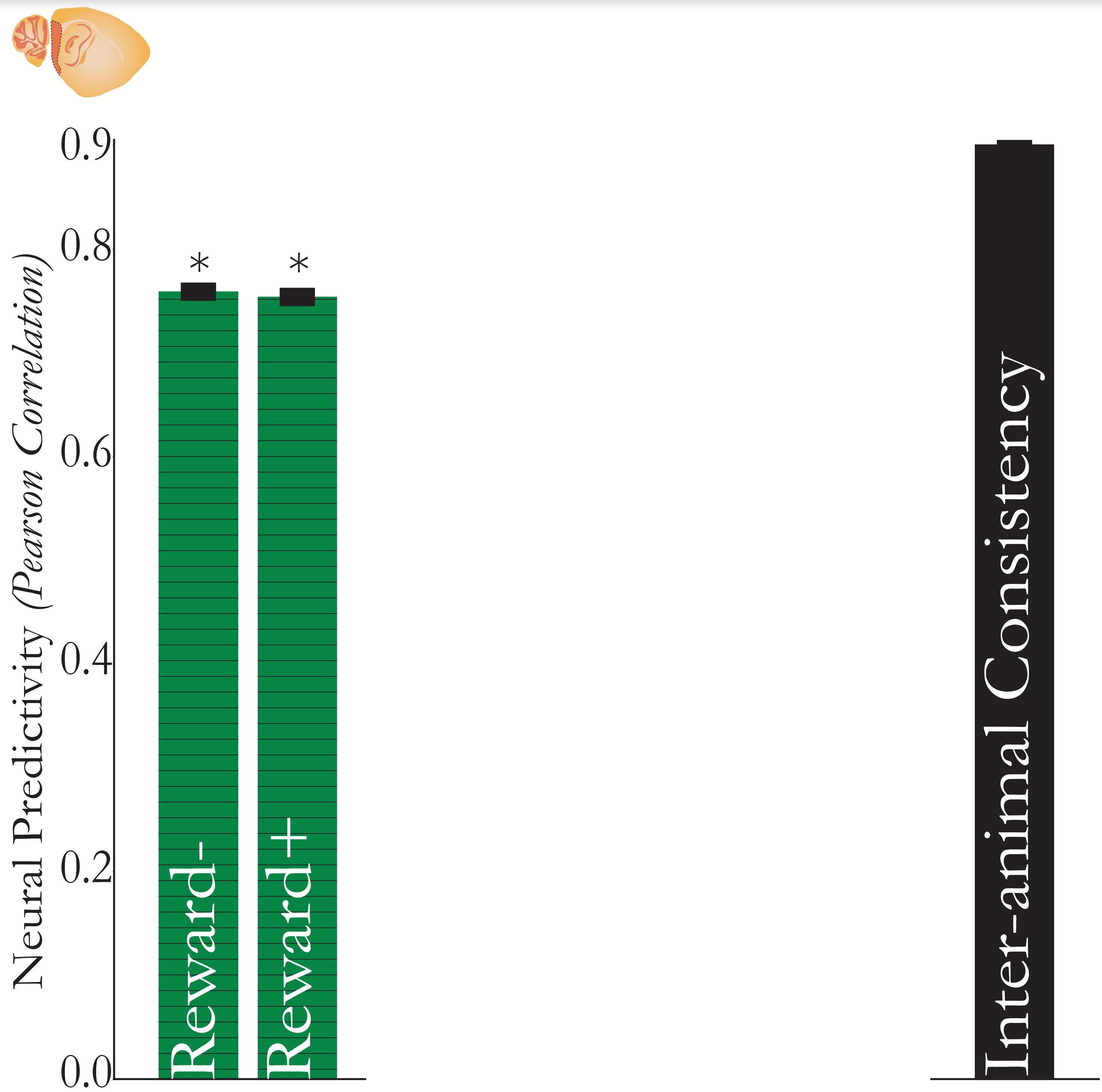


Modeling rewards - What we have done previously

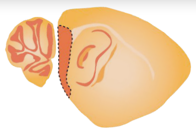


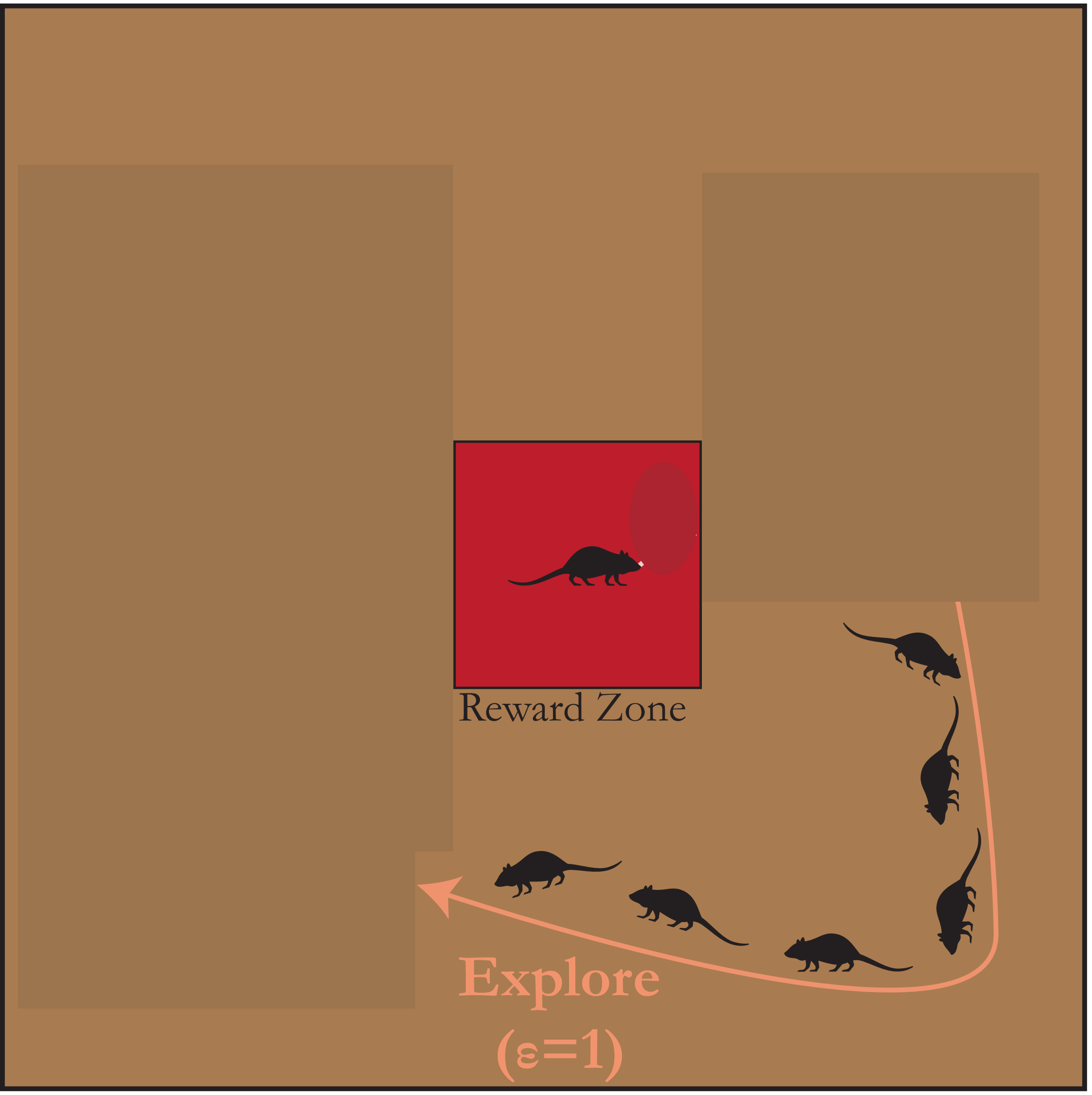
Exploration only model captures each condition *separately*

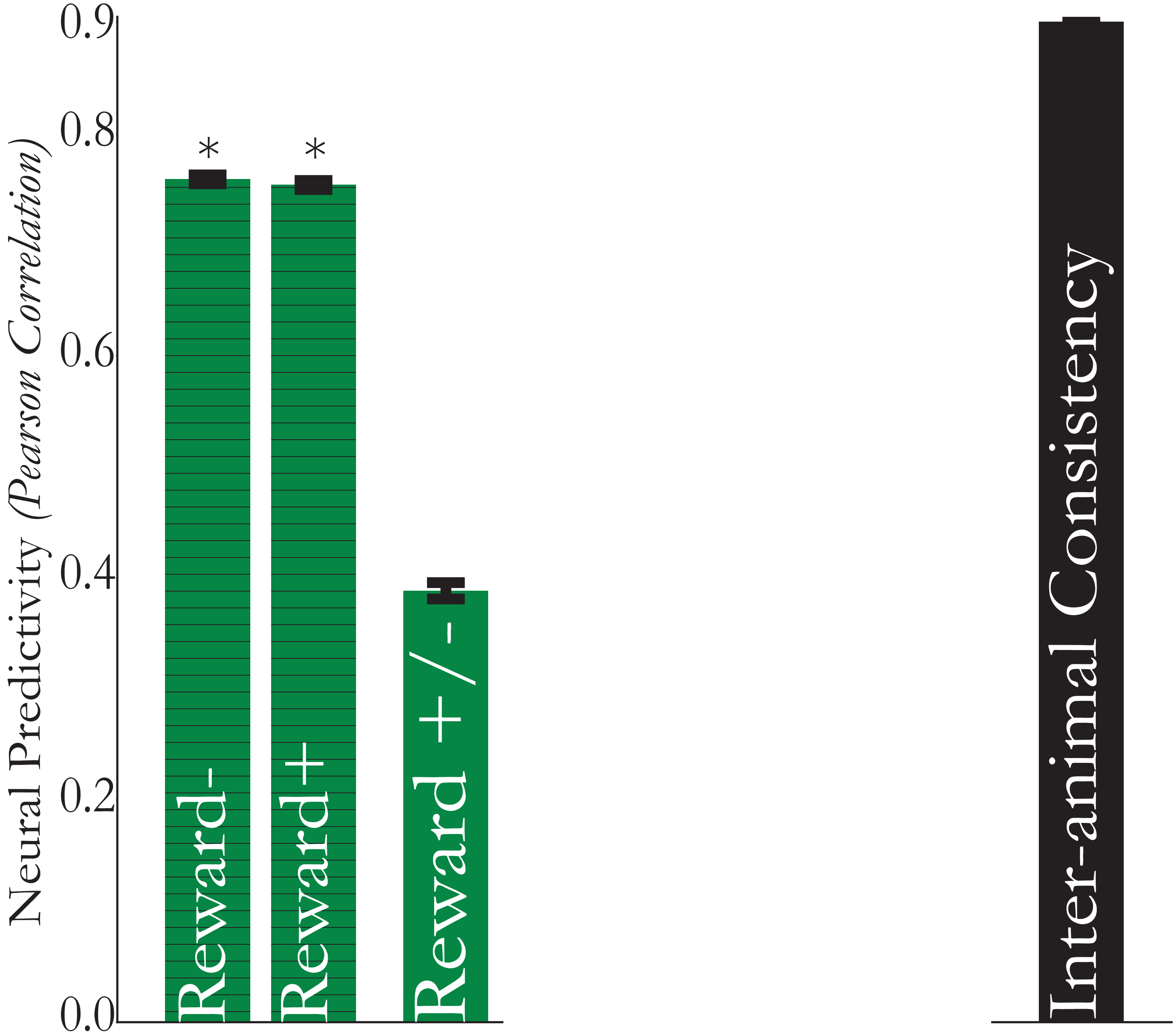




Exploration only model fails to capture remapping

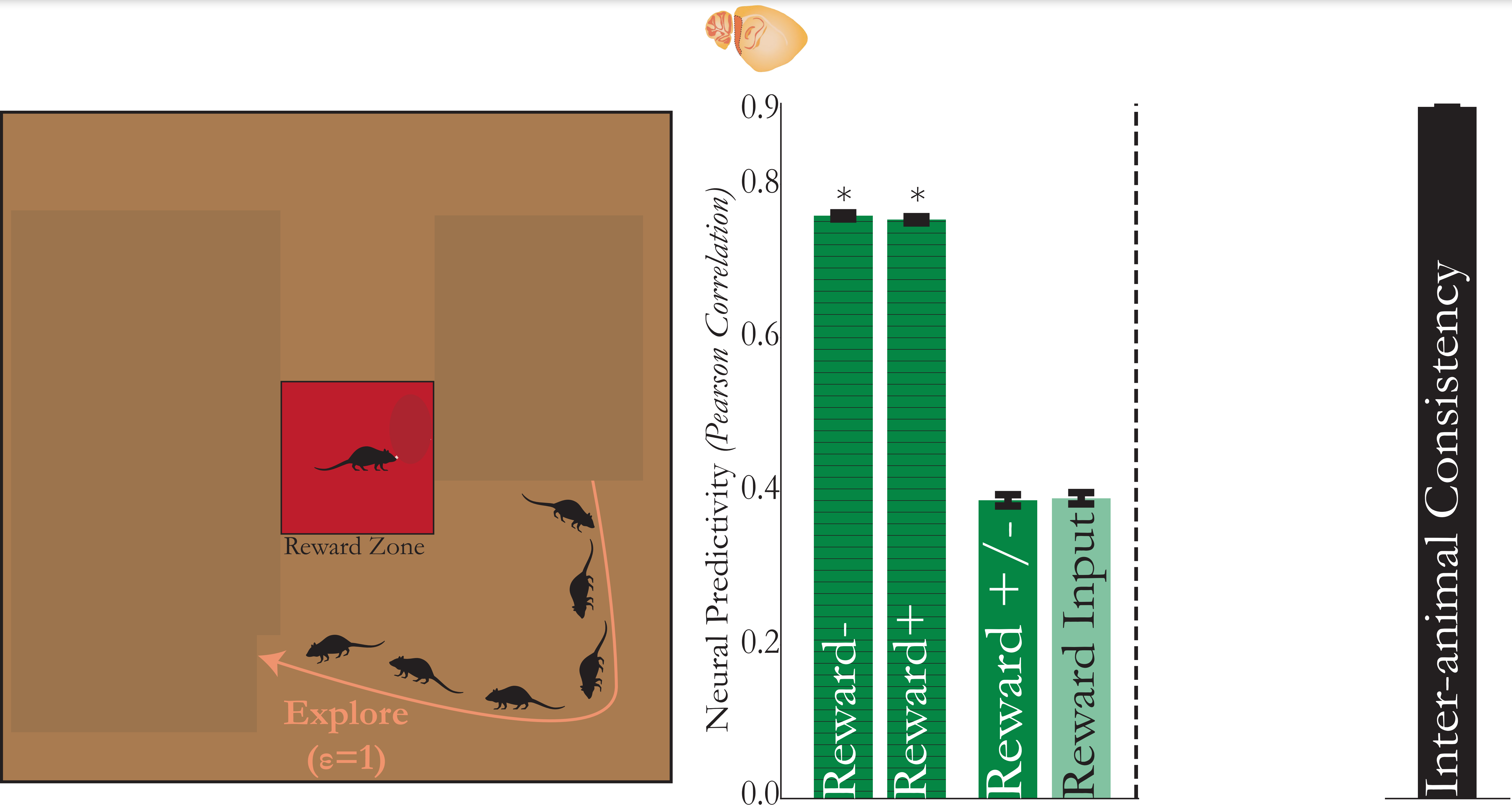






Failure of pure exploration!

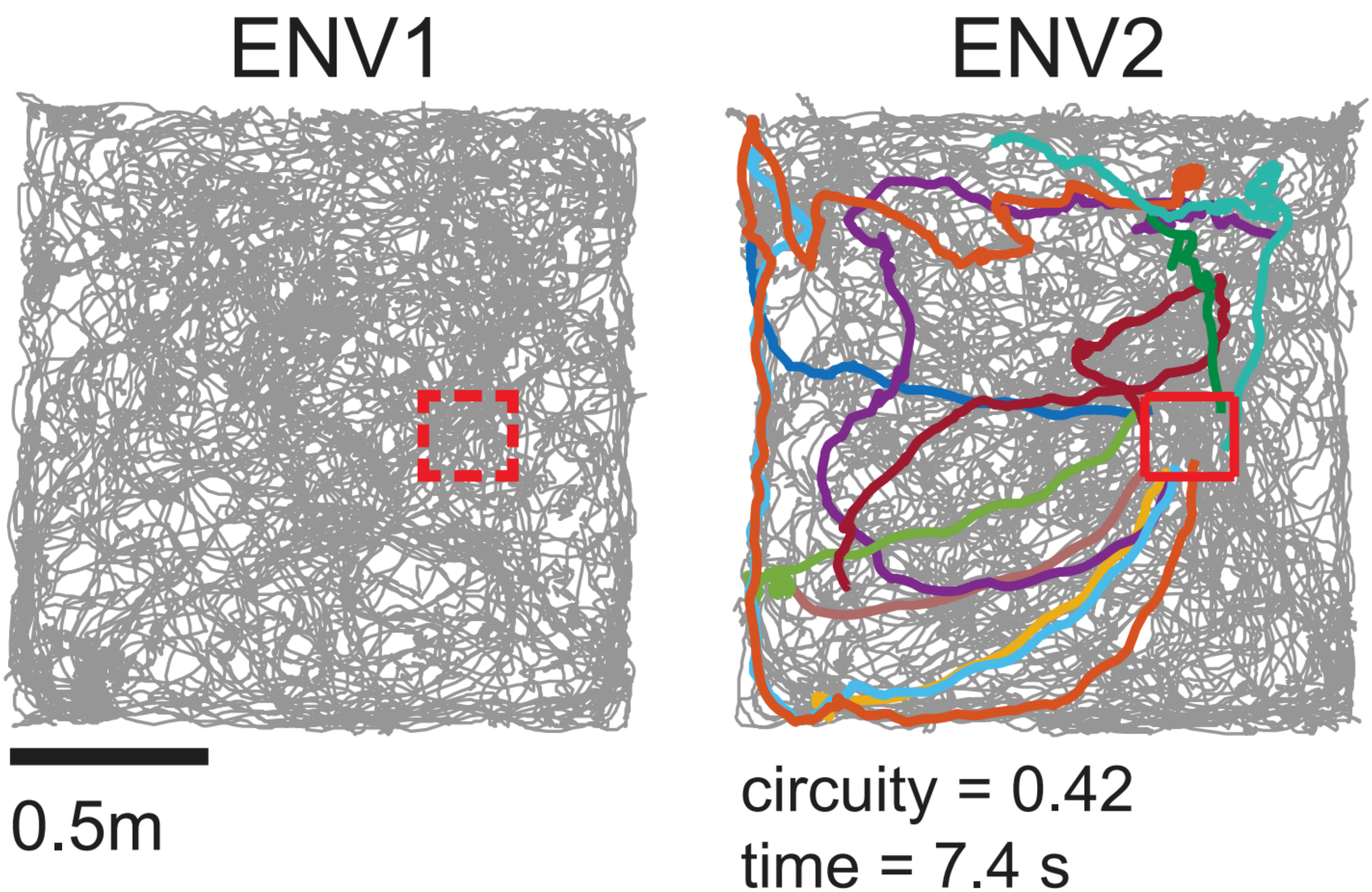
Reward must be extrinsically modeled



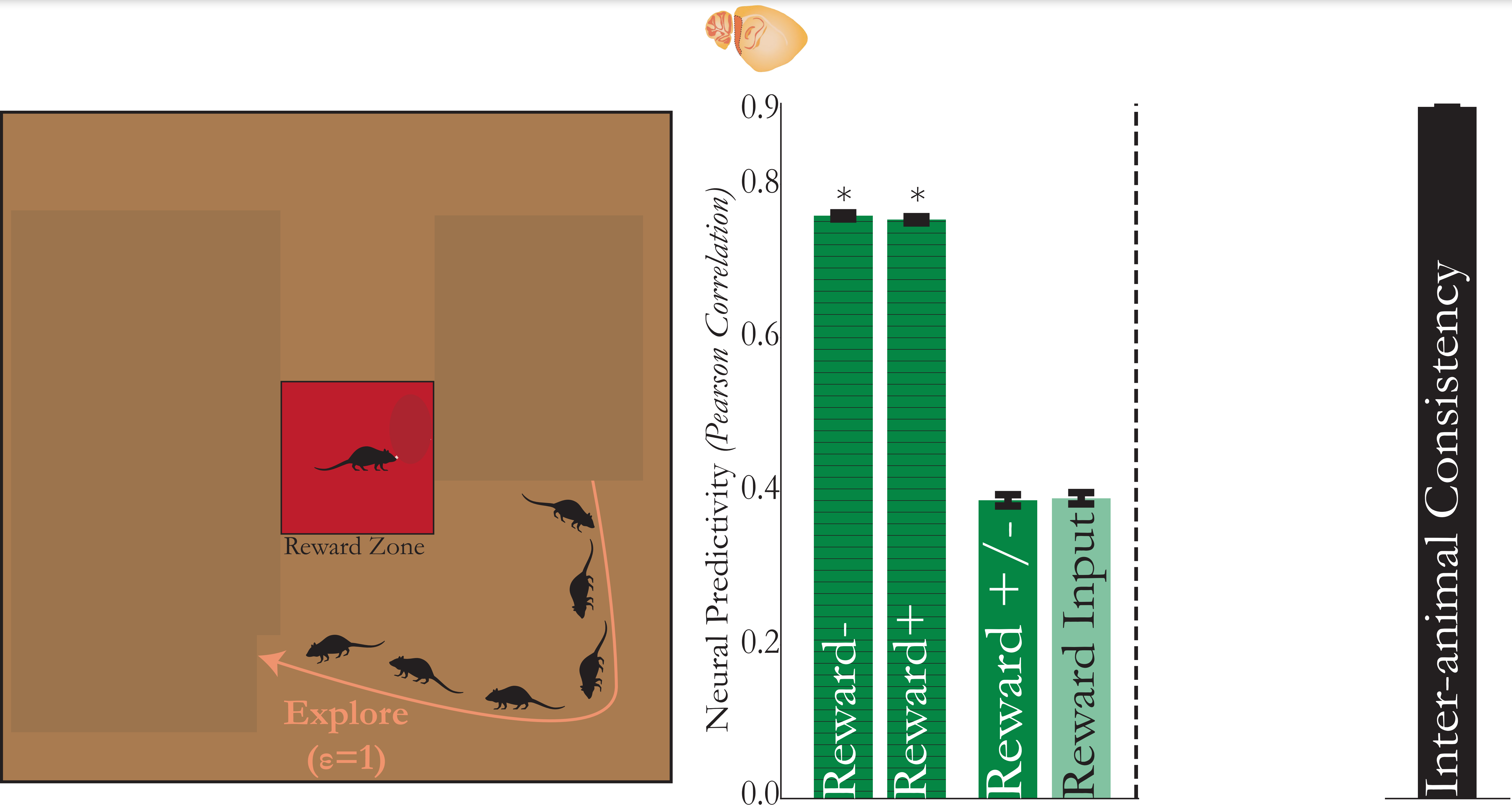
Simply augmenting inputs does not help either

Inspiration from animal behavior — rapid, direct paths

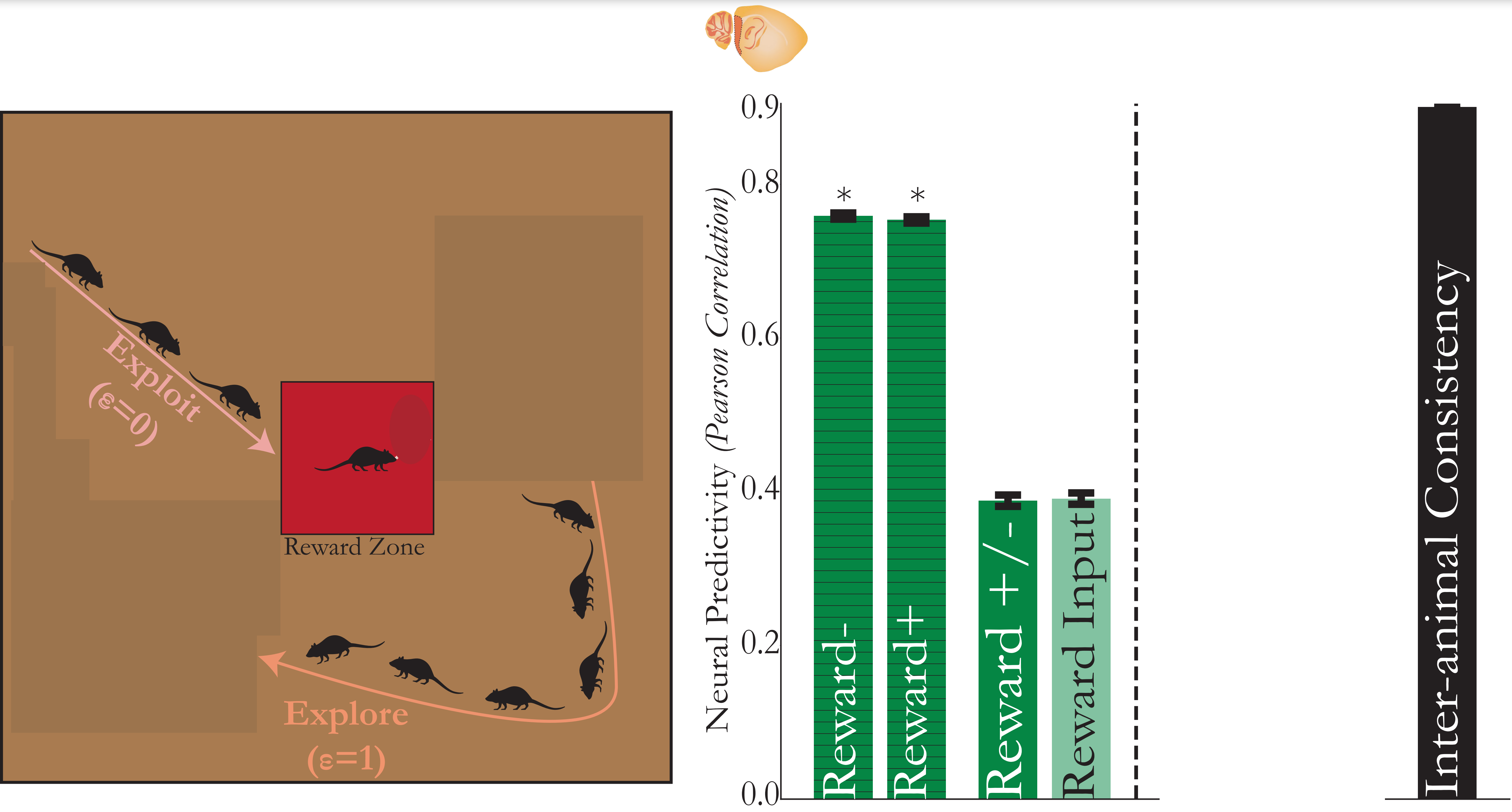
Animals tend to take rapid, direct paths to reward zone



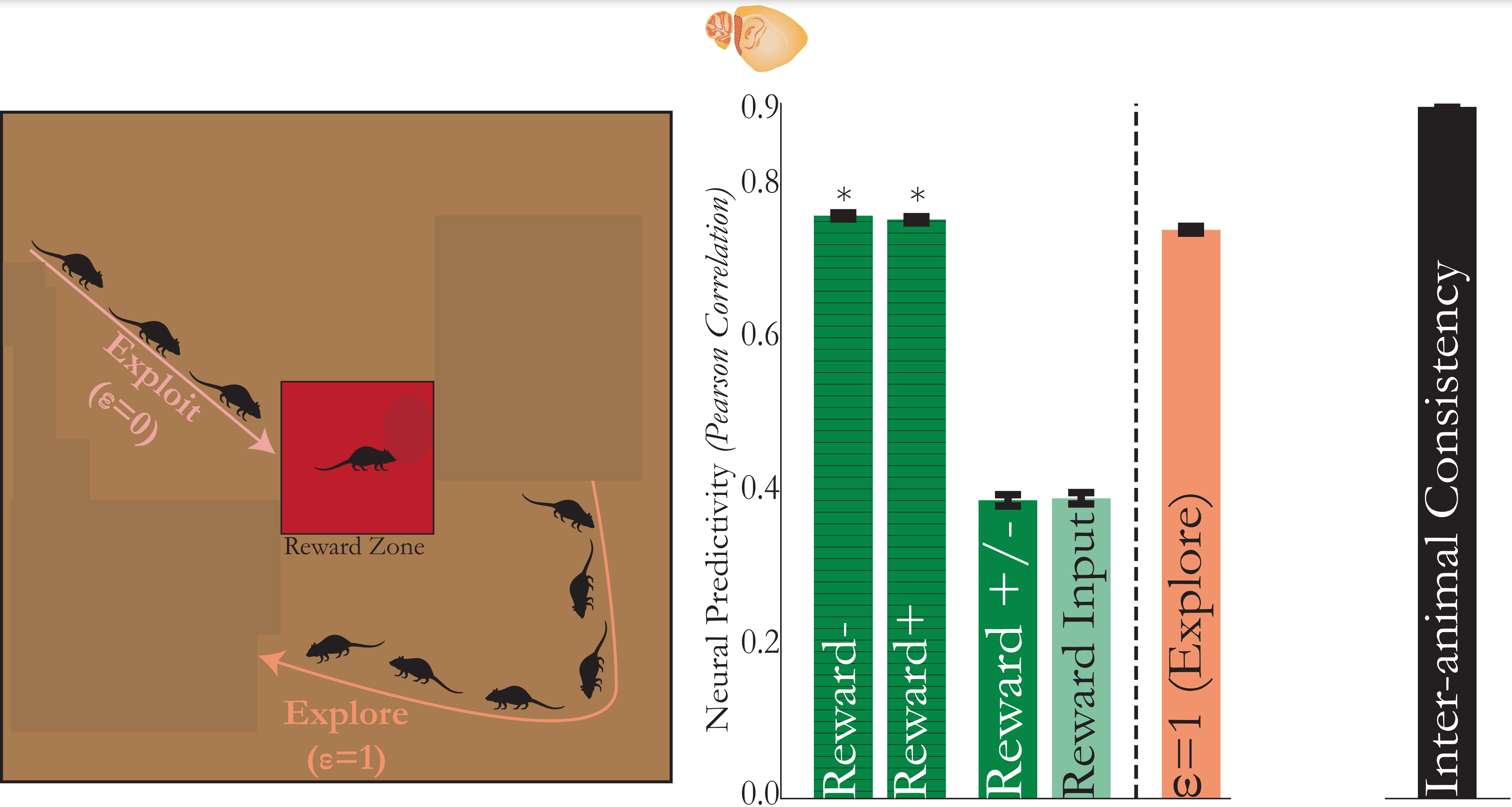
Reward must be extrinsically modeled



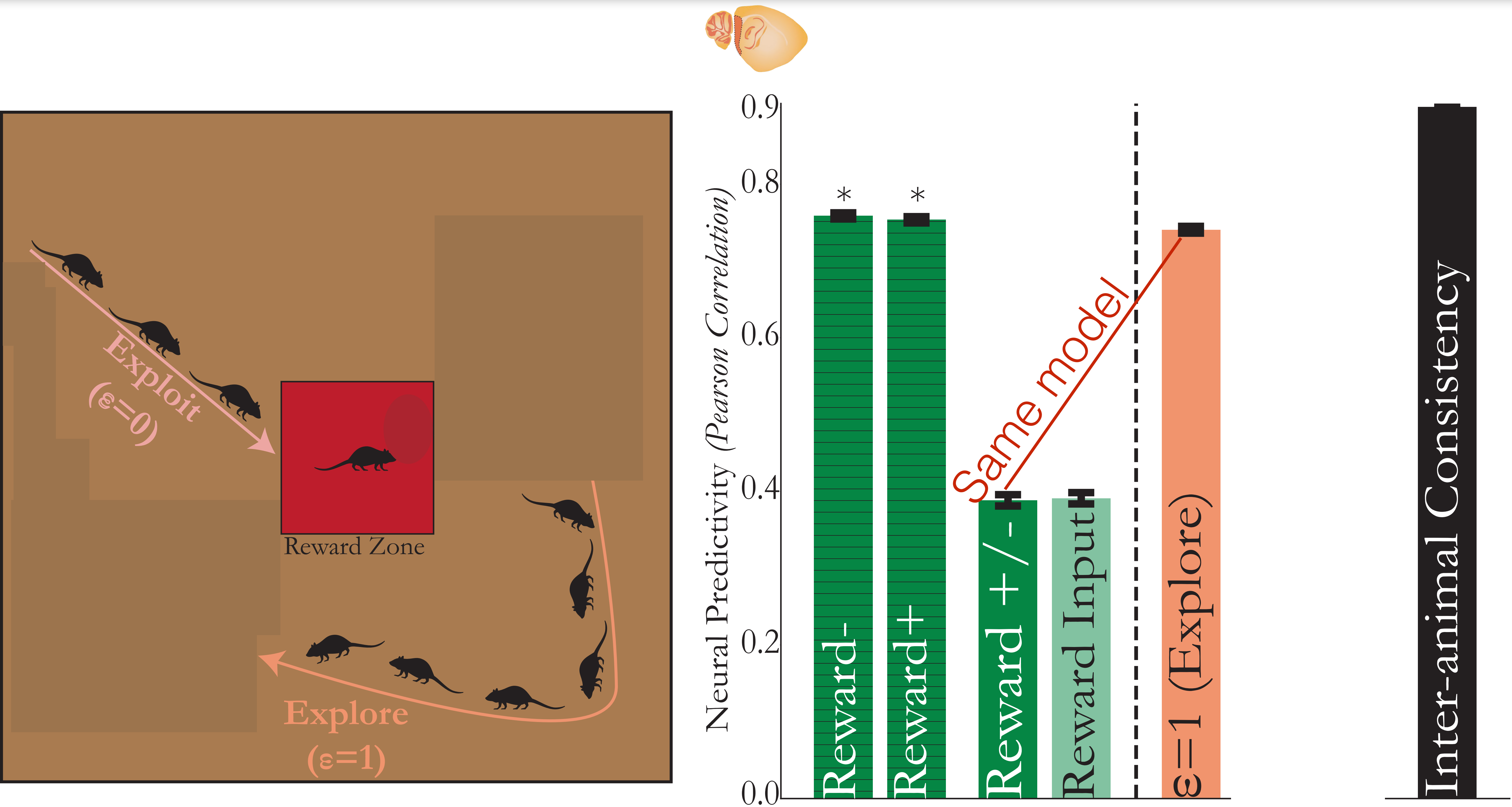
Modeling rewards as biased path integration



Modeling rewards as biased path integration

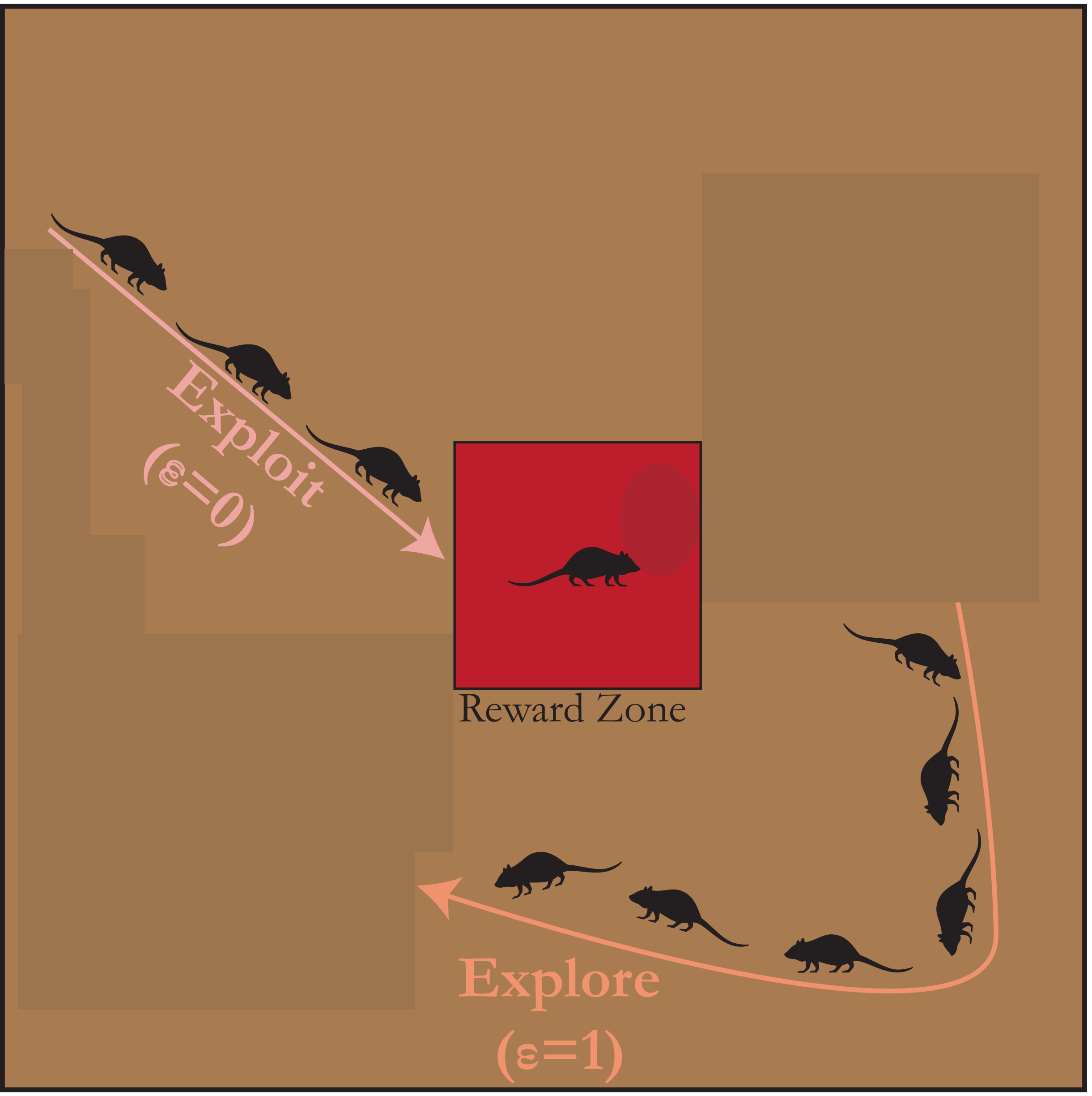


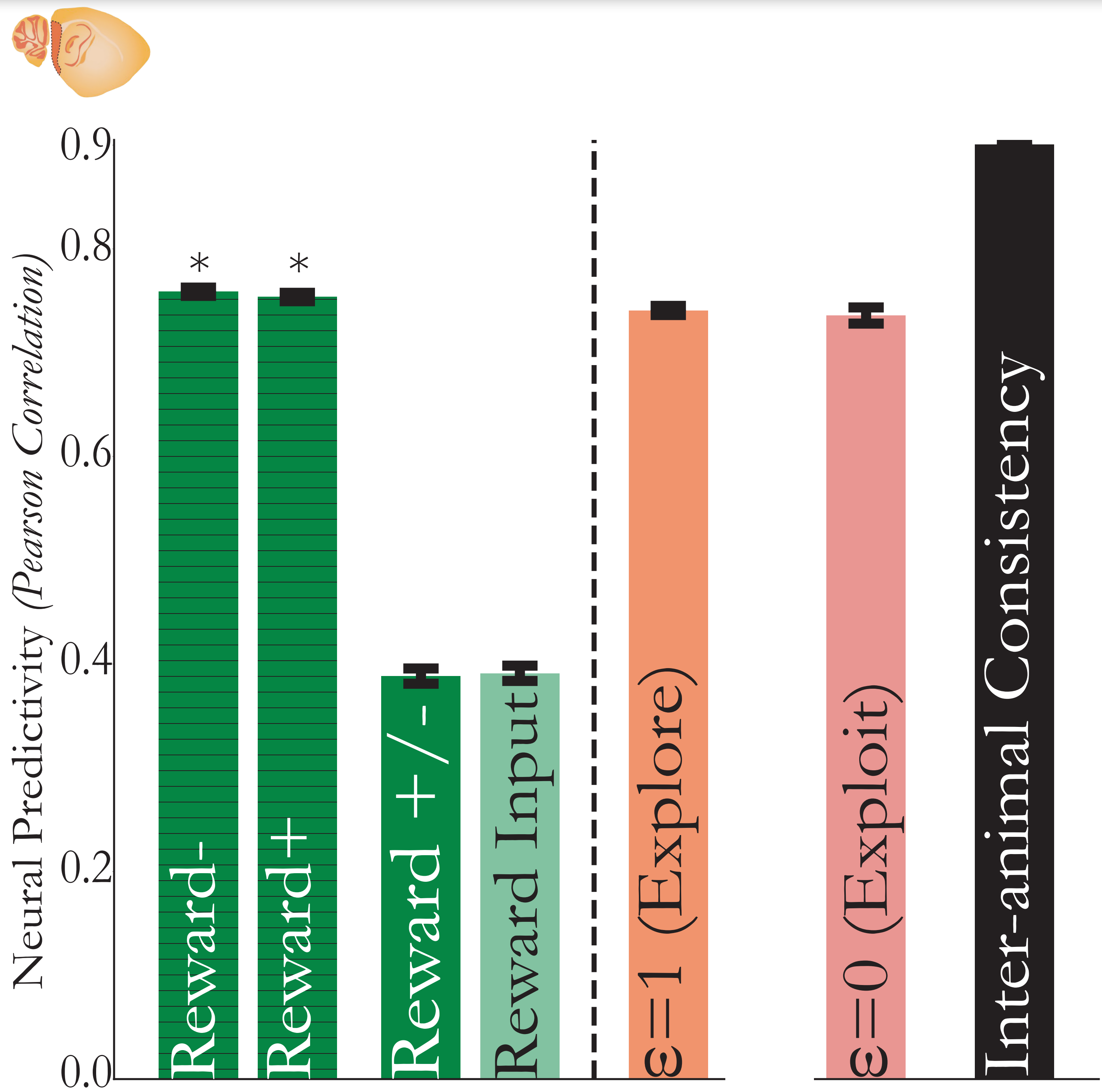
Modeling rewards as biased path integration



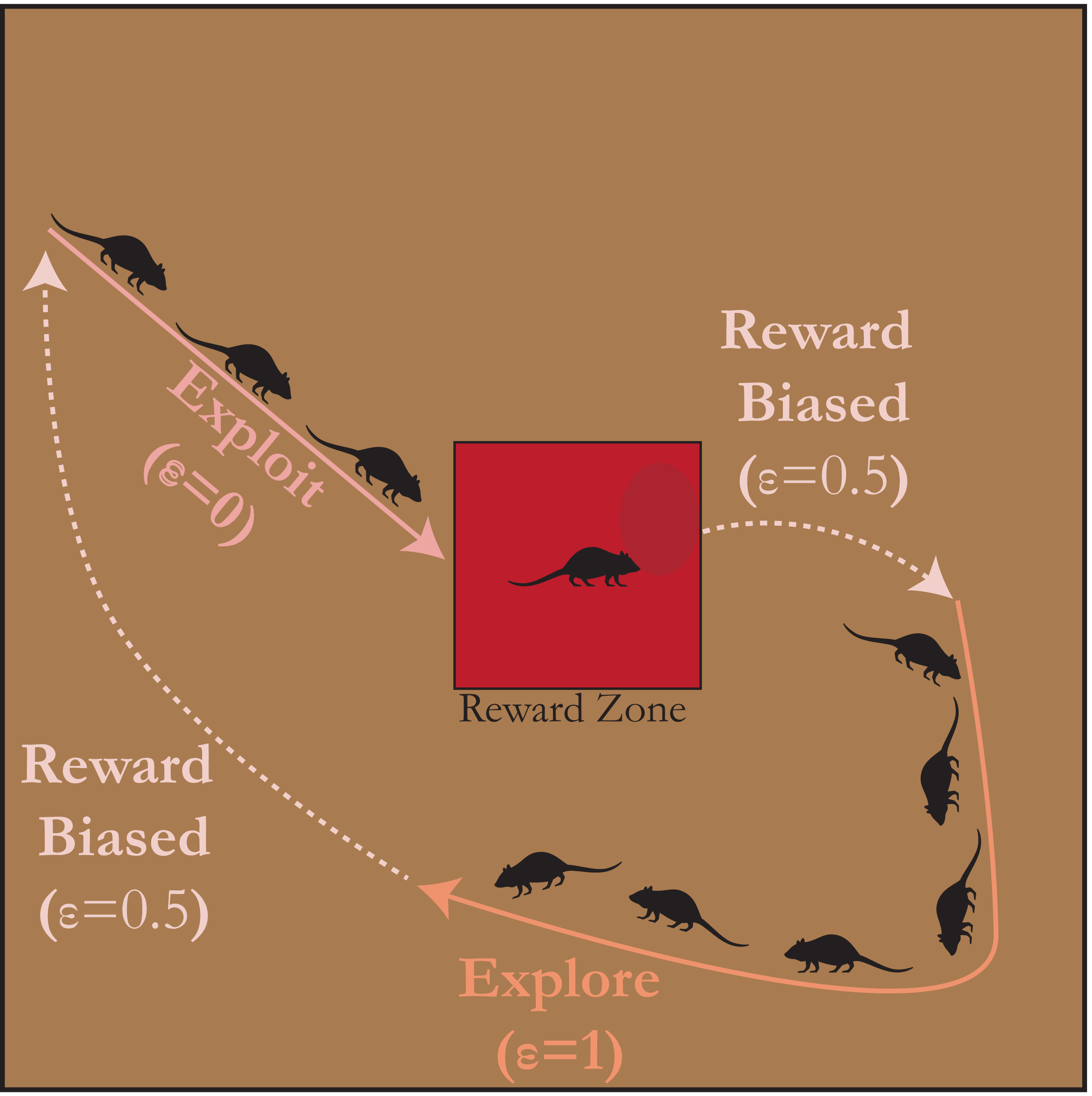
Reward remapping strongly input driven!

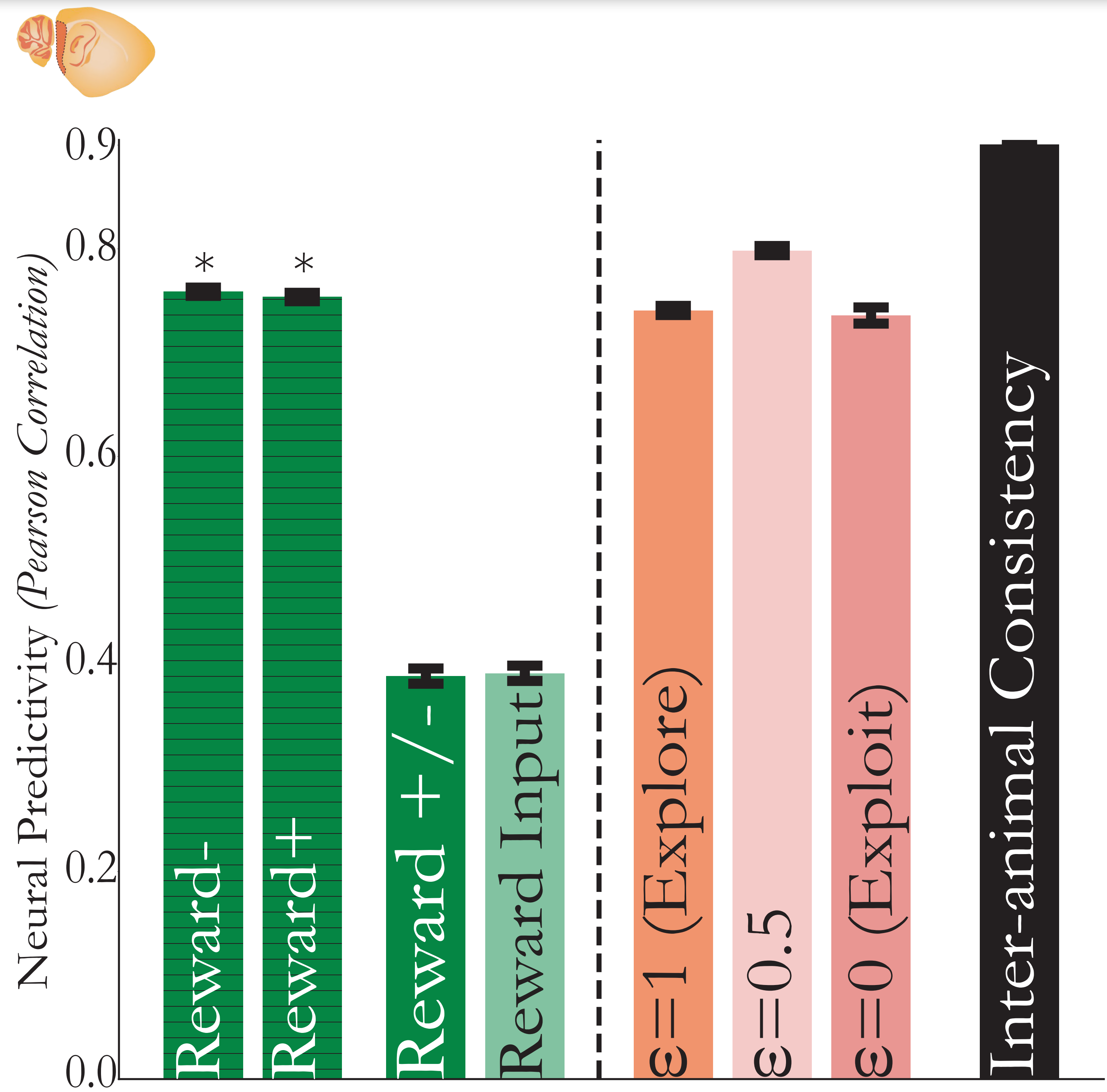
Pure exploitation isn't any better





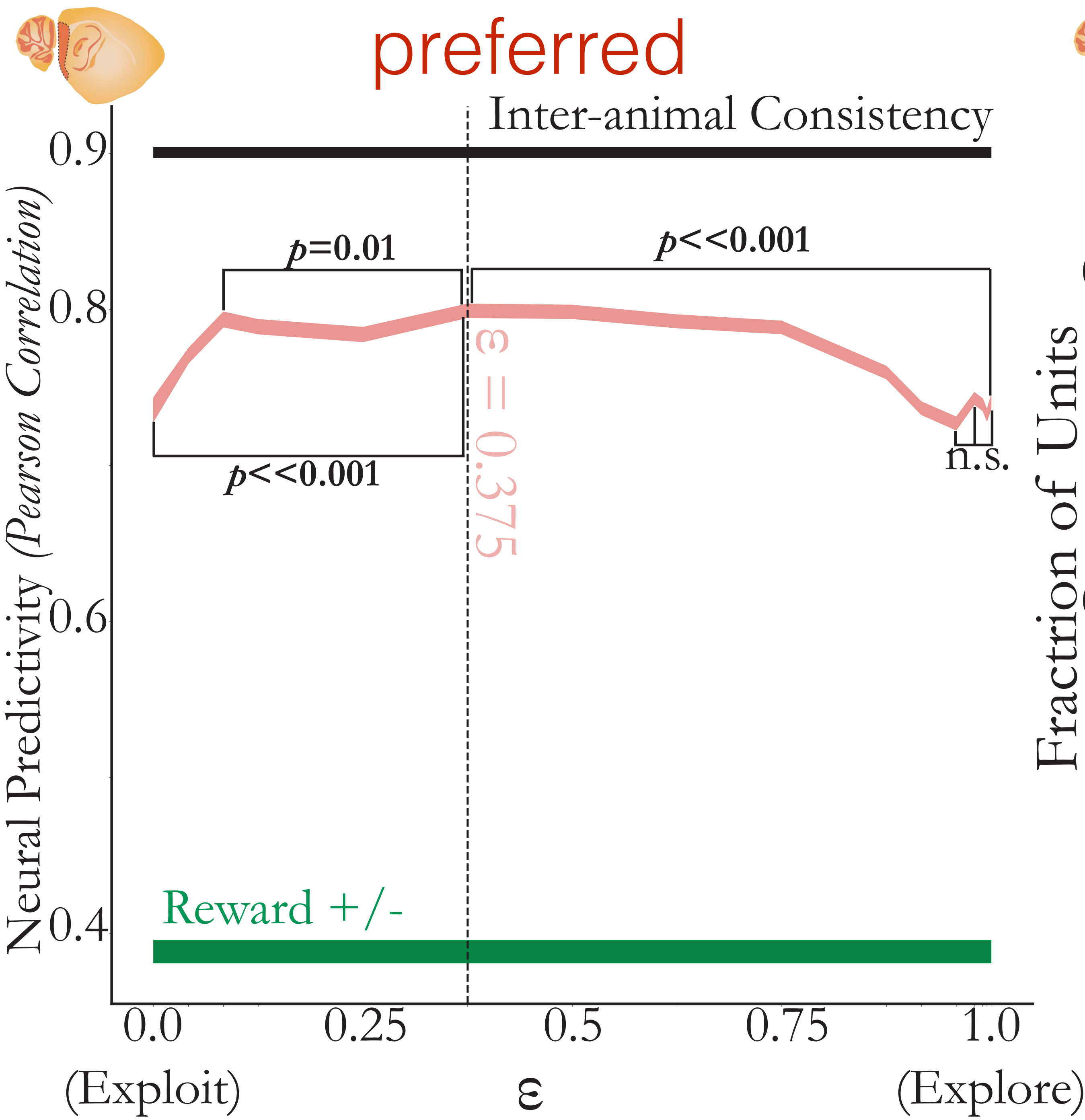
Reward-biased path integration captures remapping of responses in the presence of reward

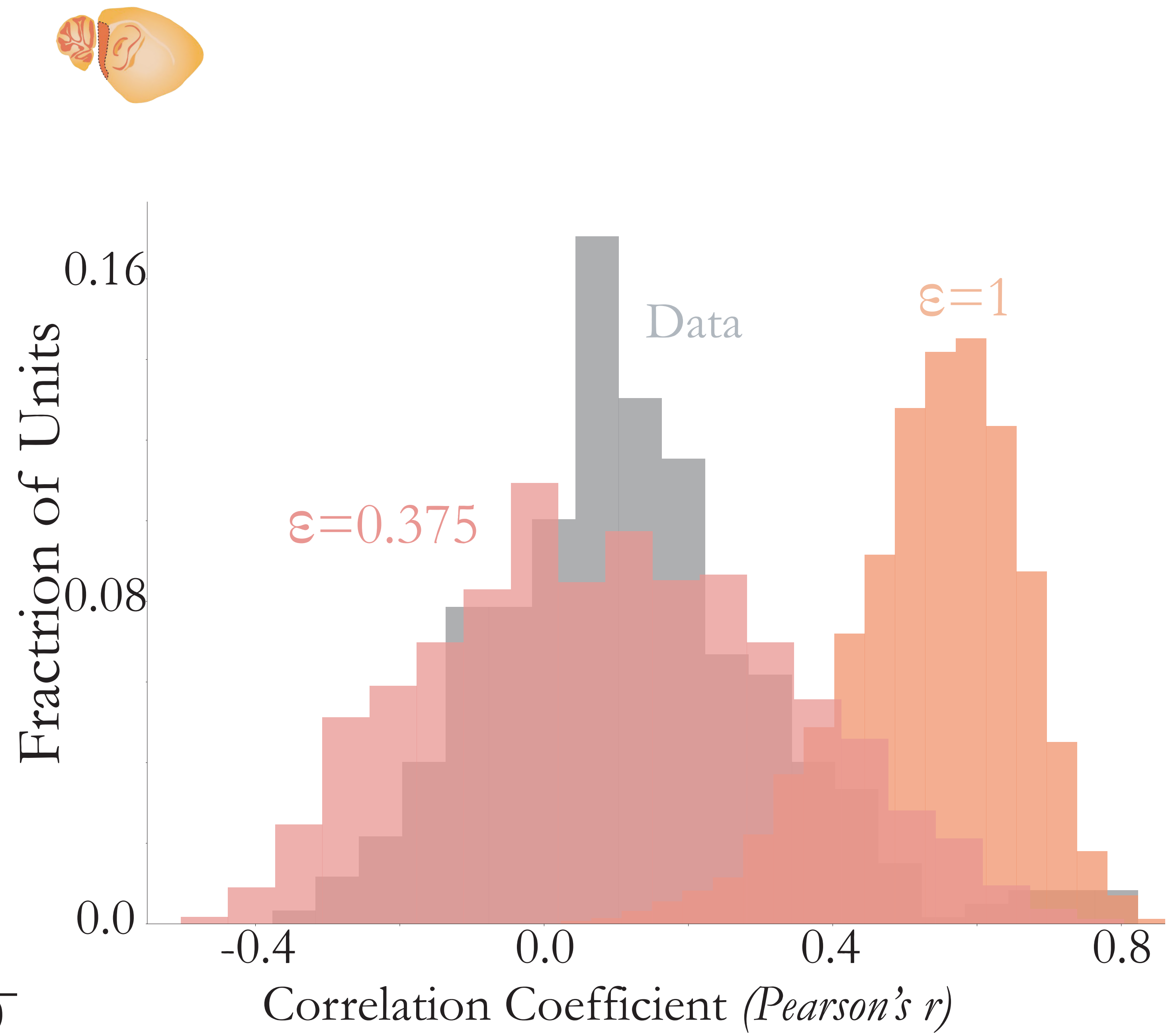




Reward-biased path integrator best captures remapping

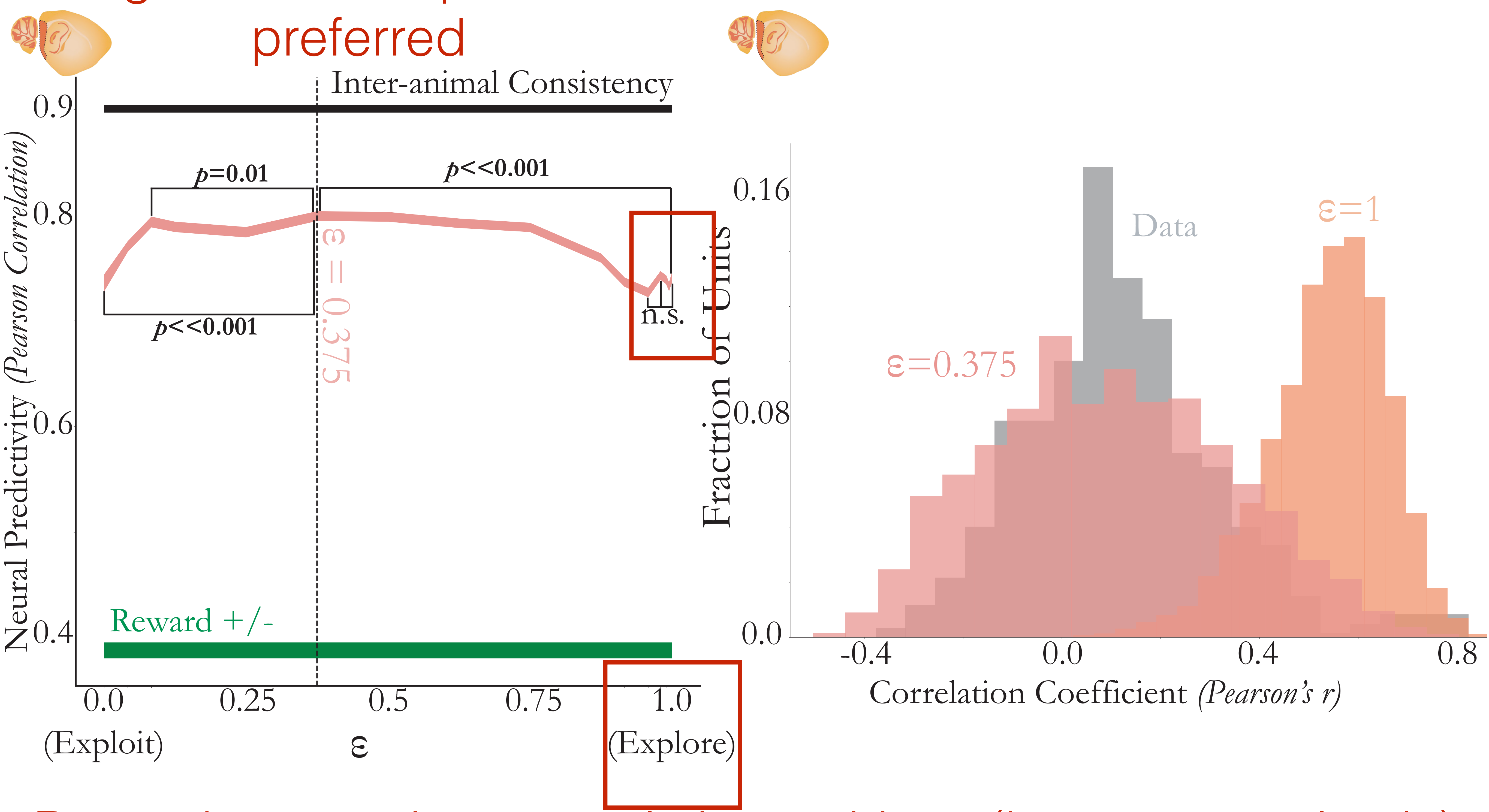
Slight bias to exploitation preferred





Reward-biased path integrator best captures remapping

Slight bias to exploitation preferred

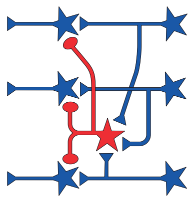


Reward remapping strongly input driven (but not completely)

Takeaways

A = *architecture class*

1. *"Circuit"*



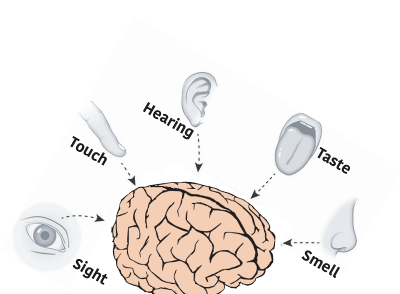
3. *"Ecological niche/behavior"*

T = *task loss*

<u>Neurobiological Puzzle(s)</u>:

1. How might we characterize what these heterogeneous cells do?

2. What functional role do these cells serve in the circuit, if any?



2. *"Environment"*

D = *data stream*

Takeaways

A *= architecture class*

1. *"Circuit"*

gating + nonnegativity

T *= task loss*

3. *"Ecological niche/behavior"*

place cell integration

~~path integration~~

<u>Neurobiological Puzzle(s)</u>:

1. How might we characterize what these heterogeneous cells do?

2. What functional role do these cells serve in the circuit, if any?

<u>*Partial* Resolution:</u>

1. <u>Characterization:</u> Close to perfect neural predictivity with the above constraints — more complex environments are needed!

2. <u>Functional Role:</u> Grid cells are not functionally unique! Both heterogeneous and grid cells arise jointly through task optimization.

2. *"Environment"*

D *= data stream*

Functions of the hippocampus

1. Behavioral inhibition theory ("slam on the breaks")

2. Memory (Milner & Scoville from HM)

3. Spatial cognition

2.5: memory as map of conceptual space.

Hippocampus as a cognitive **non-spatial** map

LETTER

doi:10.1038/nature21692

Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit

Dmitriy Aronov[1], Rhino Nevers[1] & David W. Tank[1]

"During spatial navigation, neural activity in the hippocampus and the medial entorhinal cortex (MEC) is correlated to navigational variables such as location, head direction, speed, and proximity to boundaries5. These activity patterns are thought to provide a maplike representation of physical space. However, the hippocampal–entorhinal circuit is involved not only in spatial navigation, but also in a variety of memory-guided behaviours. . . .
A conceptual framework reconciling these views is that spatial representation is just one example of a more ge mechanism for encoding continuous, task-relevant variables."

Hippocampus as a cognitive **non-spatial** map

LETTER

doi:10.1038/nature21692

Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit

Dmitriy Aronov[1], Rhino Nevers[1] & David W. Tank[1]

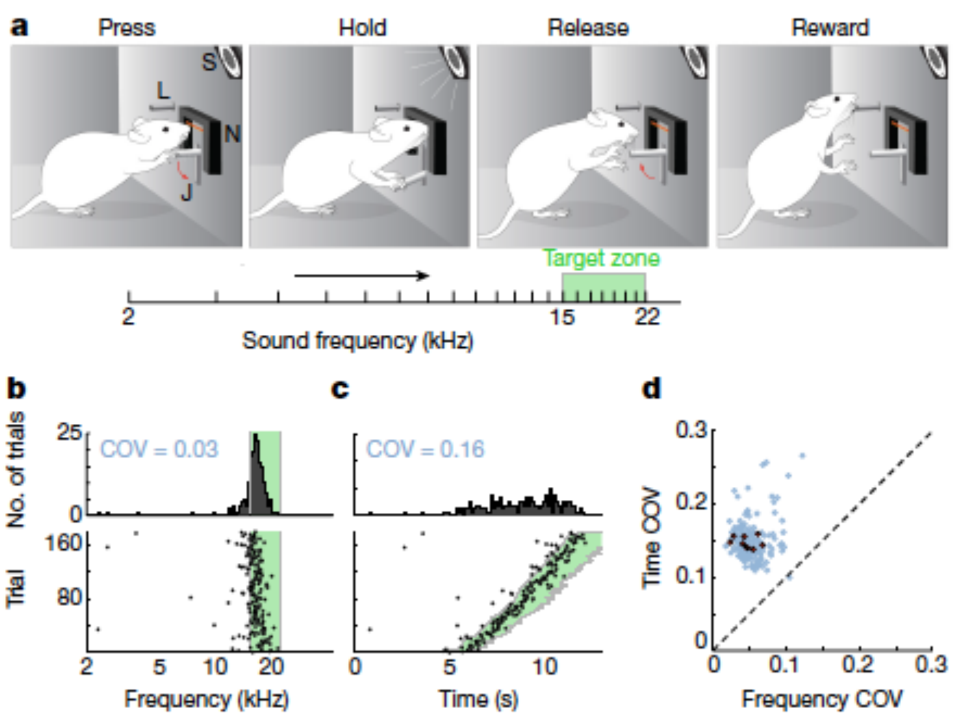


Figure 1 | Sound modulation task. a, Schematic of the SMT. Rat deflects a joystick to increase sound frequency and must release it in a target zone. J, joystick; L, lick tube; N, nosepoke; S, speaker. **b**, For a single session, frequencies at which the joystick was released on individual trials (bottom), and the distribution of these frequencies across trials (top). Most releases occurred early in the target zone (green). **c**, Same data as in **b**, but plotted as a function of time. The COV indicates a bigger spread of the distribution. **d**, COV values of frequencies and times at the joystick release across all 189 sessions from 9 rats (blue). Red circles, median values across sessions for each of the rats.

"During spatial navigation, neural activity in the hippocampus and the medial entorhinal cortex (MEC) is correlated to navigational variables such as location, head direction, speed, and proximity to boundaries5. These activity patterns are thought to provide a maplike representation of physical space. However, the hippocampal–entorhinal circuit is involved not only in spatial navigation, but also in a variety of memory-guided behaviours. . . .
<u>A conceptual framework reconciling these views is that spatial representation is just one example of a more ge mechanism for encoding continuous, task-relevant variables."</u>

Hippocampus as a cognitive **non-spatial** map

LETTER

doi:10.1038/nature21692

Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit

Dmitriy Aronov[1], Rhino Nevers[1] & David W. Tank[1]

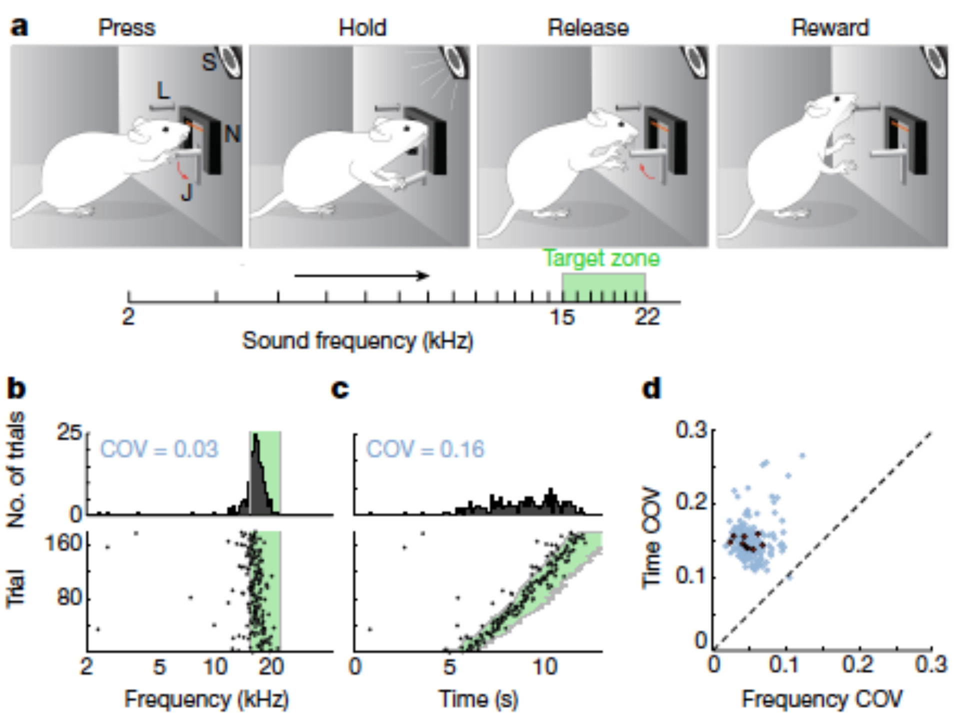


Figure 1 | Sound modulation task. a, Schematic of the SMT. Rat deflects a joystick to increase sound frequency and must release it in a target zone. J, joystick; L, lick tube; N, nosepoke; S, speaker. **b**, For a single session, frequencies at which the joystick was released on individual trials (bottom), and the distribution of these frequencies across trials (top). Most releases occurred early in the target zone (green). **c**, Same data as in **b**, but plotted as a function of time. The COV indicates a bigger spread of the distribution. **d**, COV values of frequencies and times at the joystick release across all 189 sessions from 9 rats (blue). Red circles, median values across sessions for each of the rats.