Towards an integration of deep learning and neuroscience

Adam H. Marblestone AMARBLES@MEDIA.MIT.EDU
MIT Media Lab
Cambridge, MA 02139, USA

Greg Wayne GREGWAYNE@GOOGLE.COM
Google Deepmind
London, EC4A 3TW, UK

Konrad P. Kording KK@NORTHWESTERN.EDU
Rehabilitation Institute of Chicago
Northwestern University
Chicago, IL 60611, USA

Adam Marblestone
Stanford cs379c (Tom Dean)
2017

Machine learning and neuroscience speak different languages today…

ML	Neuro
Gradient-based optimization	Circuits
Supervised learning	Representations
Augmenting neural nets with external memories	Computational motifs
	“the neural code”

Machine learning and neuroscience speak different languages today…

ML

Gradient-based optimization

Supervised learning

Augmenting neural nets with external memories

Neuro

Circuits

Representations

Computational motifs

"the neural code"

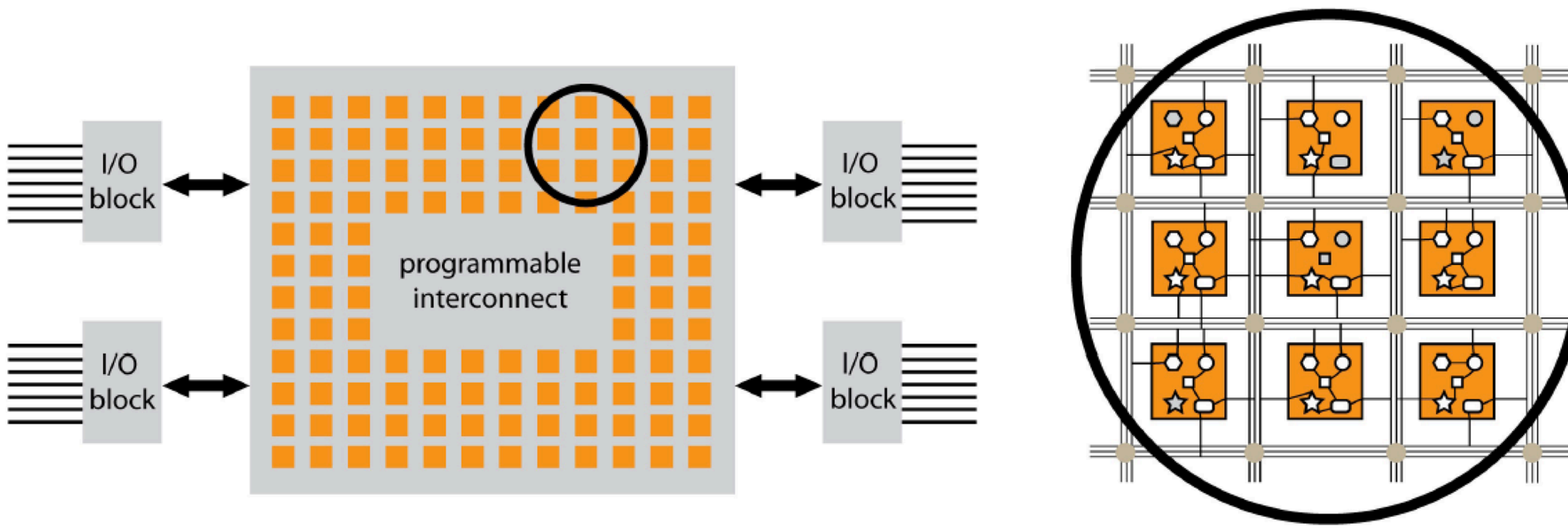
Key message:

These are not as far apart as we think

Modern ML, suitably modified, may provide a partial framework for theoretical neuro

"Atoms of computation" framework (outdated)

Apparently-uniform six-layered neocortical sheet:
common communication interface, *not* common algorithm?



The atoms of neural computation

Does the brain depend on a set of elementary, reusable computations?

By **Gary Marcus,[1] Adam Marblestone,[2] Thomas Dean[3]**

Frequently Asked Questions for: *The Atoms of Neural Computation*

Gary Marcus (NYU), Adam Marblestone (MIT), Tom Dean (Google)

“Atoms of computation” framework (outdated)

biological specializations
<>
different circuits
<>
different computations

Computation	Algorithmic/ representational realization	Neural implementation(s)	Brain location(s)
Rapid perceptual classification	Receptive fields, pooling and local contrast normalization [51,55]	Hierarchies of simple and complex cells [56]	Visual system
Complex spatiotemporal pattern recognition	Bayesian belief propagation [19,57]	Feedforward and feedback pathways in cortical hierarchy [19]	Sensory hierarchies
Learning efficient coding of inputs	Sparse coding [58]	Thresholding and local competition [59]	Sensory and other systems
Working memory	Continuous or discrete attractor states in networks [60,61]	Persistent activity in recurrent networks [62]	Prefrontal cortex
Decision making	Temporal-difference reinforcement learning algorithms [63,64]; actor-critic models [65]	Cortically implemented Bayesian inference networks combined with temporal difference reinforcement learning via the dopamine system and action selection systems in the basal ganglia [66]	Prefrontal cortex
	Winner-take-all networks [67]	Recurrent networks coupled via lateral inhibition [67]	Prefrontal cortex
Gating of information flow	Context-dependent tuning of activity in recurrent network dynamics [68]	Recurrent neural networks implementing line attractors and selection vectors [68]	Prefrontal cortex
	Shifter circuits [69]	Divergent excitatory relays and input-selective shunting inhibition in dendrites [69]	Visual system
Gain control	Divisive normalization [52]	Shunting inhibition in networks or balanced background synaptic excitation and inhibition [70]	Common across many cortical areas
Sequencing of events over time [71]	Feed-forward cascades; Serial working memories [72]	Synfire chains [73-75]; Thalamo-cortico-striatal loops [76,77]	Common across many cortical areas
Representation and transformation of variables	Population coding [78]	Time-varying firing rates of cosine-tuned neurons representing dot products with encoding vectors	Motor cortex
Variable binding	Holographic reduced representations [49,79]	Circular convolution of vectors represented by neural population codes	Cortical areas involved in sequential or symbolic processing
	Dynamic binding [80,81]	Neural synchronization [82]	

The atoms of neural computation

Does the brain depend on a set of elementary, reusable computations?

By **Gary Marcus,[1] Adam Marblestone,[2] Thomas Dean[3]**

What about this objection?

"The big, big lesson from neural networks is that there exist computational systems (artificial neural networks) for which *function only weakly relates to structure*...

A neural network needs a cost function and an optimization procedure to be fully described; and an optimized neural network's computation is more predictable from this cost function than from the dynamics or connectivity of the neurons themselves."

Greg Wayne (DeepMind) in response to Atoms of Neural Computation paper

Three hypotheses for linking neuroscience and ML

1) Existence of cost functions:
the brain optimizes cost functions (~ as powerfully as backprop)

2) Diversity of cost functions:
the cost functions are diverse, area-specific and systematically regulated in space and time
(not a single "end-to-end" training procedure)

3) Embedding within a structured architecture:
optimization occurs within a specialized architecture containing pre-structured systems (e.g., memory systems, routing systems) that support efficient optimization

Three <u>hypotheses</u> for linking neuroscience and ML

1) Existence of cost functions:

the brain optimizes cost functions (~ as powerfully as backprop)

<u>Not</u> just the trivial "neural dynamics can be *described* in terms of cost function(s)"… it actually has <u>machinery</u> to <u>do</u> optimization

2) Diversity of cost functions:

the cost functions are diverse, area-specific and systematically regulated in space and time
(not a single "end-to-end" training procedure)

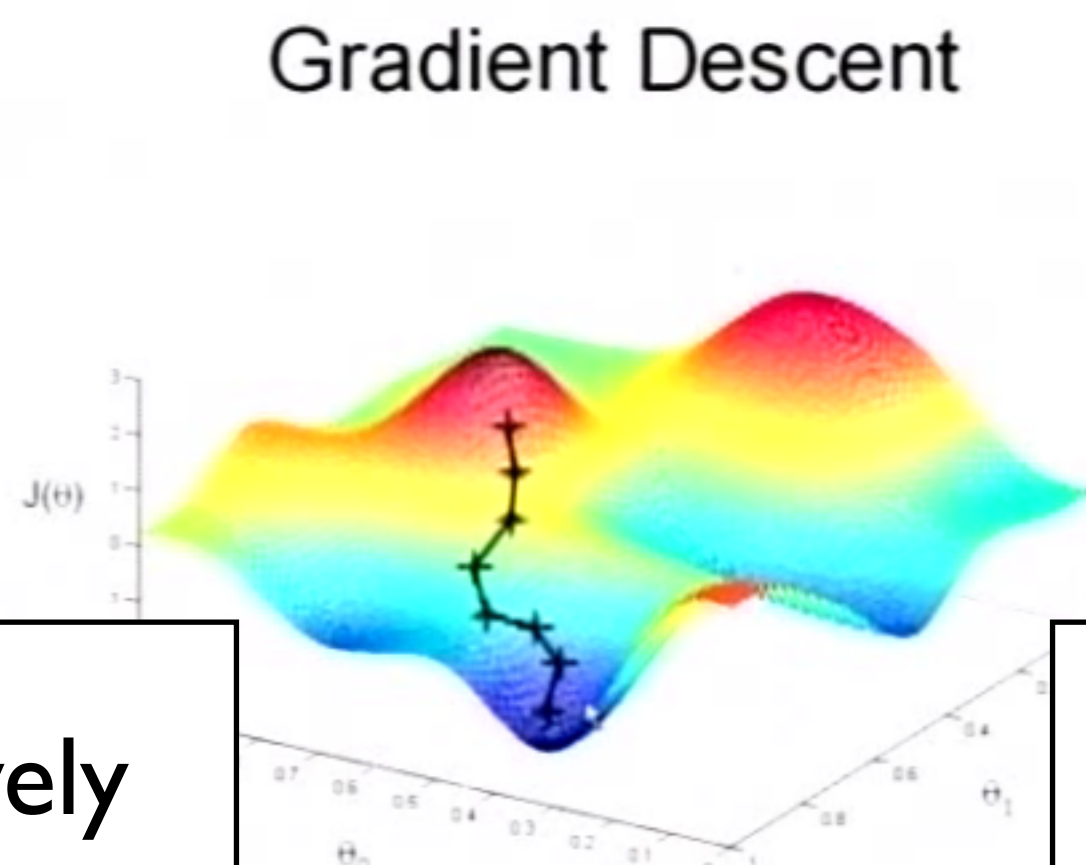
3) Embedding within a structured architecture:

optimization occurs within a specialized architecture containing pre-structured systems (e.g., memory systems, routing systems) that support efficient optimization

Three <u>hypotheses</u> for linking neuroscience and ML

1) Existence of cost functions:

the brain optimizes cost functions (~ at least as powerfully as backprop)



Relatively unstructured network → *Trained* relatively unstructured network

1) Existence of cost functions:

Ways to perform optimization in a neural network

Back-propagation	*efficient, exact gradient computation by propagating errors through multiple layers*
Node perturbation Serial Parallel	*slow, high-variance gradient computation*
Weight perturbation Serial Parallel	*slow, high-variance gradient computation*

1) Existence of cost functions:

Back-propagation is much more efficient and precise,
***but** computational neuroscience has mostly rejected it*

It has instead focused on local synaptic plasticity rules,
or occasionally on weight or node perturbation

Example:

Gradient learning in spiking neural networks by dynamic perturbation of conductances

Ila R. Fiete[1] and H. Sebastian Seung[2]

[1] *Kavli Institute for Theoretical Physics, University of California, Santa Barbara, CA 93106*

[2] *Howard Hughes Medical Institute and Department of Brain and Cognitive Sciences, M.I.T., Cambridge, MA 02139*

We present a method of estimating the gradient of an objective function with respect to the synaptic weights of a spiking neural network. The method works by measuring the fluctuations in the objective function in response to dynamic perturbation of the membrane conductances of the neurons. It is compatible with recurrent networks of conductance-based model neurons with dynamic synapses. The method can be interpreted as a biologically plausible synaptic learning rule, if the dynamic perturbations are generated by a special class of "empiric" synapses driven by random spike trains from an external source.

1) Existence of cost functions:

Neural nets and the brain

It is hardly surprising that such achievements have produced a heady sense of euphoria. But is this what the brain actually does? Alas, the back-drop nets are unrealistic in almost every respect, as indeed some of their inventors have admitted. They usually violate the rule that the outputs of a single neuron, at least in the neocortex, are either excitatory synapses or inhibitory ones, but not both[12]. It is also extremely difficult to see how neurons would implement the back-prop algorithm. Taken at its face value this seems to require the rapid transmission of information backwards along the axon, that is, antidromically from each of its synapses. It seems highly unlikely that this actually happens in the brain. Attempts to make more realistic nets to do this[13], though ingenious, seem to me to be very forced. Moreover the theorists working

NATURE VOL. 337 12 JANUARY 1989 COMMENTARY 129

The recent excitement about neural networks

Francis Crick

The remarkable properties of some recent computer algorithms for neural networks seemed to promise a fresh approach to understanding the computational properties of the brain. Unfortunately most of these neural nets are unrealistic in important respects.

1) **Existence of cost functions**:

Do you really need information to flow "backwards along the axon"?

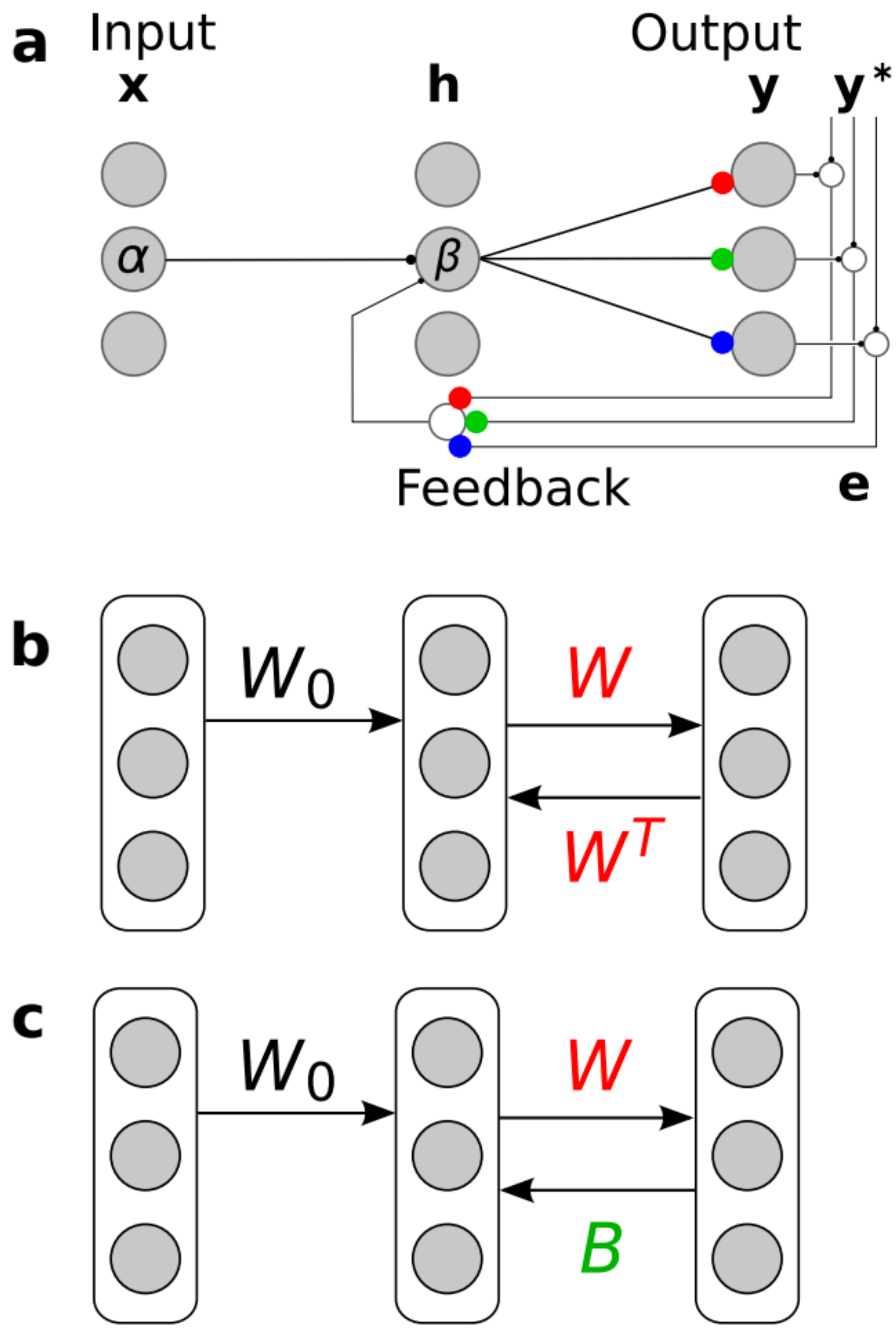
Or more generally, is the "weight transport" problem a genuine one?

I) Existence of cost functions:

Random feedback weights support learning in deep neural networks

Timothy P. Lillicrap, Daniel Cownden, Douglas B. Tweed, Colin J. Akerman

(Submitted on 2 Nov 2014)



transpose(**W**) x **e**
gets fed back
into the hidden units

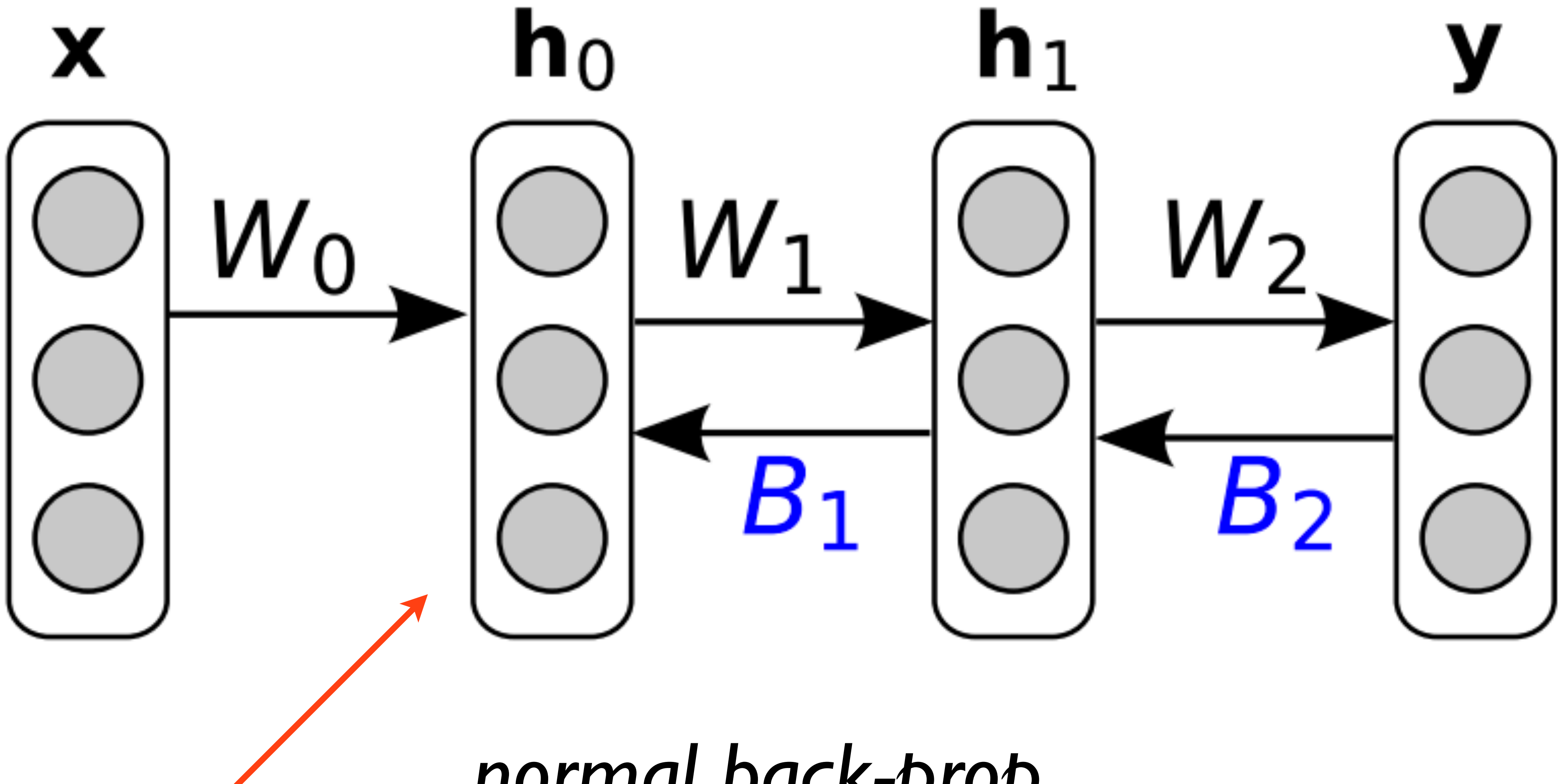
B x **e**
gets fed back
into the hidden units

I) Existence of cost functions:

Random feedback weights support learning in deep neural networks

Timothy P. Lillicrap, Daniel Cownden, Douglas B. Tweed, Colin J. Akerman

(Submitted on 2 Nov 2014)



normal back-prop

$\Delta \mathbf{h}^0_{\text{BP}} = W_1^T((W_2^T \mathbf{e}) \circ \mathbf{h}'_1)$, where $\circ$ is element-wise multiplication

fixed random feedback weights

$\Delta \mathbf{h}^0_{\text{FA}} = B_1((B_2 \mathbf{e}) \circ \mathbf{h}'_1)$, where B_1 and B_2 are random matrices

1) Existence of cost functions:

Even *spiking, recurrent* networks may be trainable using similar ideas

works. Learning in recurrent spiking networks is notoriously difficult because local changes in connectivity may have an unpredictable effect on the global dynamics. The most commonly used learning rules, such as temporal back-propagation, are not local and thus not biologically plausible. Furthermore, reproducing the Poisson-like statistics of neural responses requires the use of networks with balanced excitation and inhibition. Such balance is easily destroyed during learning. Using a top-down approach, we show how networks of integrate-and-fire neurons can learn arbitrary linear dynamical systems by feeding back their error as a feed-forward input. The network uses two types of recurrent connections: fast and slow. The fast connections learn to balance excitation and inhibition using a voltage-based plasticity rule. The slow connections are trained to minimize the error feedback using a current-based Hebbian learning rule. Importantly, the balance maintained by fast connections is crucial to ensure that global error signals are available locally in each neuron, in turn resulting in a local learning rule for the slow connections. This demonstrates that spiking networks can learn complex

Enforcing balance allows local supervised learning in spiking recurrent networks

Ralph Bourdoukan
Group For Neural Theory, ENS Paris
Rue dUlm, 29, Paris, France
ralph.bourdoukan@gmail.com

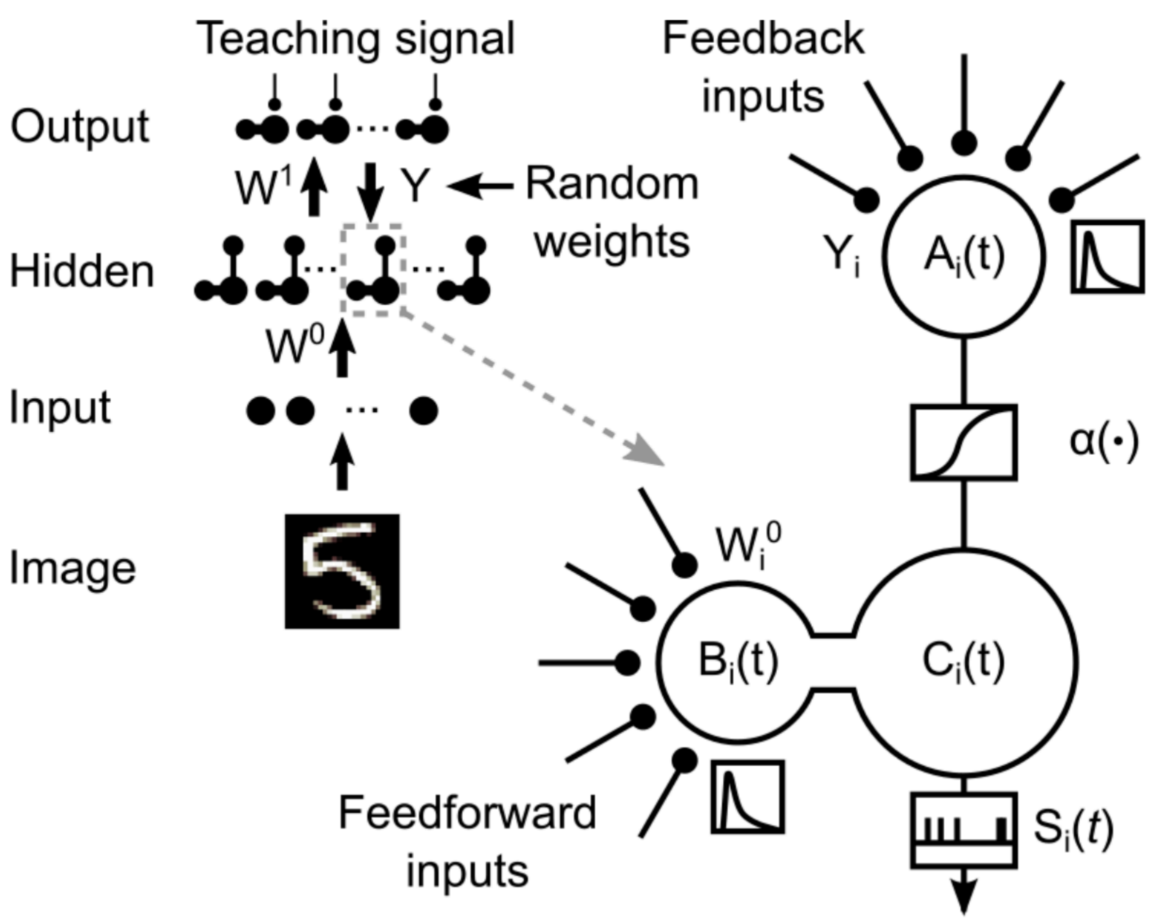
Sophie Deneve
Group For Neural Theory, ENS Paris
Rue dUlm, 29, Paris, France
sophie.deneve@ens.fr

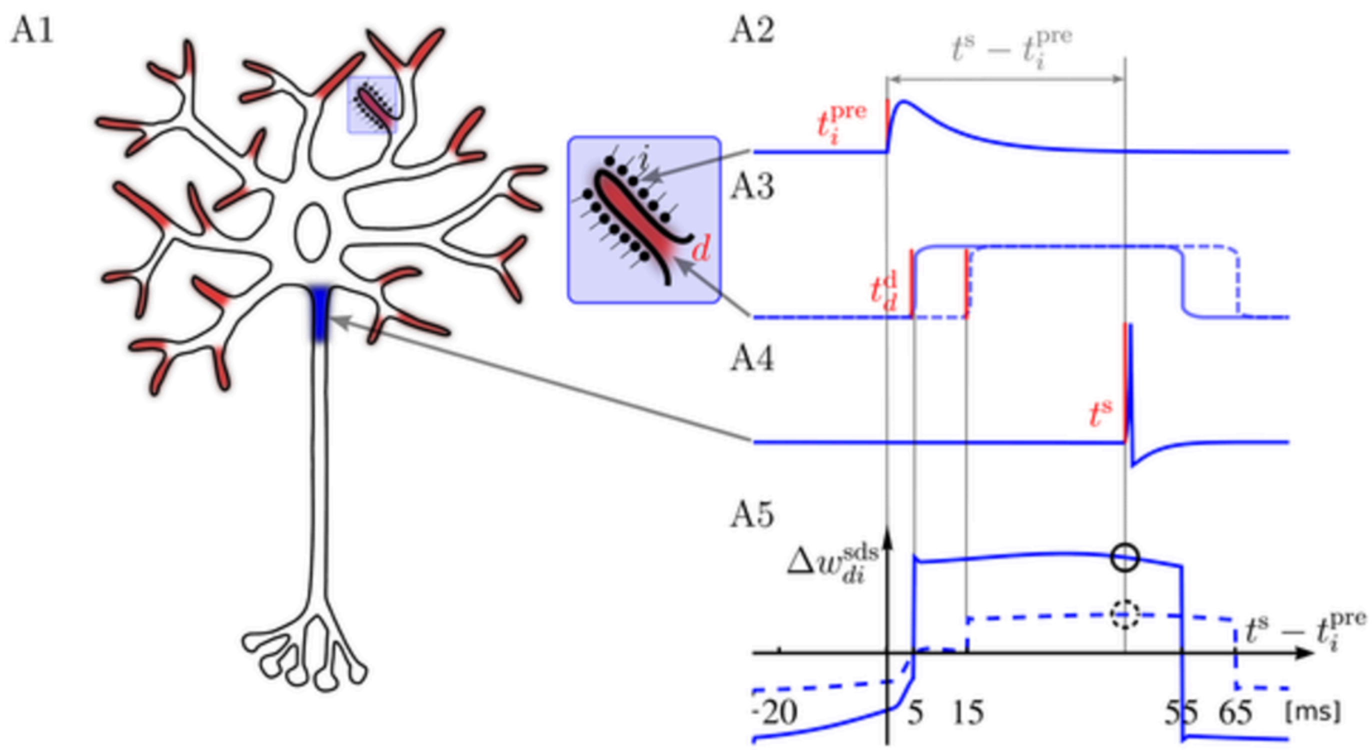
I) Existence of cost functions:

Use multiple dendritic compartments to store both "activations" and "errors"

soma voltage ~ activation

dendritic voltage ~ error derivative





Supervised and Unsupervised Learning with Two Sites of Synaptic Integration

KONRAD P. KÖRDING AND PETER KÖNIG
Institute of Neuroinformatics, ETH/UNI Zürich, Winterthurerstr. 190, 8057 Zürich, Switzerland

Deep learning with segregated dendrites

Jordan Guergiev[1,2], Timothy P. Lillicrap[4], Blake A. Richards[1,2,3,*]

Somato-dendritic Synaptic Plasticity and Error-backpropagation in Active Dendrites

Mathieu Schiess, Robert Urbanczik, Walter Senn

1) Existence of cost functions:

Or use temporal properties of the neuron to encode the signal

firing rate ~ activation

d(firing rate)/dt ~ error derivative

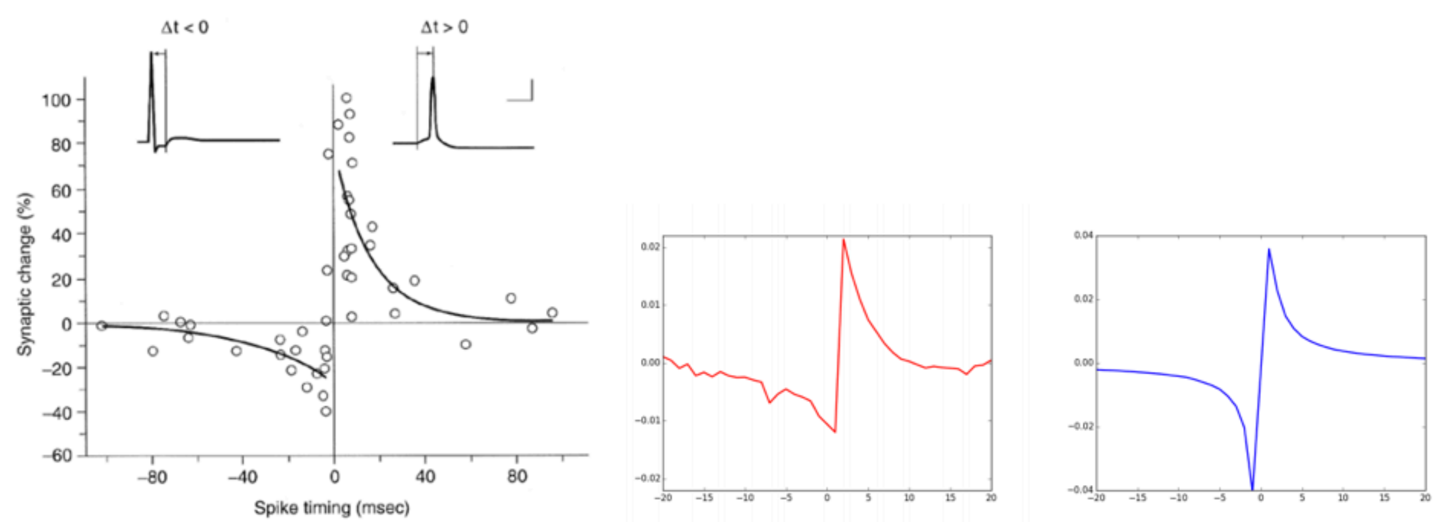


Figure 1: Left: Biological observation of STDP weight change, vertical axis, for different spike timing differences (post minus pre), horizontal axis. From Shepherd (2003), with data from Bi and Poo (2001). Compare with the result of the simulations using the objective function proposed here (middle).
Middle and right:Spike-based simulation shows that when weight updates follow SGD on the proposed predictive objective function, we recover the biologically observed relationship between spike timing difference (horizontal axis, postsynaptic spike time minus presynaptic spike time) and the weight update (vertical axis). Middle: the weight updates are obtained with the proposed update rule (Eq. 1). Right: the weight updates are obtained using the nearest neighbor STDP rule. Compare with the biological finding, left.

STDP as presynaptic activity times rate of change of postsynaptic activity

Yoshua Bengio, Thomas Mesnard, Asja Fischer, Saizheng Zhang, Yuhuai Wu

See also similar claims by Hinton

1) **Existence of cost functions**:

But isn't gradient descent only compatible with "supervised" learning?

1) **Existence of cost functions**:

But isn't gradient descent only compatible with "supervised" learning?

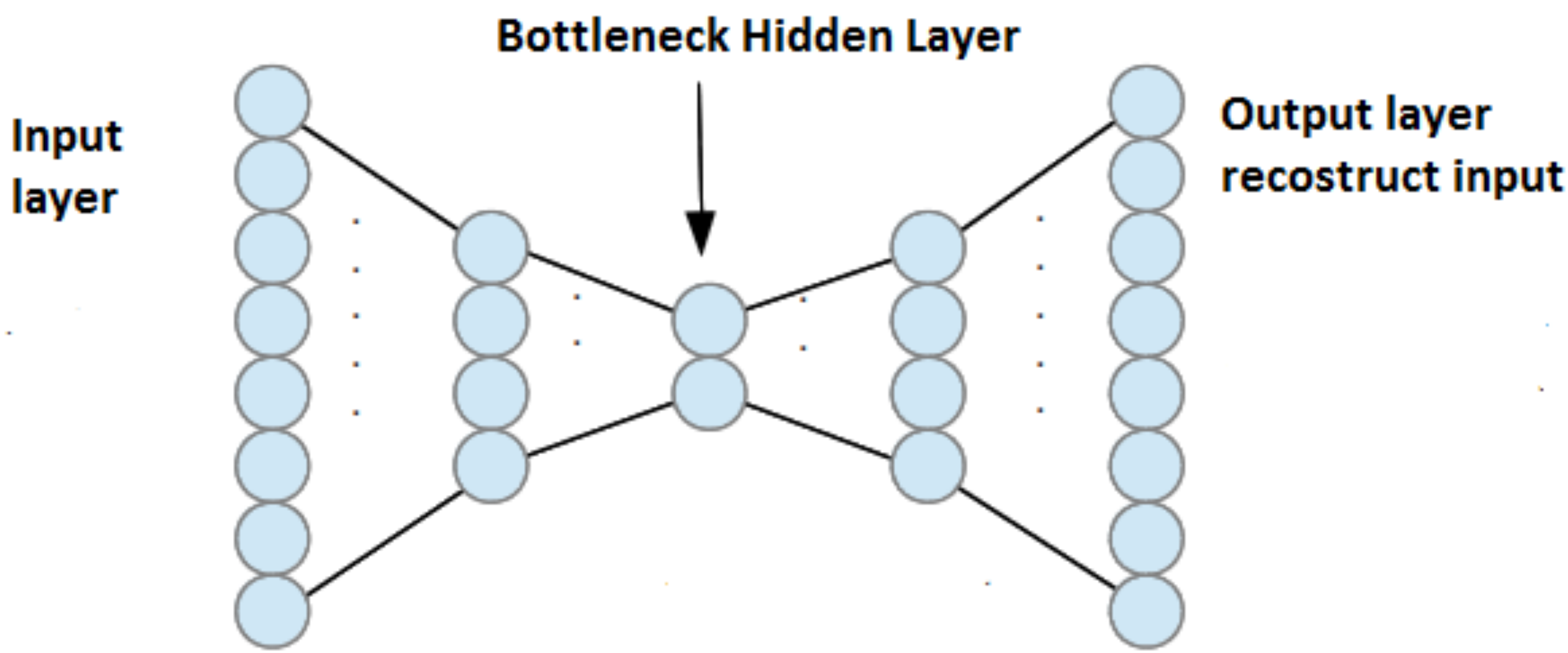
No! Lots of unsupervised learning paradigms operate via gradient descent…

1) **Existence of cost functions**:

But isn't gradient descent only compatible with "supervised" learning?

No! Lots of unsupervised learning paradigms operate via gradient descent…

classic auto-encoder



1) Existence of cost functions:

But isn't gradient descent only compatible with "supervised" learning?

No! Lots of unsupervised learning paradigms operate via gradient descent…

filling in

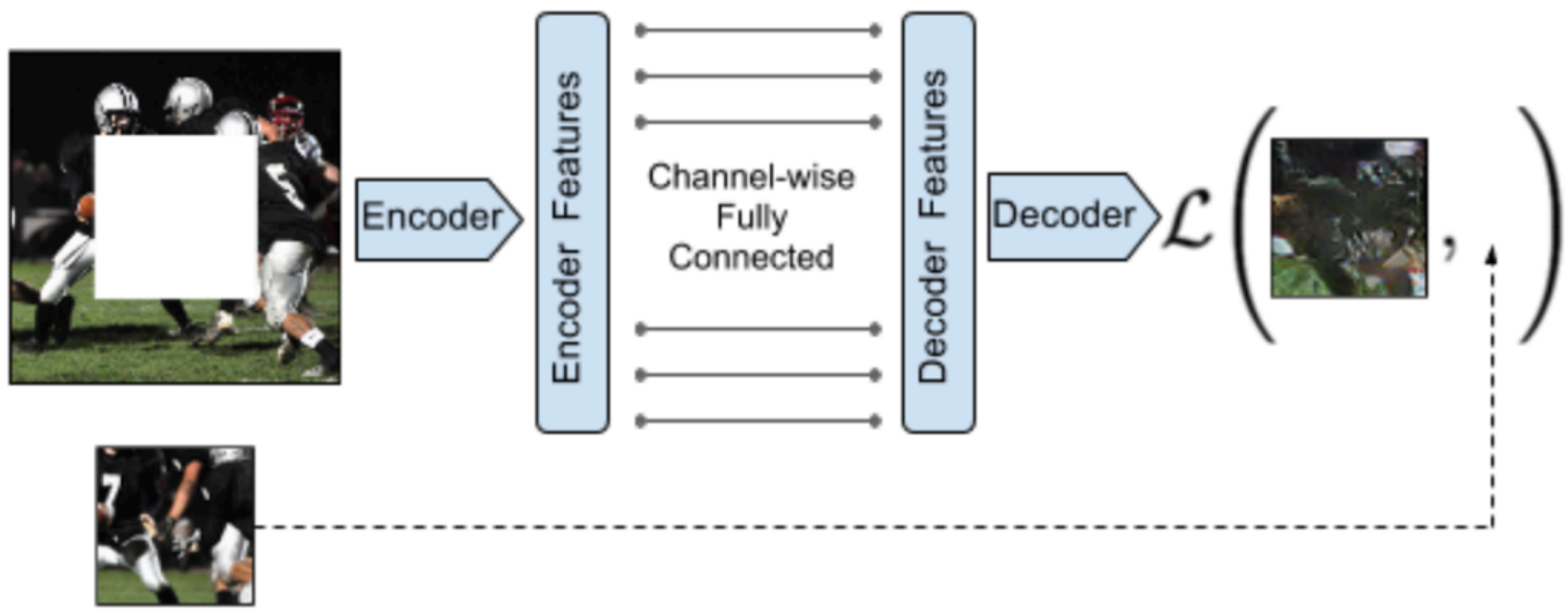


Figure 2: Context Encoder. The context image is passed through the encoder to obtain features which are connected to the decoder using channel-wise fully-connected layer as described in Section 3.1. The decoder then produces the missing regions in the image.

Context Encoders: Feature Learning by Inpainting

Deepak Pathak Philipp Krähenbühl Jeff Donahue Trevor Darrell Alexei A. Efros
University of California, Berkeley
{pathak,philkr,jdonahue,trevor,efros}@cs.berkeley.edu

I) **Existence of cost functions**:

But isn't gradient descent only compatible with "supervised" learning?

No! Lots of unsupervised learning paradigms operate via gradient descent…

prediction of the next frame of a movie

UNDERSTANDING VISUAL CONCEPTS WITH CONTINUATION LEARNING

William F. Whitney, Michael Chang, Tejas Kulkarni, and Joshua B. Tenenbaum
Department of Brain and Cognitive Science
Massachusetts Institute of Technology
{wwhitney,mbchang,tejask,jbt}@mit.edu

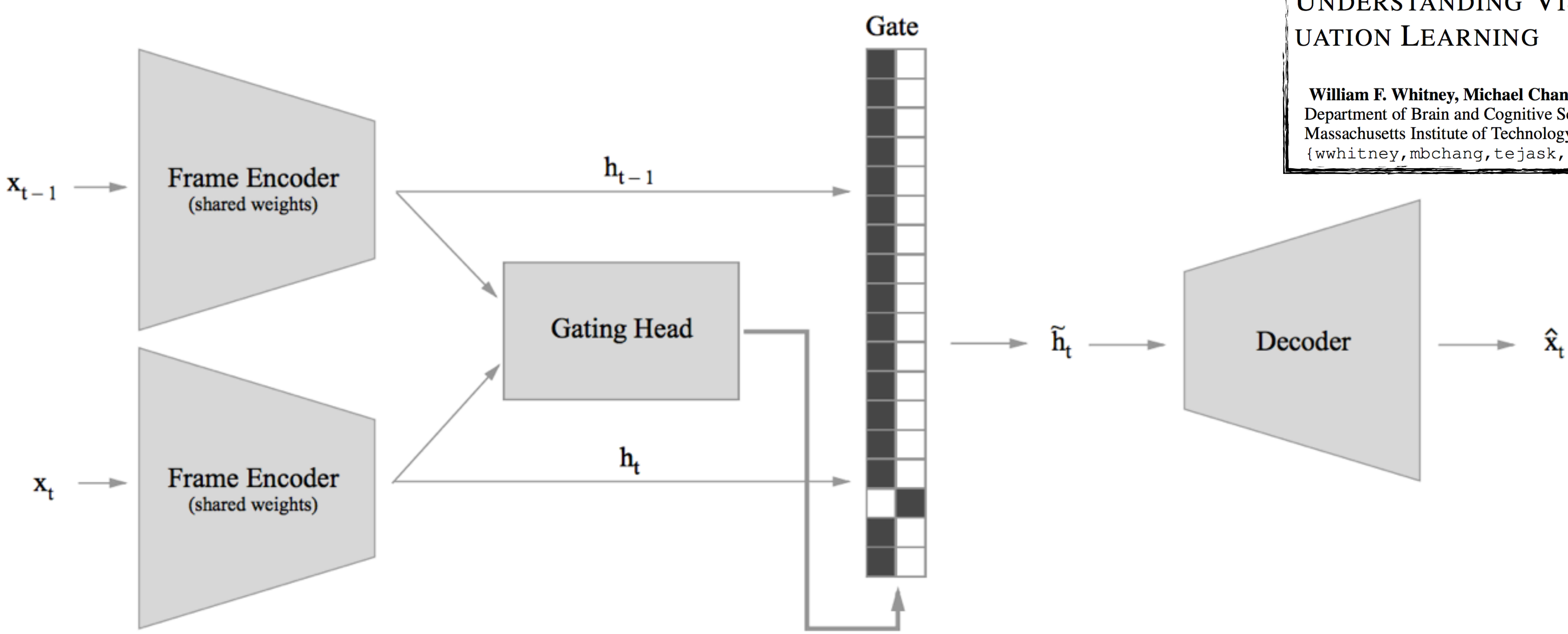


Figure 1: The gated model. Each frame encoder produces a representation from its input. The gating head examines both these representations, then picks one component from the encoding of time t to pass through the gate. All other components of the hidden representation are from the encoding of time $t-1$. As a result, each frame encoder predicts what it can about the next frame and encodes the "unpredictable" parts of the frame into one component.

1) Existence of cost functions:

But isn't gradient descent only compatible with "supervised" learning?

No! Lots of unsupervised learning paradigms operate via gradient descent…

prediction of the next frame of a movie

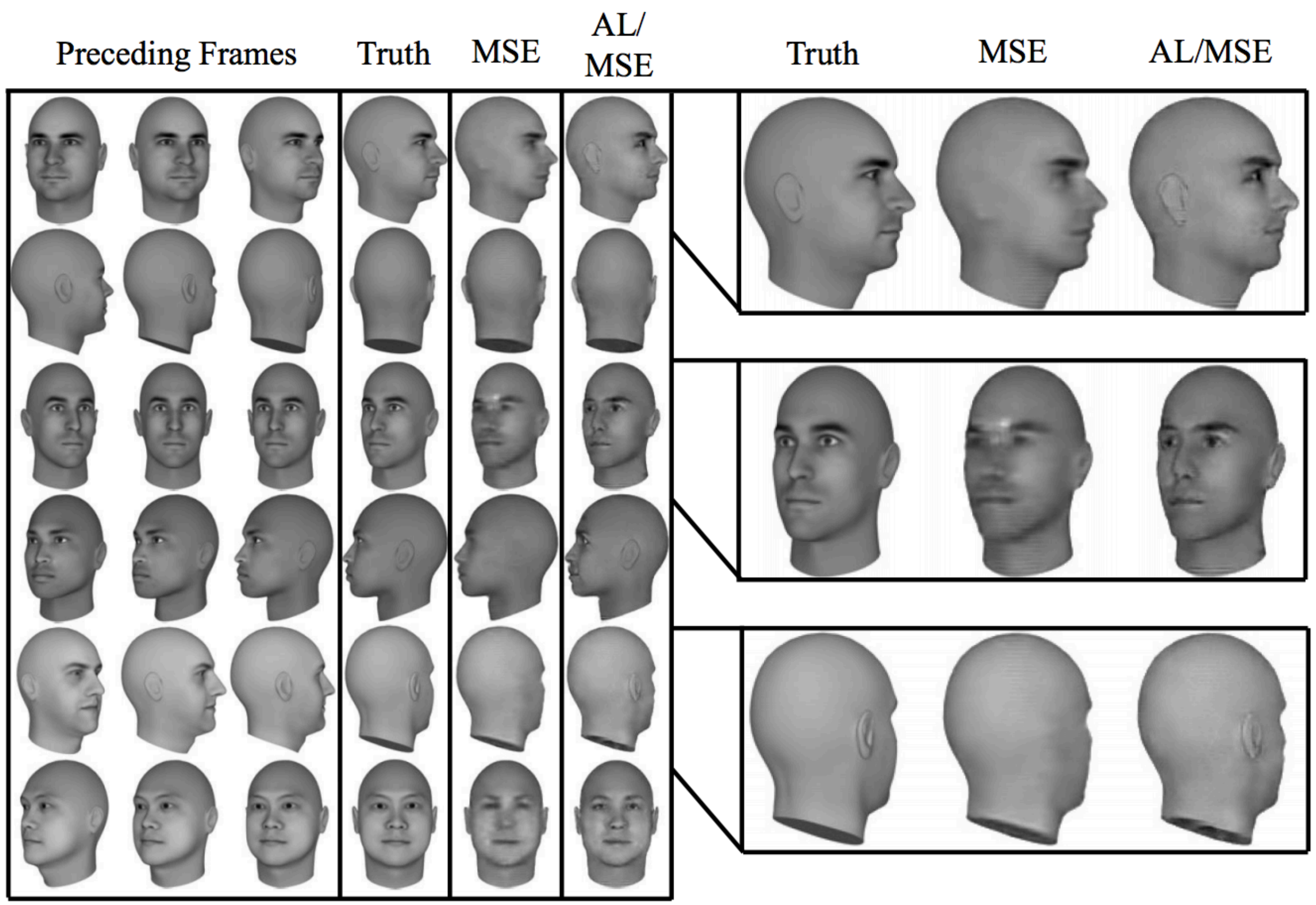


Figure 3: Example predictions for the rotating faces dataset. Predictions for models trained with MSE and a weighted MSE and adversarial loss (AL) are shown.

Unsupervised Learning of Visual Structure using Predictive Generative Networks

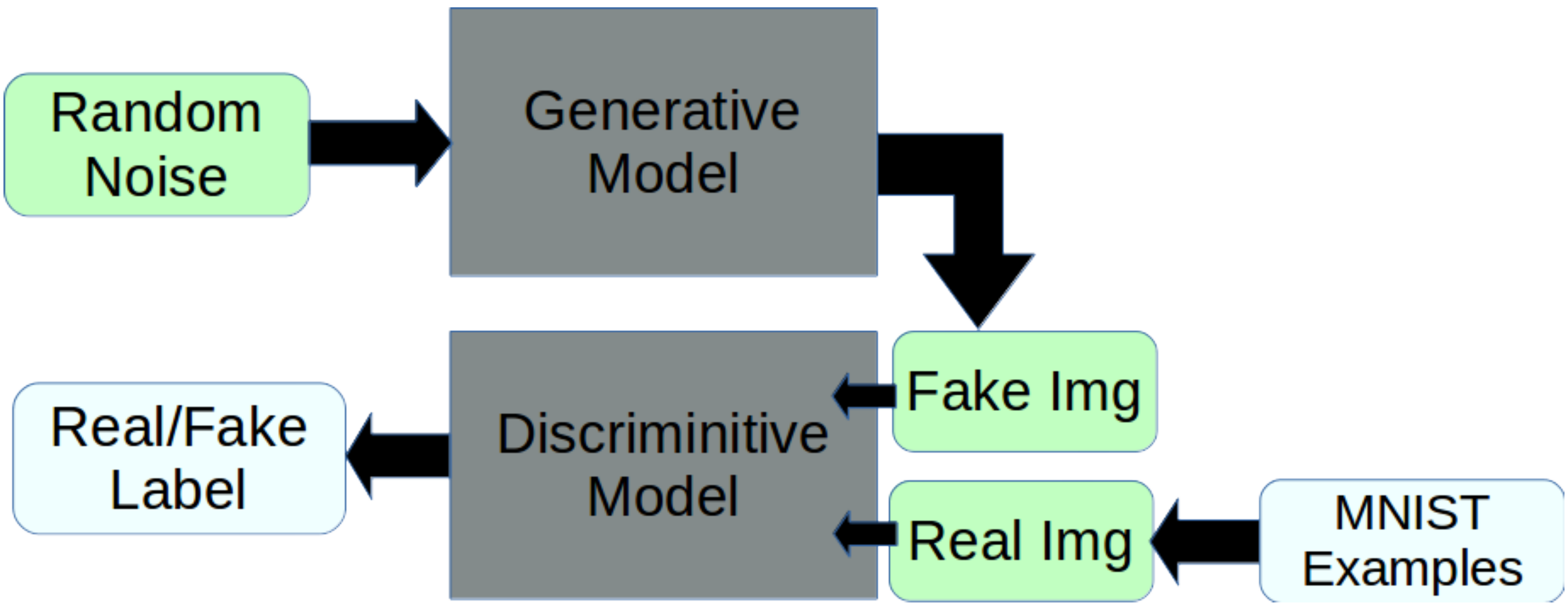
William Lotter, Gabriel Kreiman, David Cox

1) Existence of cost functions:

But isn't gradient descent only compatible with "supervised" learning?

No! Lots of unsupervised learning paradigms operate via gradient descent…

generative adversarial network



1) Existence of cost functions:

Signatures of error signals being computed in the visual hierarchy?!

Here, to disambiguate between the different types of inference that can be implemented in a hierarchy through feedforward, lateral, or feedback connections, we measured the temporal dynamics of neural signals across three stages of face processing in macaque IT. We found that many neurons in the intermediate processing stages reversed their initial preference – they rapidly switched from face preferring to anti-face preferring. Standard feedforward models including those employing local recurrences such as adaptation, lateral inhibition, and normalization could not fully capture the dynamics of face selectivity in our data. Instead, our modeling revealed that the reversals of face selectivity in intermediate processing stages are a natural dynamical signature of a family of hierarchical models that use feedback connections to implement "error coding." We interpret this very good fit to our data as evidence that the ventral stream is implementing a model in this family. If correct, this model family informs us that we should not interpret neural spiking activity at each level of the ventral stream as only an explicit representation of the variables of interest (e.g. is a face present?) but should interpret much of spiking activity as representing the necessary layer-wise errors propagated across the hierarchy. In addition, we find that, without additional parameter modifications, the same error coding hierarchical model family explains seemingly disparate IT neural response phenomena, hence unifying our results with previous findings under a single computational framework.

Evidence that the ventral stream codes the errors used in hierarchical inference and learning

Elias B. Issa, Charles F. Cadieu, James J. DiCarlo

1) Existence of cost functions:

Take Away

The brain *could* efficiently compute approximate gradients of its multi-layer weight matrix via propagating credit through multiple layers of neurons.

Diverse *potential* mechanisms available.

Such a core capability for error-driven learning could underpin diverse supervised and unsupervised learning paradigms.

1) **Existence of cost functions**:

Key Research Questions

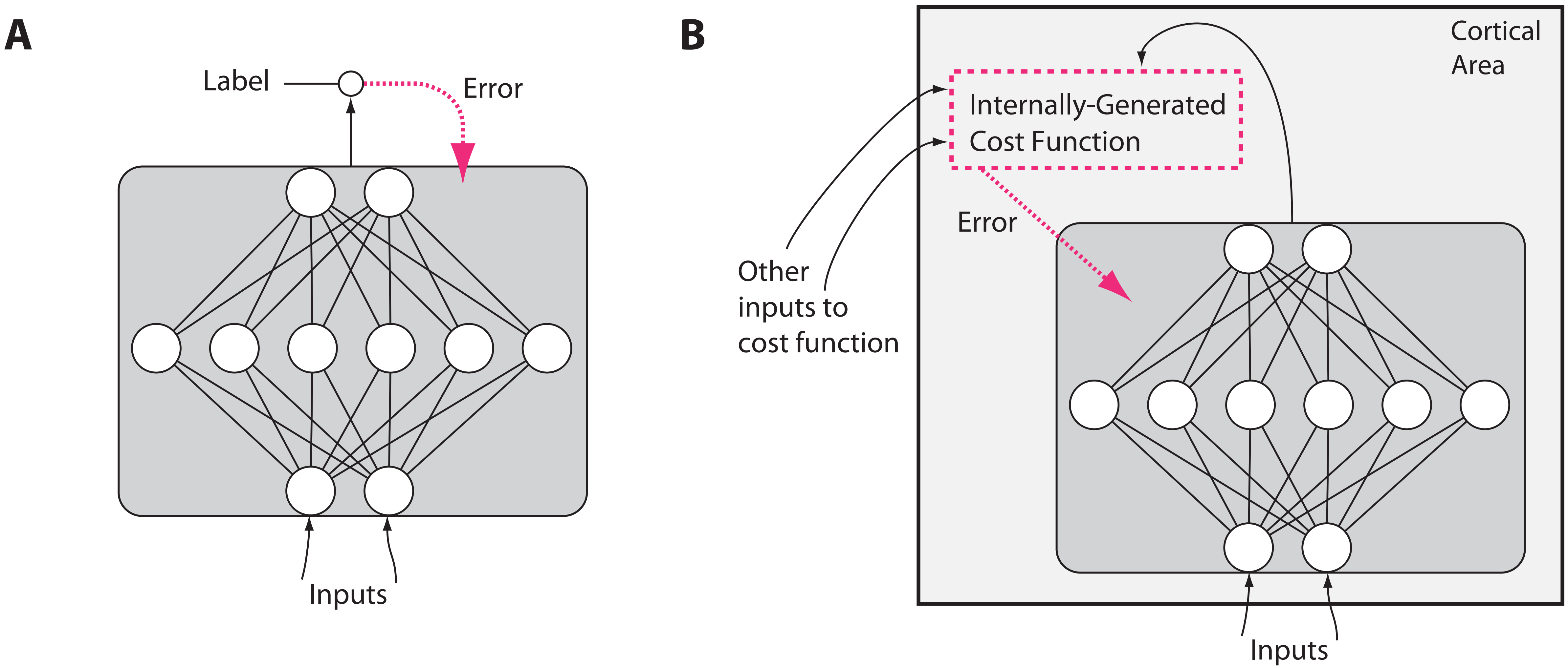
Does it *actually* do this?

Can this be used to explain features of the cortical architecture, e.g., dendritic computation in pyramidal neurons?

Three <u>hypotheses</u> for linking neuroscience and ML

2) Biological fine-structure of cost functions:

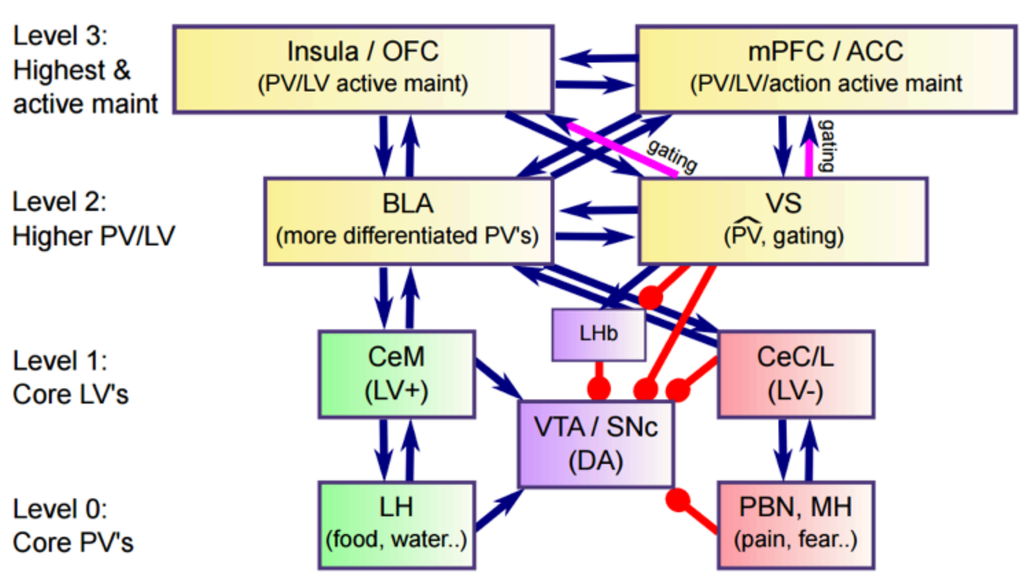
the cost functions are diverse, area-specific and systematically regulated in space and time

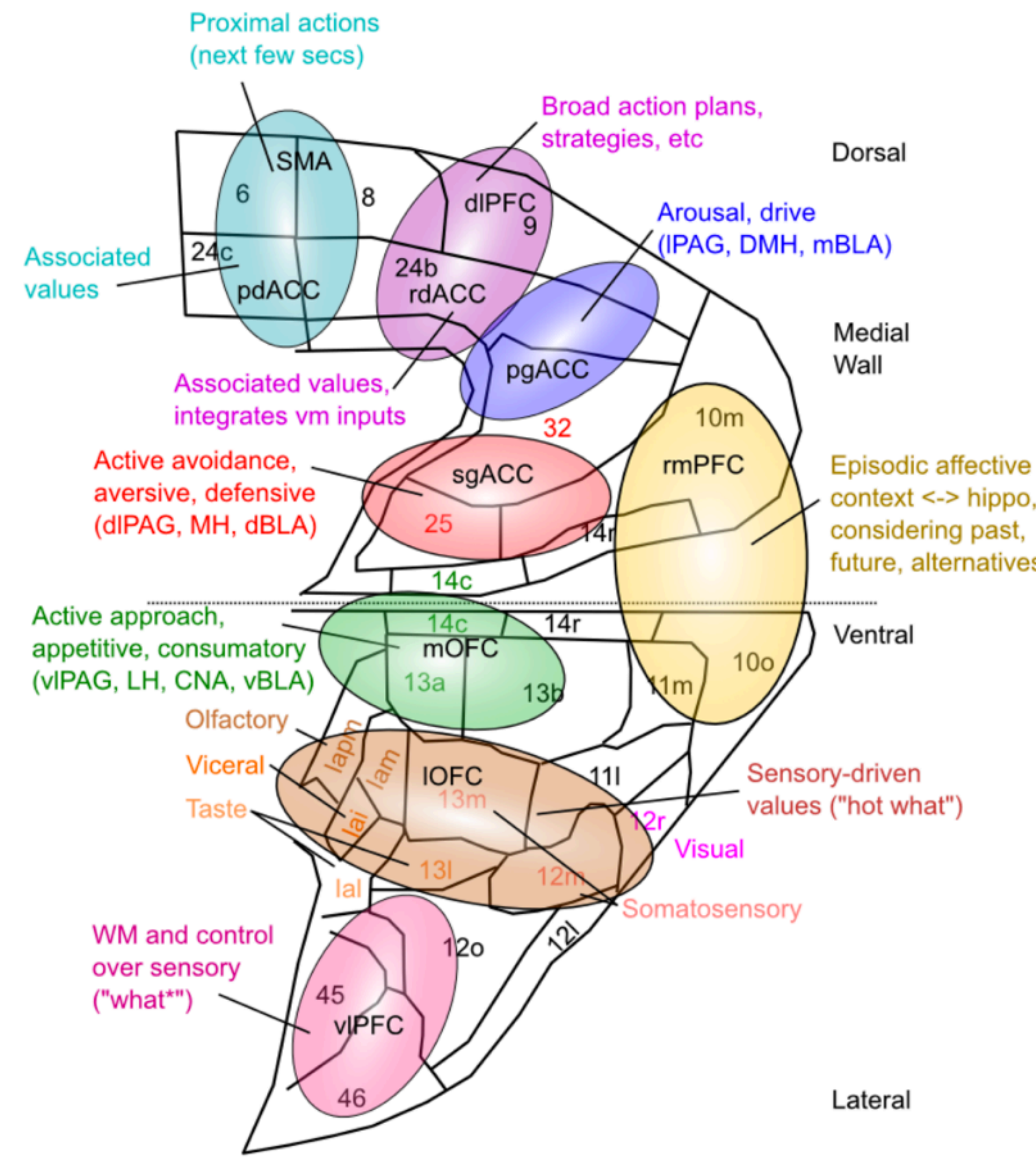


2) Biological fine-structure of cost functions:

the cost functions are diverse, area-specific and systematically regulated in space and time

Global "value functions" vs. *multiple local internal cost functions*

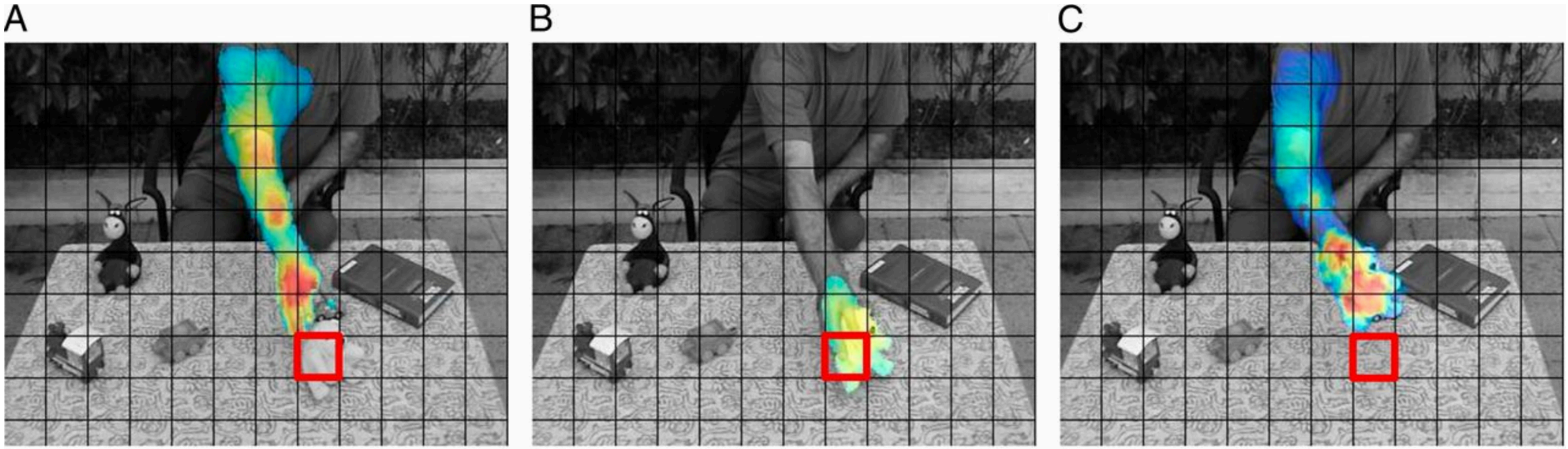




These diagrams describe a global "value function" for "end-to-end" training of the entire brain…
but these aren't the whole story!

Randal O'Reilly

Internally-generated **bootstrap cost functions**: *against* "end to end" training



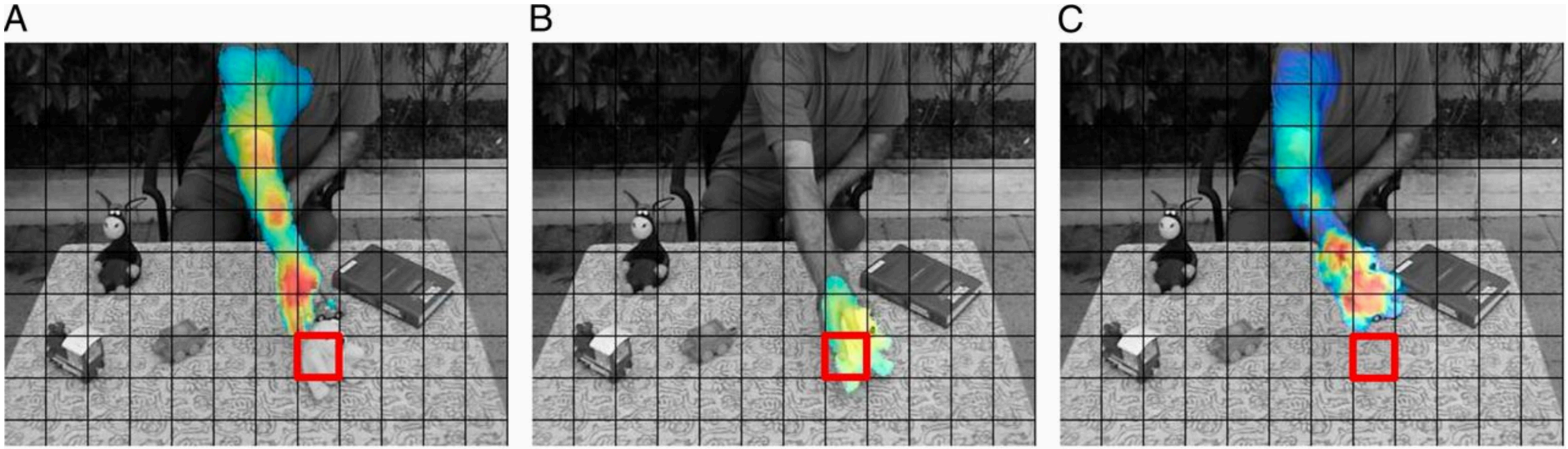
Simple optical flow calculation provides an
internally generated "bootstrap" training signal for hand recognition

Optical flow: bootstraps hand recognition
Hands + faces: bootstraps gaze direction recognition
Gaze direction (and more): bootstraps more complex social cognition

From simple innate biases to complex visual concepts

Shimon Ullman[1,2], Daniel Harari[1], and Nimrod Dorfman[1]

Internally-generated bootstrap cost functions: *against* "end to end" training



Generalizations of this idea could be a key architectural principle for how the biological brain would generate and use ***internal*** *training signals (a form of "weak label")*

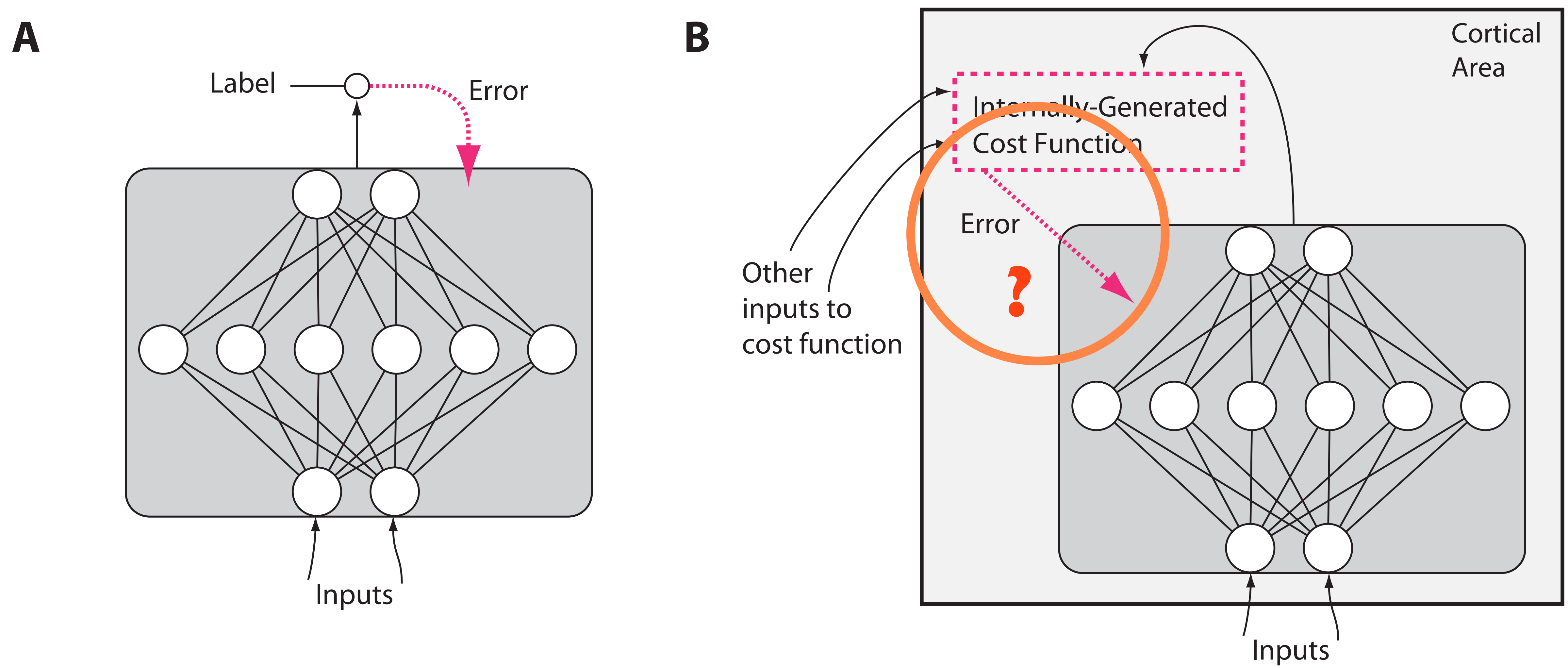
From simple innate biases to complex visual concepts

Shimon Ullman[1,2], Daniel Harari[1], and Nimrod Dorfman[1]

But how are internal cost functions *represented* and *delivered*?

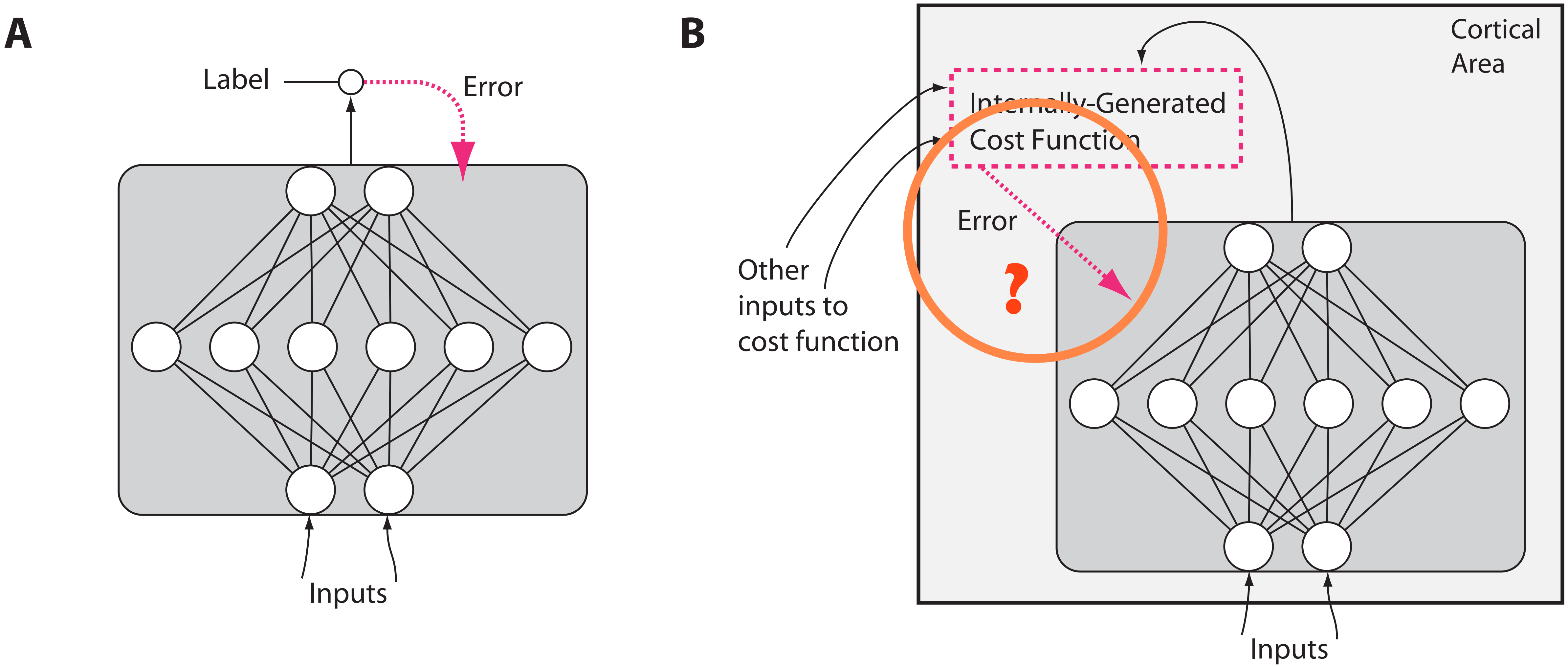
<u>Normal backprop</u>: need a full vectorial target pattern to train towards

<u>Reinforcement</u>: problems of credit assignment are even worse



But how are internal cost functions *represented* and *delivered*?

Normal backprop: need a full vectorial target pattern to train towards
Reinforcement: problems of credit assignment are even worse



Possibility: *The brain may re-purpose deep* ***reinforcement*** *learning to optimize diverse internal cost functions, which are computed internally and delivered as scalars*

Ways of making deep *RL* efficient

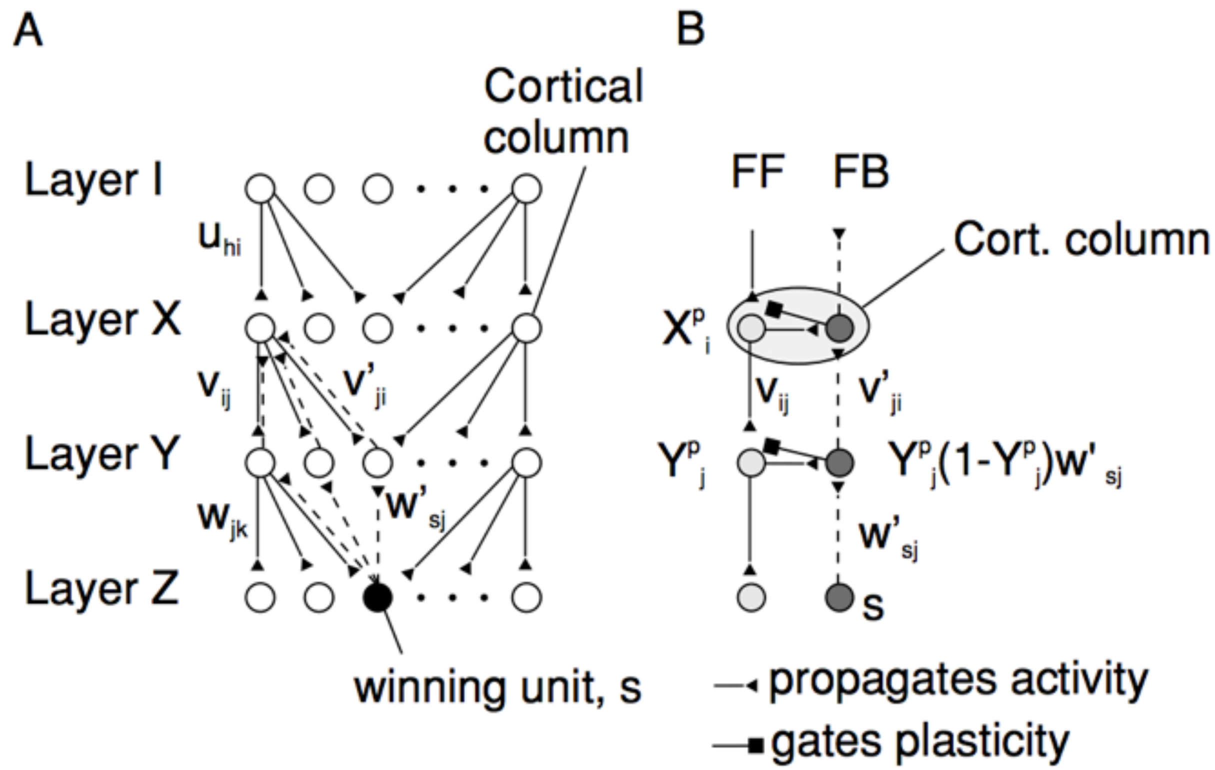


Figure 4: Generalization of AGREL to networks with more than three layers. (A) Feedforward connections u, v, and w propagate activity from the input layer I through two hidden layers to the output layer Z. The winning output unit, s, feeds back to units in layer Y through connections w'_{sj}. All units in Y that receive feedback from Z propagate it to layer X through feedback connections v'_{ji}. (B) Units of AGREL are hypothesized to correspond to cortical columns that contain FF neurons (light gray circles) that propagate activity to the next higher layer as well as FB neurons (dark gray) that propagate activity to the previous layer. FB neurons gate plasticity in the FF pathway, but they do not directly influence the activity of FF neurons (connection with square).

Neural Comput. 2005 Oct;17(10):2176-214.

Attention-gated reinforcement learning of internal representations for classification.

Roelfsema PR[1], van Ooyen A.

Ways of making deep *RL* efficient

Algorithm 1 Deep Q-learning with Experience Replay

Initialize replay memory $\mathcal{D}$ to capacity N
Initialize action-value function Q with random weights
for episode $= 1, M$ **do**
  Initialise sequence $s_1 = \{x_1\}$ and preprocessed sequenced $\phi_1 = \phi(s_1)$
  for $t = 1, T$ **do**
    With probability ϵ select a random action a_t
    otherwise select $a_t = \max_a Q^*(\phi(s_t), a; \theta)$
    Execute action a_t in emulator and observe reward r_t and image x_{t+1}
    Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
    Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in $\mathcal{D}$
    Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from $\mathcal{D}$
    Set $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$
    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ according to equation 3
  end for
end for

“biologically plausible”?

Playing Atari with Deep Reinforcement Learning

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller

A complex molecular and cellular basis for reinforcement-based training in *primary* visual cortex

Reinforcement in striatum: VTA dopaminergic projections
Reinforcement in cortex: basal forebrain cholinergic projections

with a *glial* intermediate!

A Cholinergic Mechanism for Reward Timing within Primary Visual Cortex

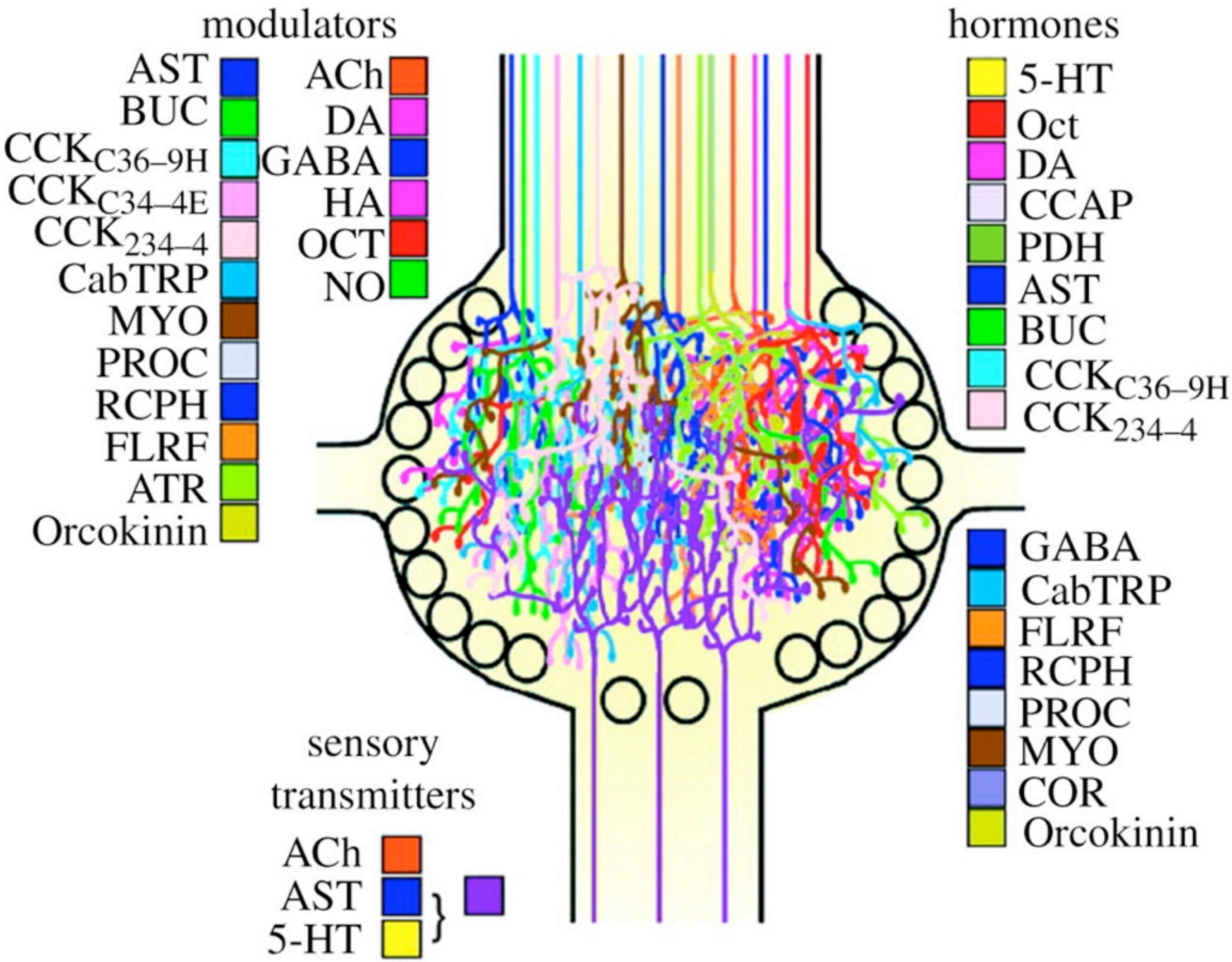
Alexander A. Chubykin[3], Emma B. Roach[3], Mark F. Bear, Marshall G. Hussain Shuler

[3] These authors contributed equally to this work

Nucleus basalis-enabled stimulus-specific plasticity in the visual cortex is mediated by astrocytes (i.e., glia not neurons)

Naiyan Chen[a,1], Hiroki Sugihara[a,1], Jitendra Sharma[a,b], Gertrudis Perea[a], Jeremy Petravicz[a], Chuong Le[a], and Mriganka Sur[a,2]

A diversity of reinforcement-like signals?



Classic work by Eve Marder in the crab stomatogastric ganglion

2) **Biological fine-structure of cost functions**:

the cost functions are diverse, area-specific and systematically regulated in space and time

Take Away

Not a single "end-to-end" cost function

A series of cost functions generated internally and deployed to particular brain areas at particular times in a genetically and developmentally regulated fashion

Bootstrapping of learning based on heuristics and weak labels ("prior knowledge" encoded into the training process)

Reinforcement system may be re-purposed for diverse internal cost functions, and coupled with multi-layer credit assignment in deep networks

2) Biological fine-structure of cost functions:
the cost functions are diverse, area-specific and systematically regulated in space and time

Key Research Questions

Can we find some concrete examples of how cost functions are actually computed, represented, and applied in the brain?

Which forms of "bootstrapping" of learning (e.g., cues, heuristics, internally generated reward signals) are enabled by evolutionary "prior knowledge" of the human body/ environment, encoded by evolution into staged developmental learning processes?

What is the *full* map of the brain's reinforcement pathways, e.g., extending all the way into primary visual areas?

2) Biological fine-structure of cost functions:
the cost functions are diverse, area-specific and systematically regulated in space and time

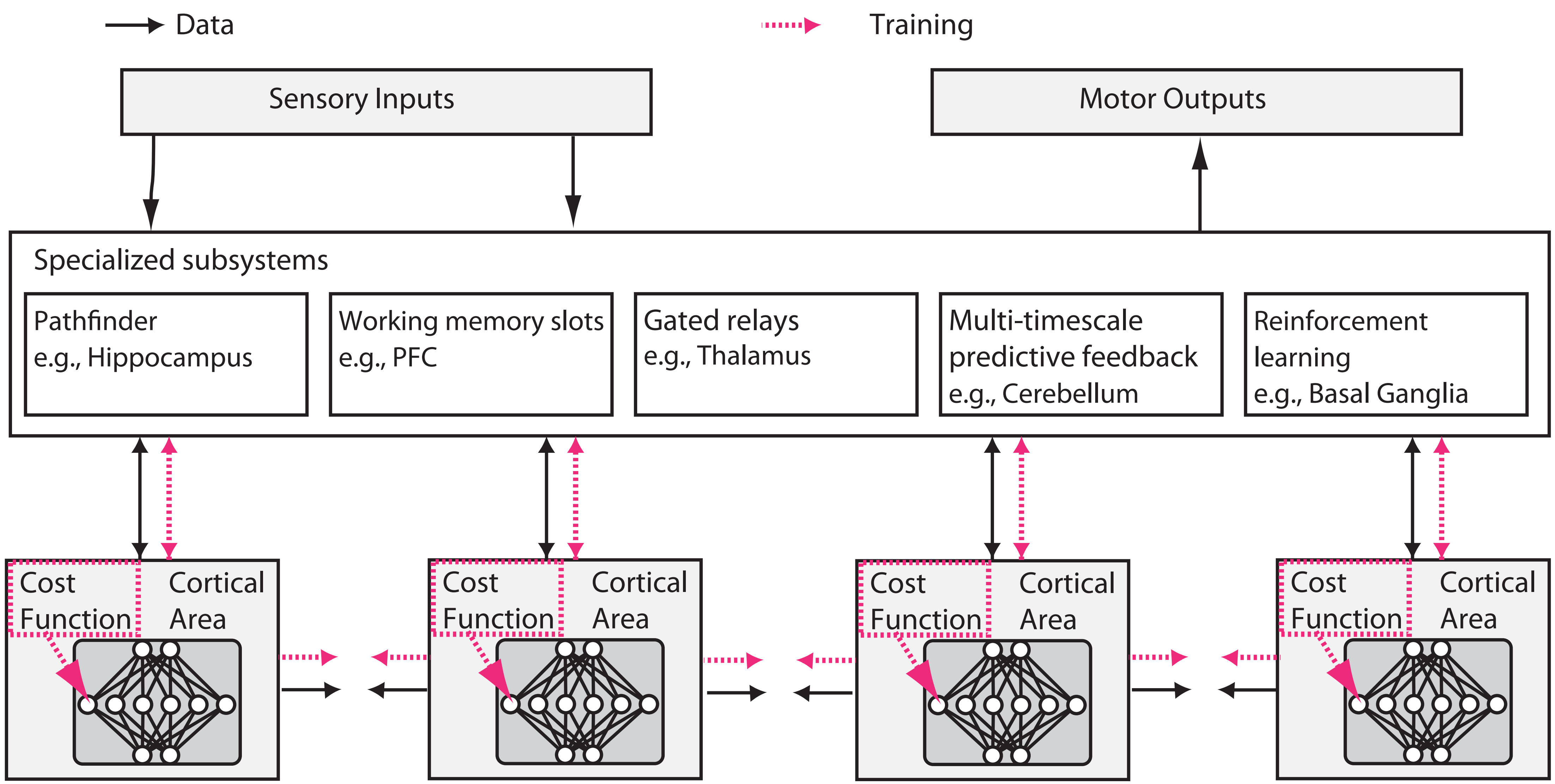
Key Research Questions

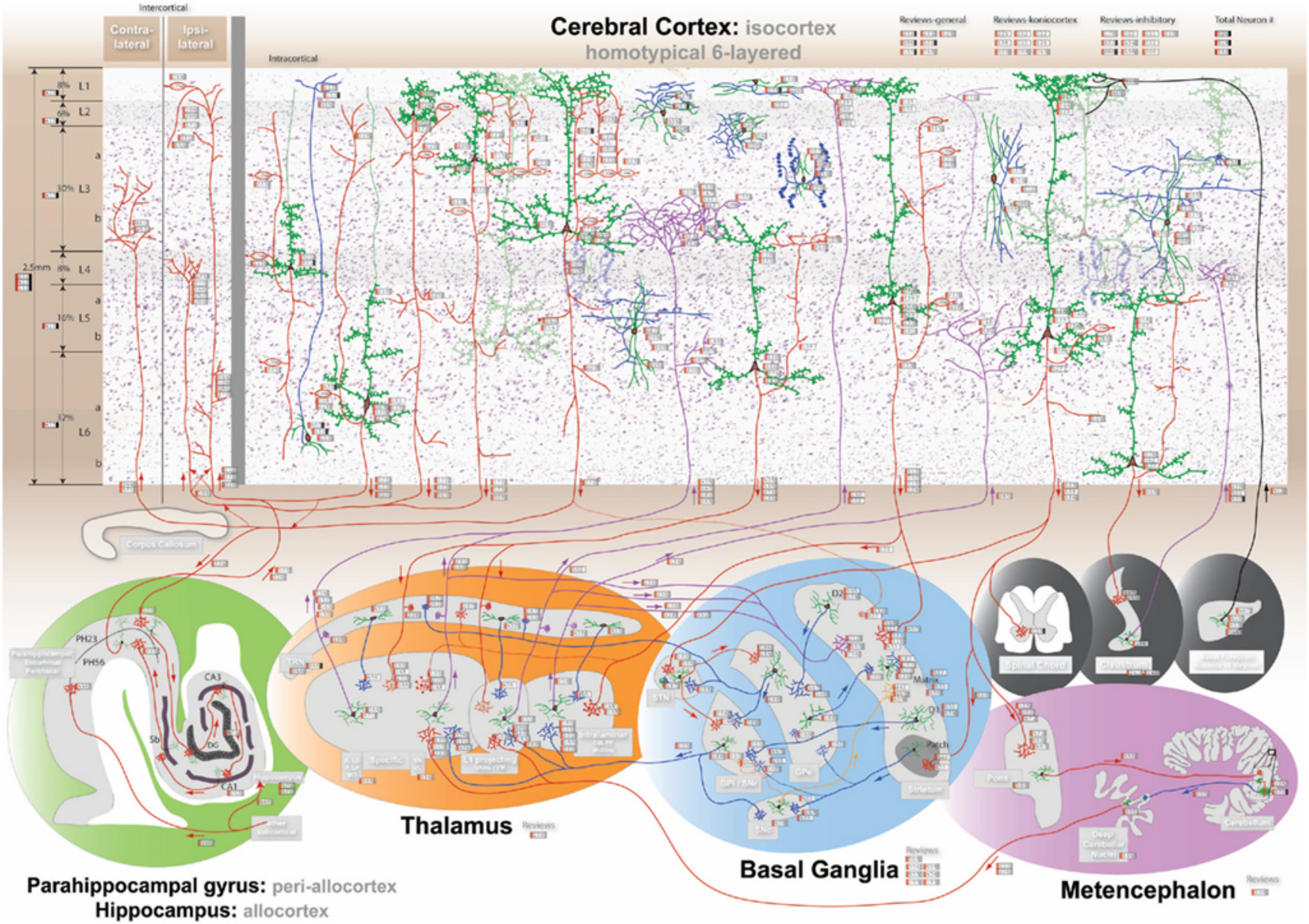
Dopamine neurons encode performance error in singing birds

Vikram Gadagkar, Pavel A. Puzerey, Ruidong Chen, Eliza Baird-Daniel*, Alexander R. Farhang†, Jesse H. Goldberg‡

Three hypotheses for linking neuroscience and ML

3) Embedding within a pre-structured architecture:
the brain contains dedicated, specialized systems for efficiently solving key problems whose solutions are not easily bootstrapped by learning, such as information routing and variable binding

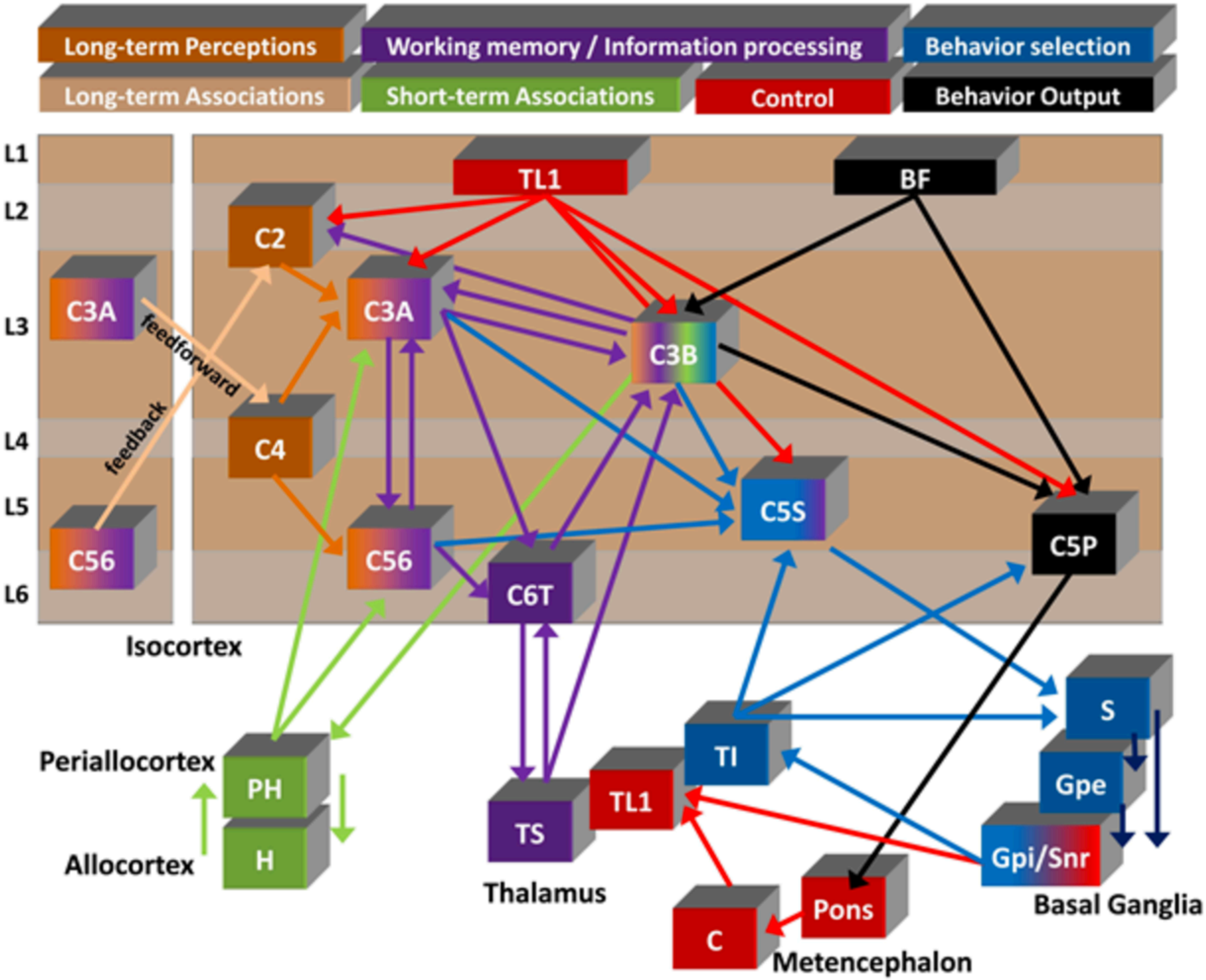




Cognitive consilience: primate non-primary neuroanatomical circuits underlying cognition

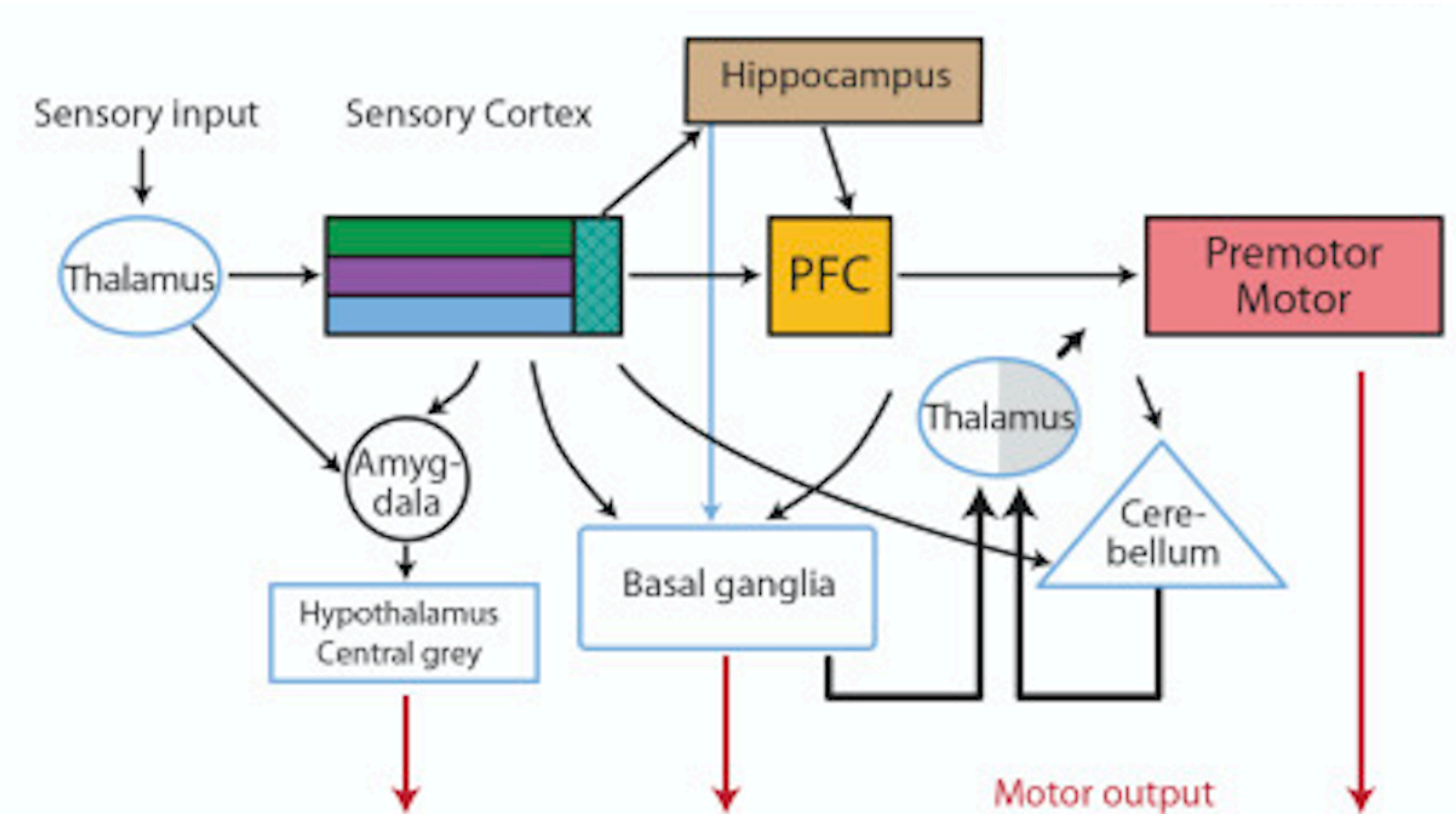
Soren Van Hout Solari[1,2]* and Rich Stoner[3]*

Solari and Stoner cognitive model



Solari and Stoner 2011

Neuroscience broadly has found an array of specialized structures

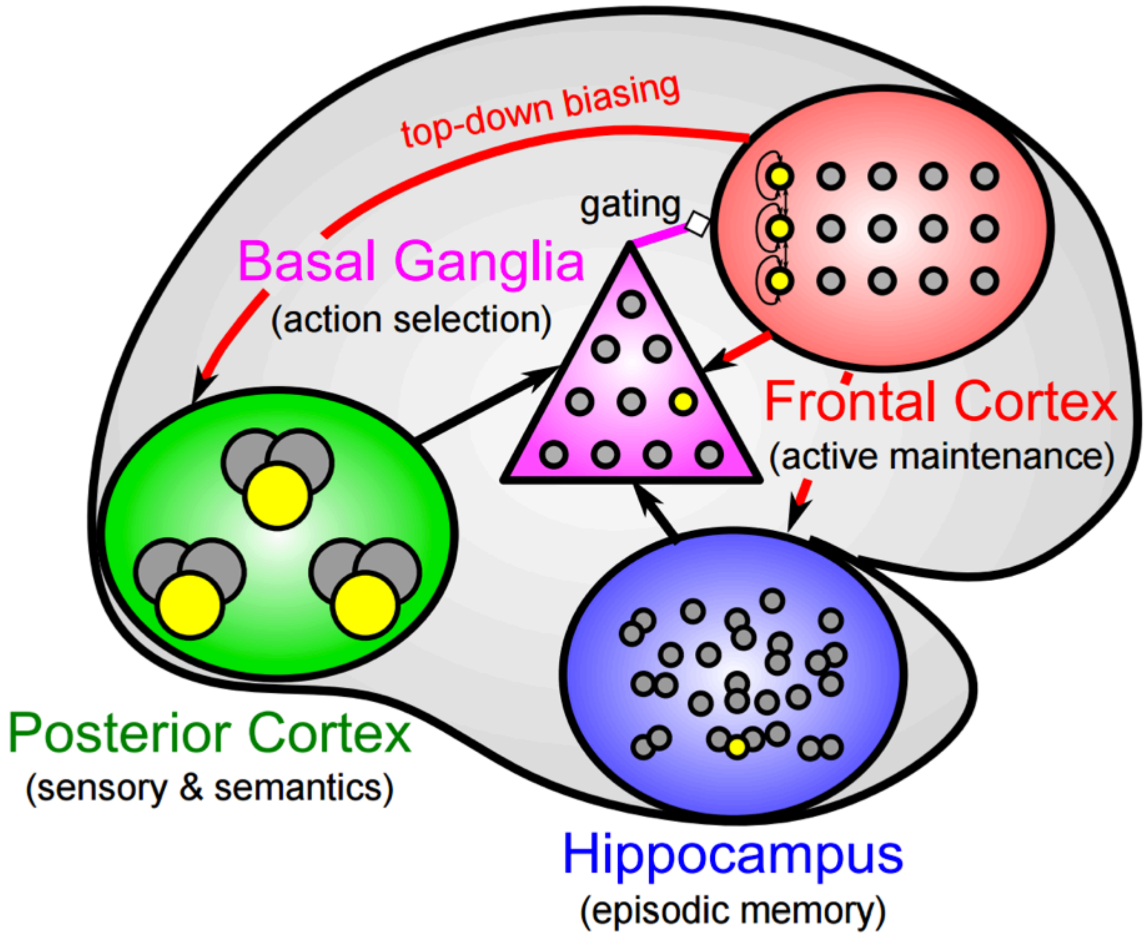


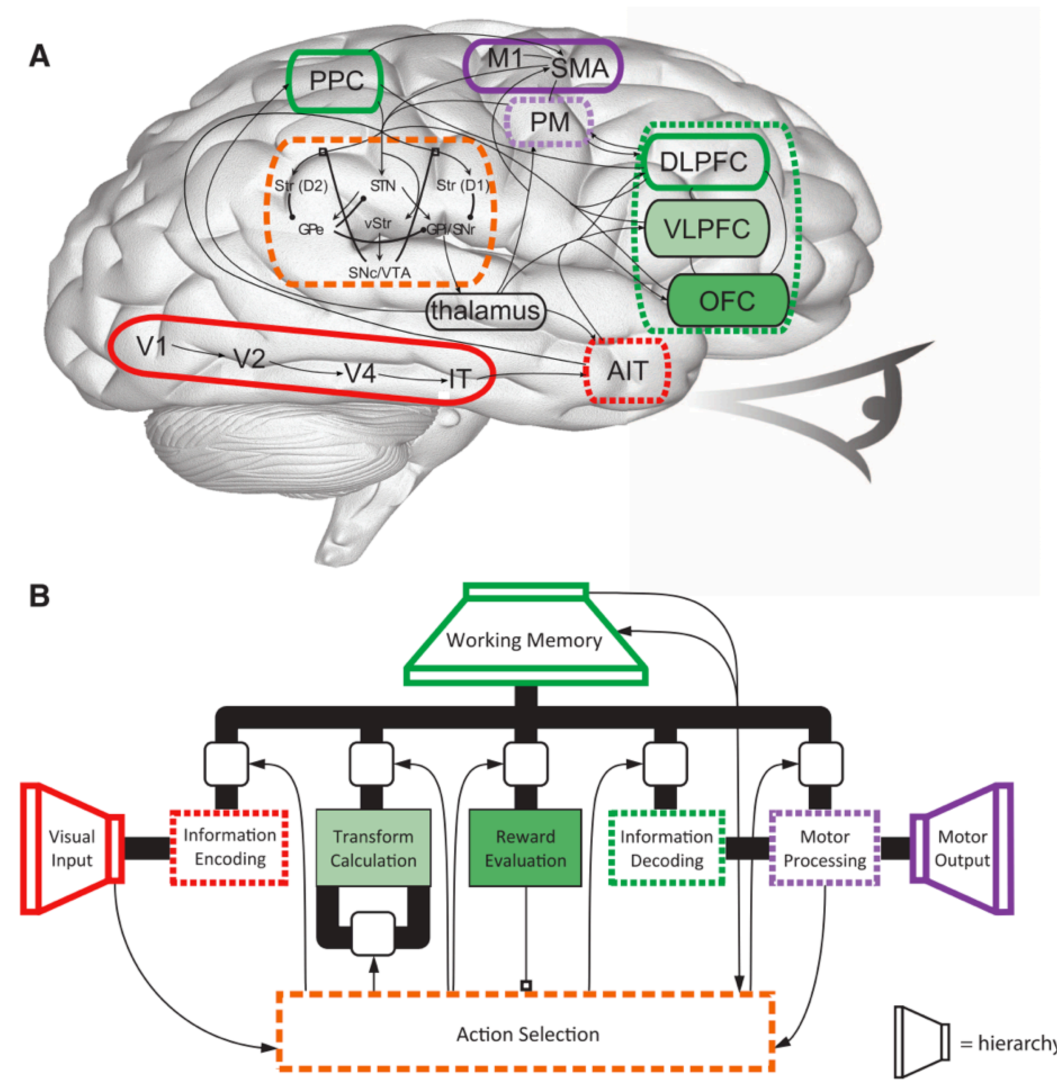
Perspective

The Challenge of Understanding the Brain: Where We Stand in 2015

John Lisman[1]

Integrated "biological" cognitive architectures: LEABRA and SPAUN





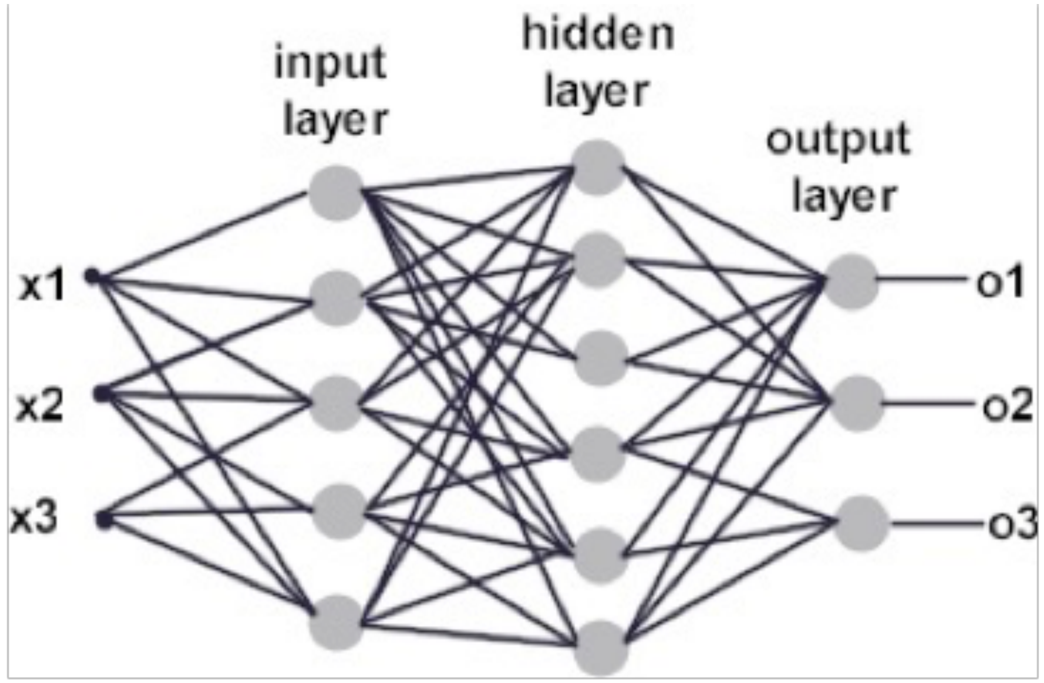
Interesting but do not show "powerful" AI performance

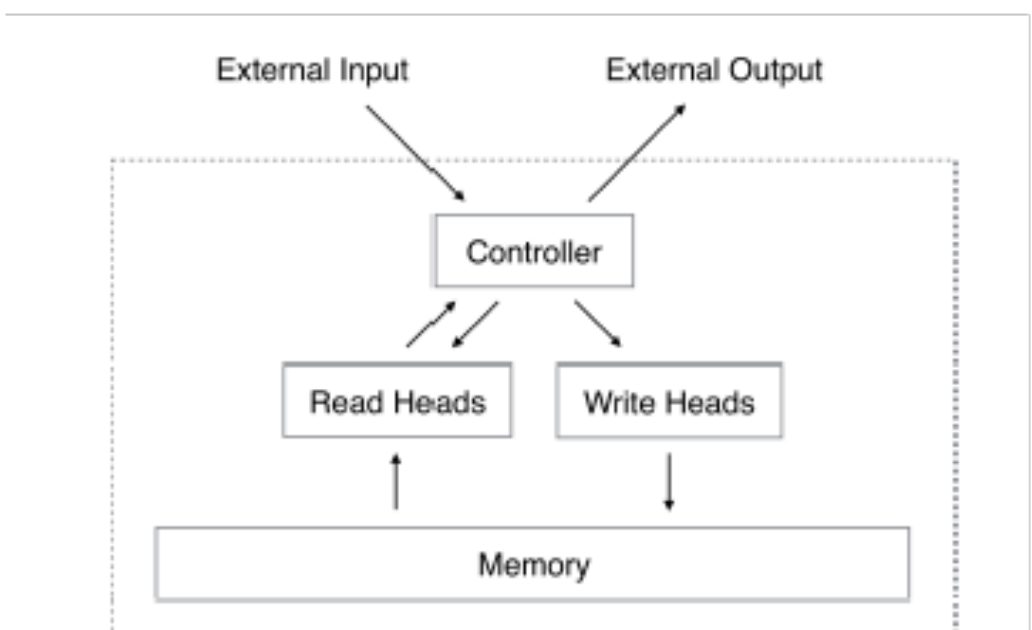
The Leabra Cognitive Architecture:
How to Play 20 Principles with Nature and Win!

Randall C. O'Reilly, Thomas E. Hazy, and Seth A. Herd
Department of Psychology and Neuroscience
University of Colorado Boulder
345 UCB
Boulder, CO 80309
randy.oreilly@colorado.edu

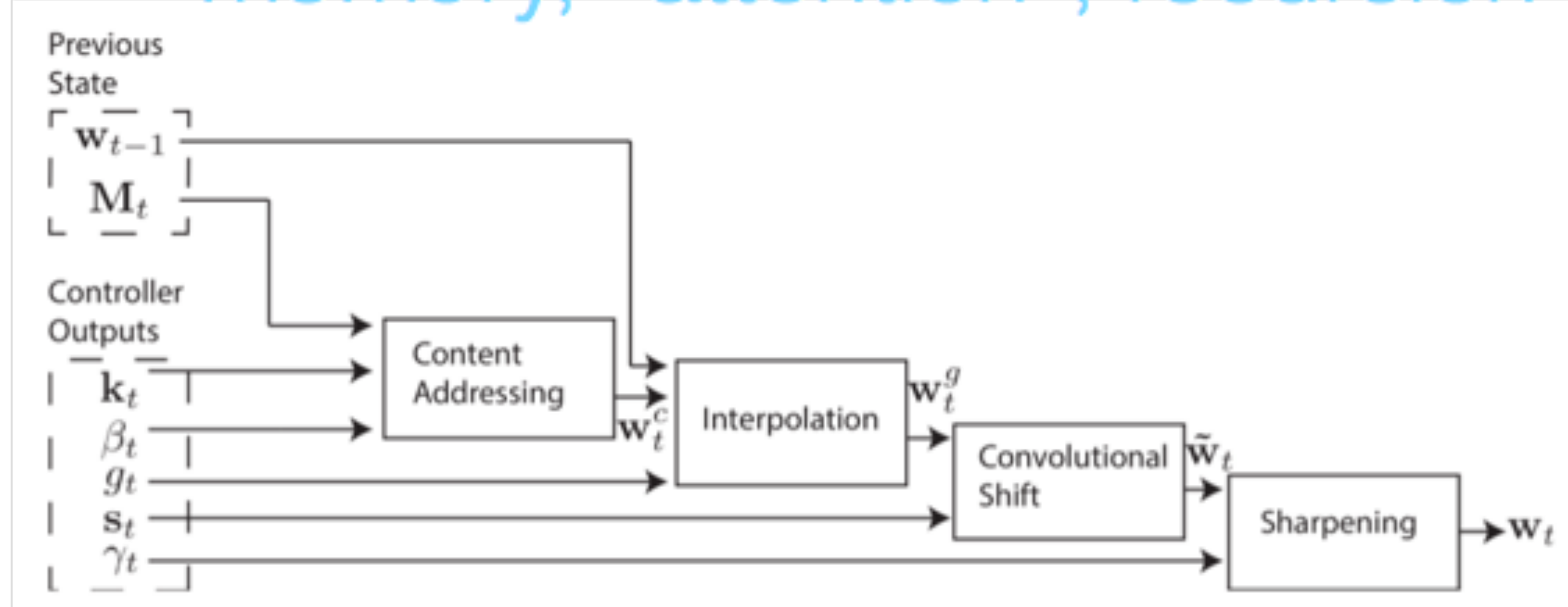
A Large-Scale Model of the Functioning Brain
Chris Eliasmith *et al.*
Science **338**, 1202 (2012);
DOI: 10.1126/science.1225266

<u>Compare</u>: Emerging structured machine learning architectures



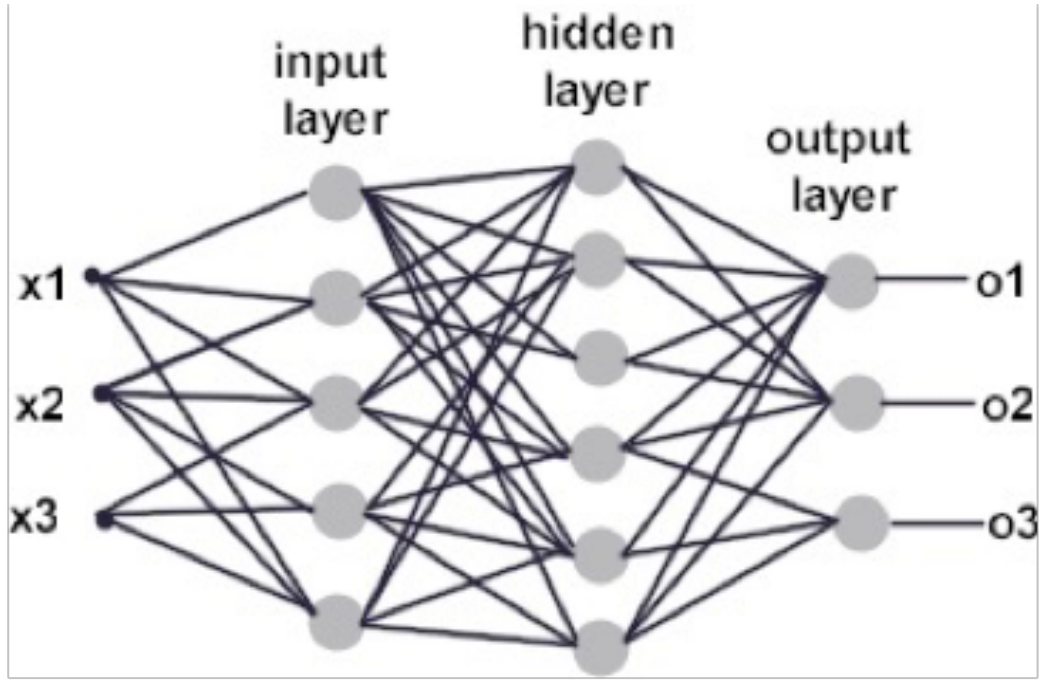


memory, “attention”, recursion



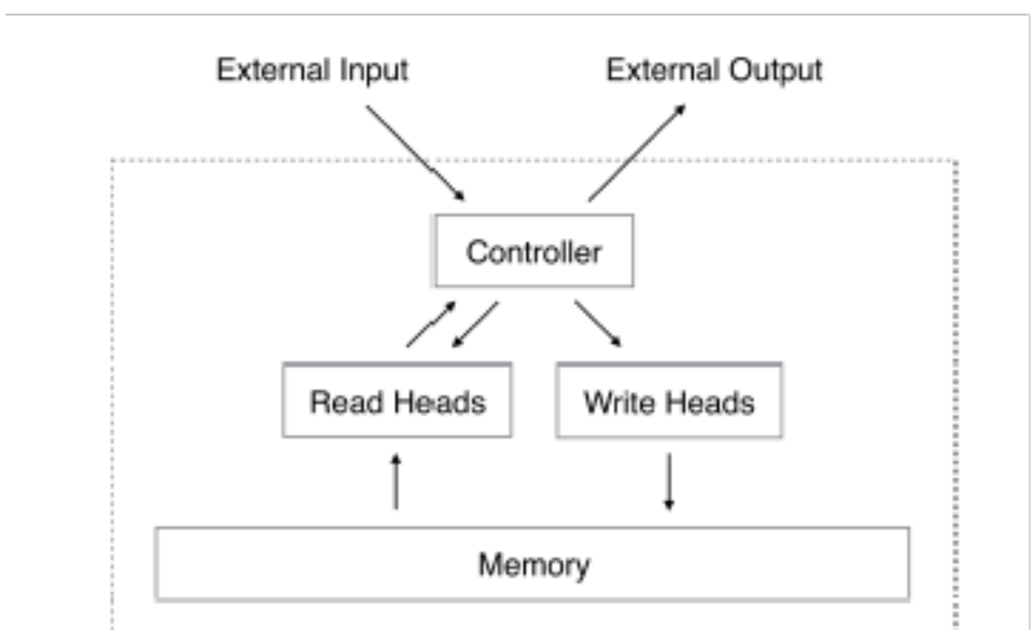
Graves, Wayne, Danihelka (2014)

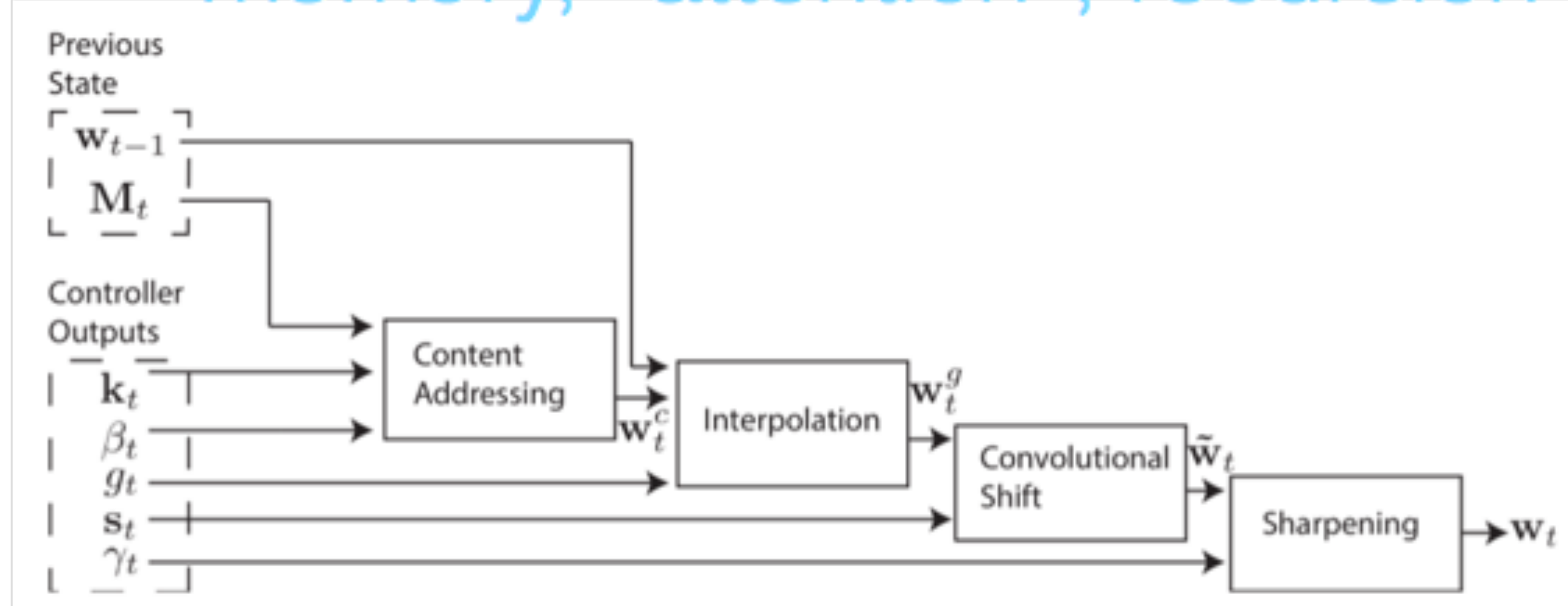
Compare: Emerging structured machine learning architectures



Need a "hippocampus" for fast associations, buffers for "working memory", and fast routing/control, because "cortical deep learning" is slow and statistical…

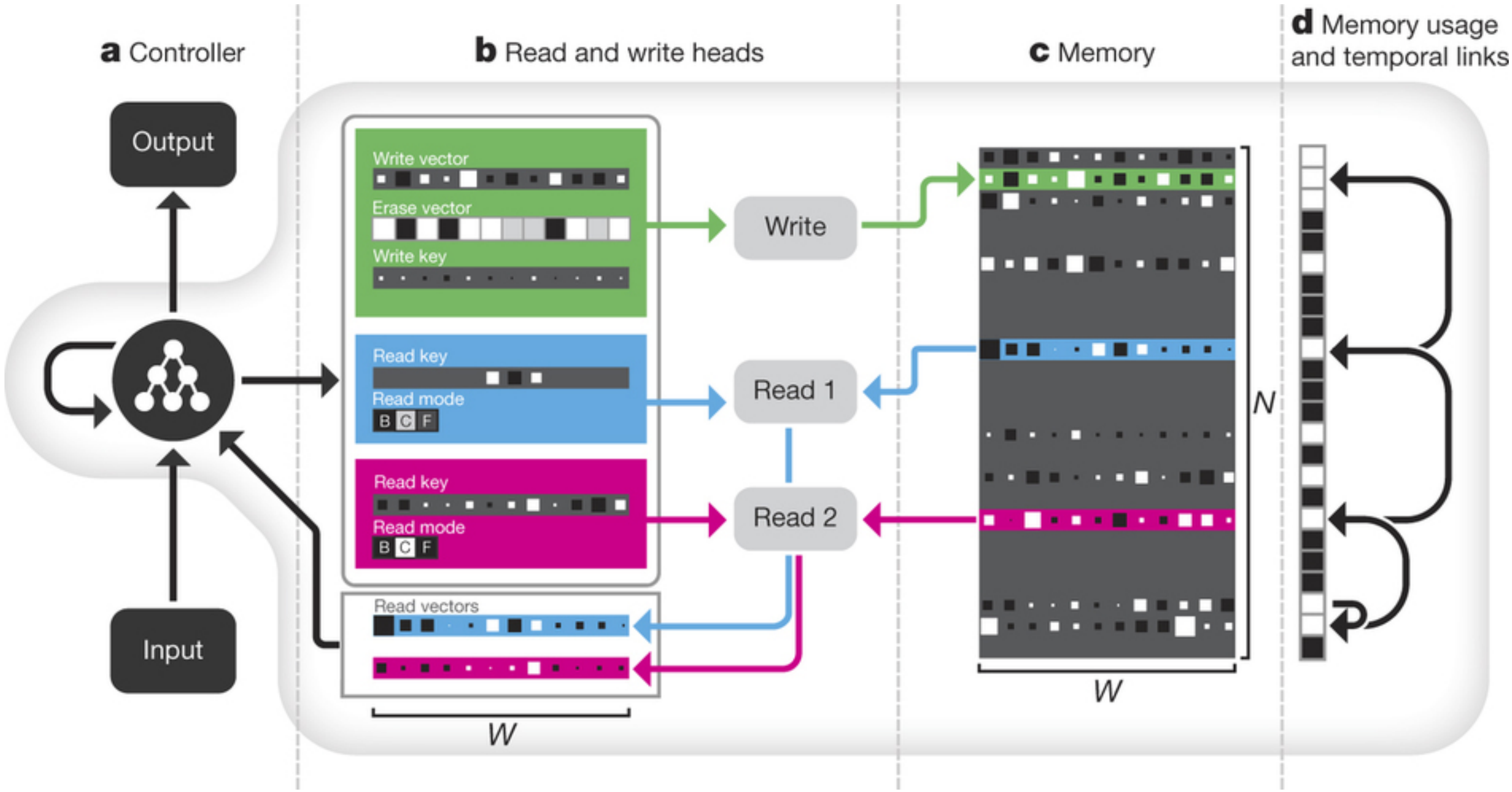
memory, "attention", recursion





Graves, Wayne, Danihelka (2014)

<u>Compare</u>: Emerging structured machine learning architectures

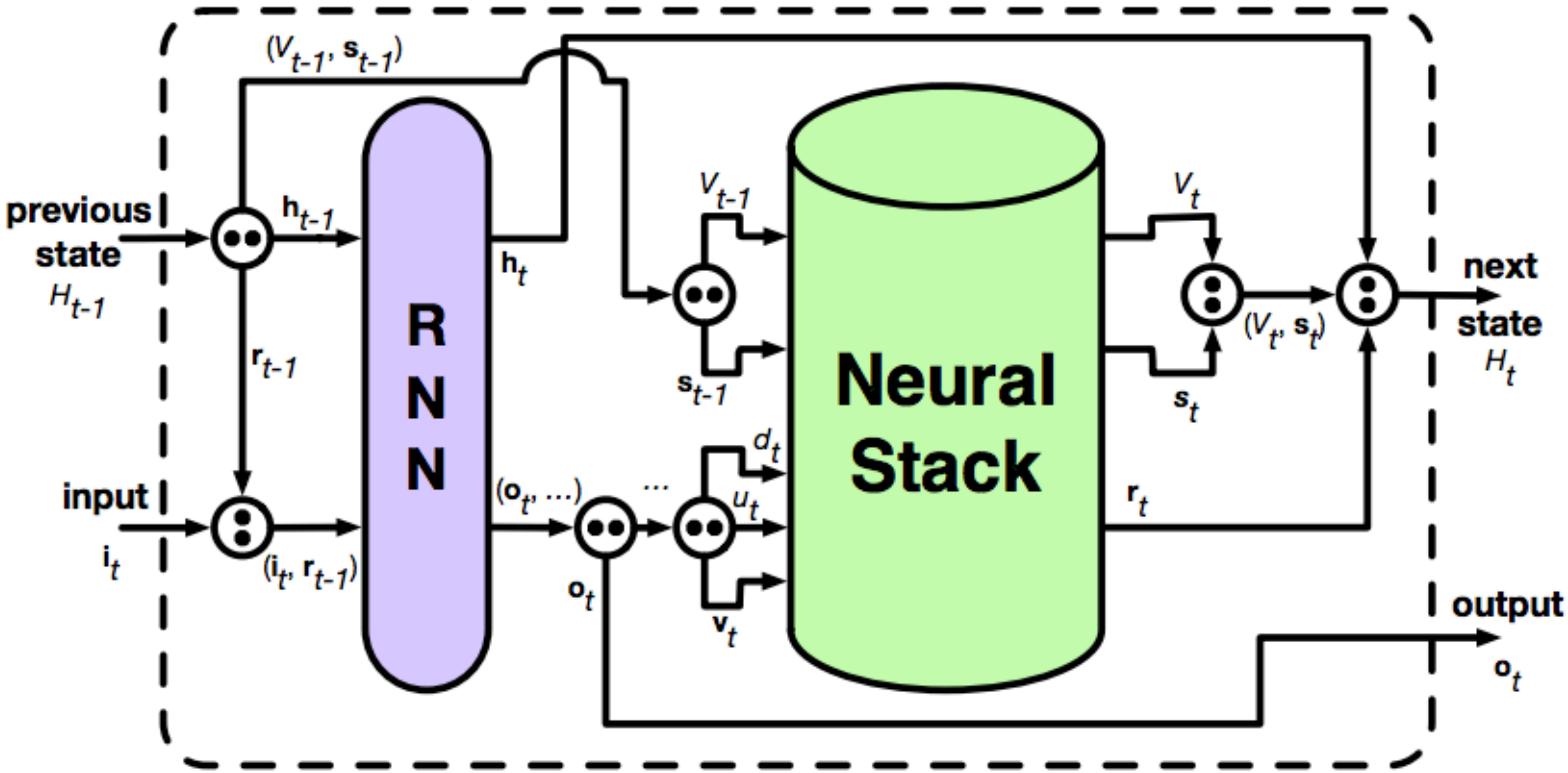


Memory system is already somewhat hippocampus-inspired…

Hybrid computing using a neural network with dynamic external memory

Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, Adrià Puigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen King, Christopher Summerfield, Phil Blunsom, Koray Kavukcuoglu & Demis Hassabis

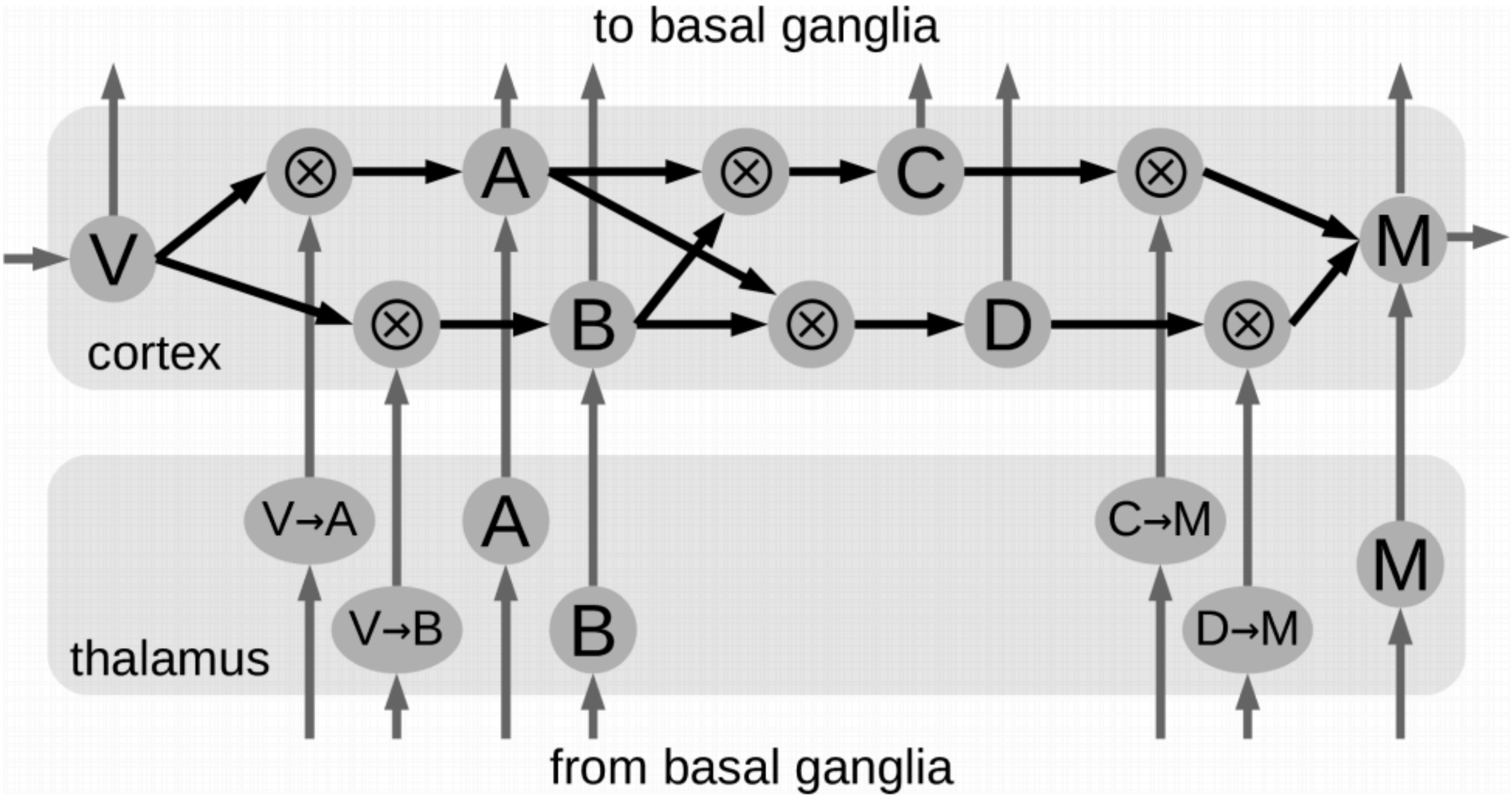
<u>Compare</u>: Emerging structured machine learning architectures



(c) RNN Controlling a Stack

Figure 1: Illustrating a Neural Stack's Operations, Recurrent Structure, and Control

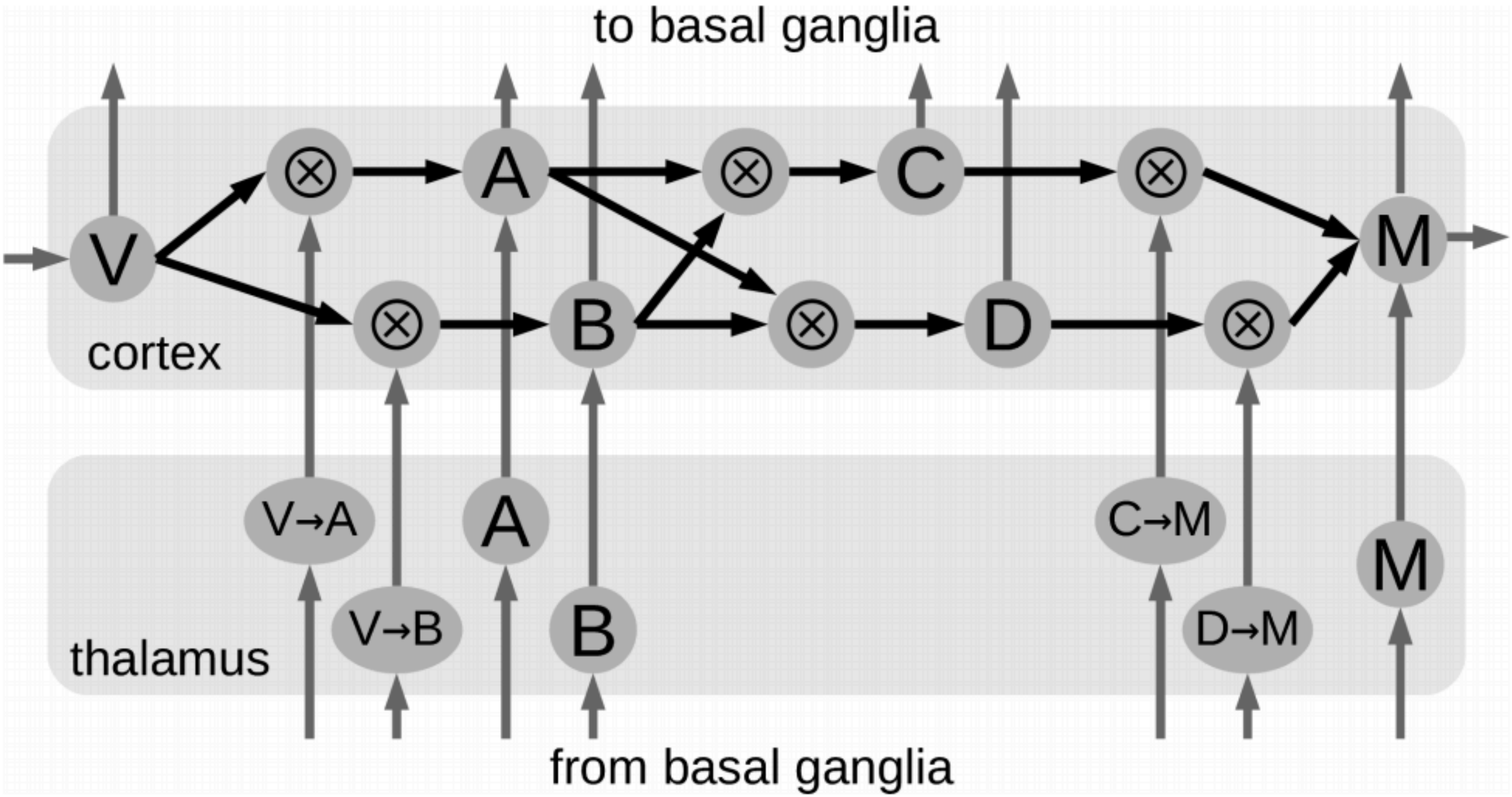
Pre-structured architectures in the brain: to make learning efficient?



thalamic gating of "copy and paste" operations between cortical working memory buffers, executing a sequence of steps controlled by the basal ganglia

Stewart, Eliasmith et al 2010

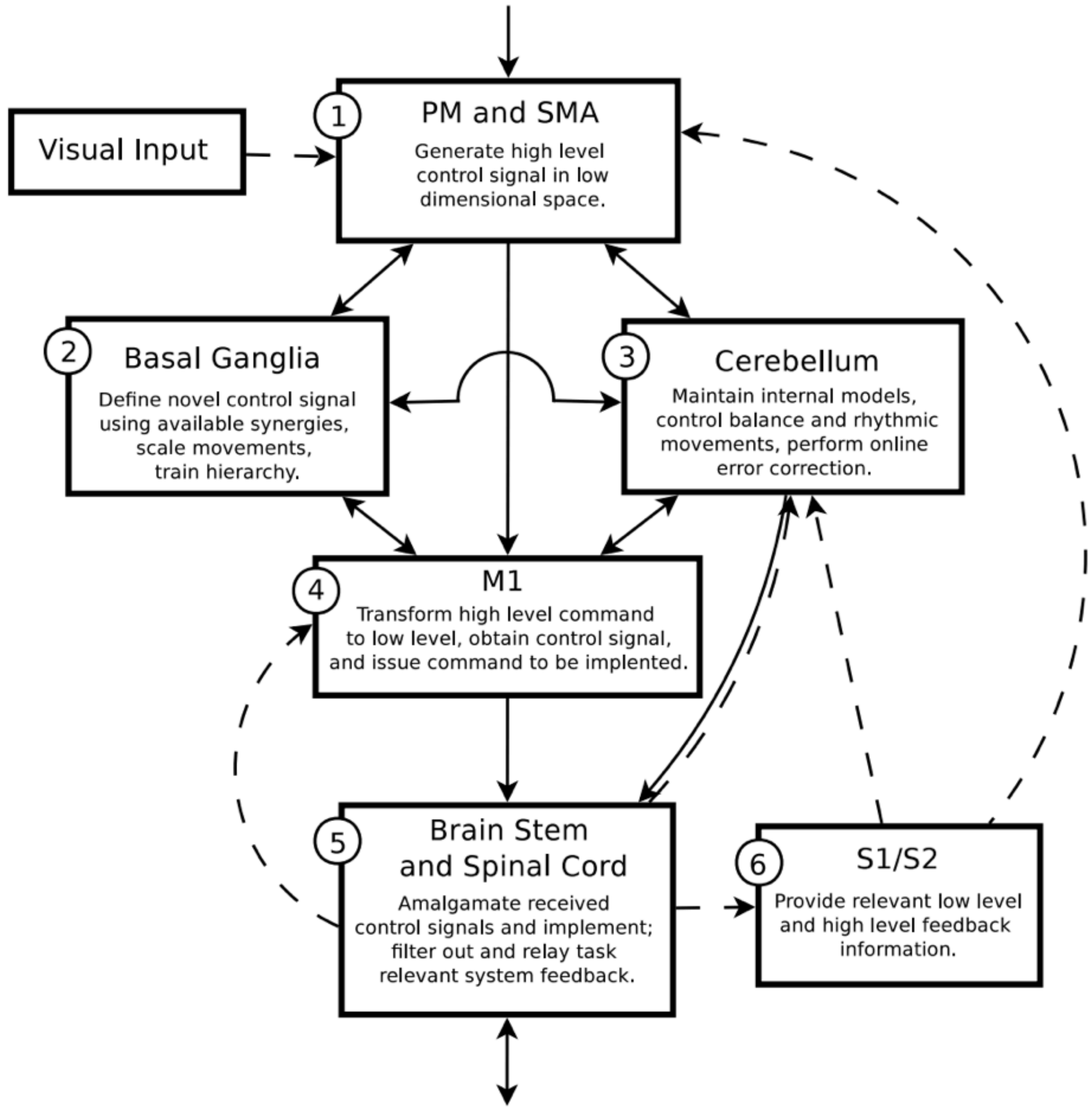
Pre-structured architectures in the brain: to make learning efficient?



needs this for flexible routing and discrete state changes (i.e., "programs")?

Stewart, Eliasmith et al 2010

Pre-structured architectures in the brain: to make learning efficient?



The neural optimal control hierarchy for motor control

T DeWolf and C Eliasmith

3) Embedding within a pre-structured architecture:
the brain contains dedicated, specialized systems for efficiently solving key problems whose solutions are not easily bootstrapped by learning, such as information routing and variable binding

Take Away

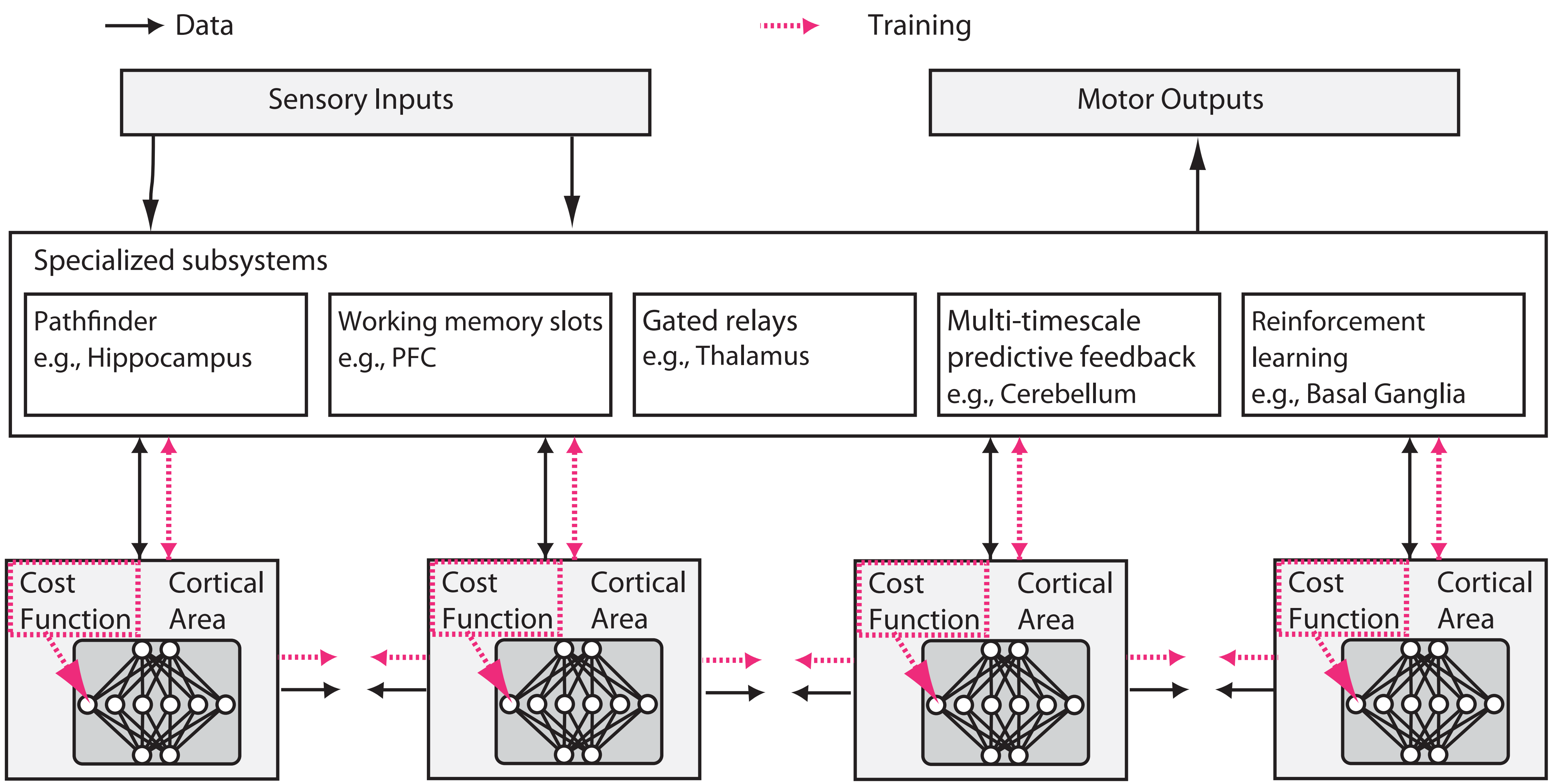
Specialized brain systems (memory, routing, attention, control, …) may allow optimization to solve otherwise inaccessible problems, much as external memories can augment deep artificial neural networks

3) Embedding within a pre-structured architecture:
the brain contains dedicated, specialized systems for efficiently solving key problems whose solutions are not easily bootstrapped by learning, such as information routing and variable binding

Key Research Questions

How does the hippocampus encode short-term memories and can the same principles be applied to create an optimal "external memory" for artificial neural networks?

Does the brain have specialized systems to enable "symbolic" processing, e.g., "variable binding"?



Data
Training
Sensory Inputs
Motor Outputs
Specialized subsystems
Pathfinder
e.g., Hippocampus
Working memory slots
e.g., PFC
Gated relays
e.g., Thalamus
Multi-timescale
predictive feedback
e.g., Cerebellum
Reinforcement
learning
e.g., Basal Ganglia
Cost
Function
Cortical
Area
Cost
Function
Cortical
Area
Cost
Function
Cortical
Area
Cost
Function
Cortical
Area

Differences with today's deep learning

Information represented via assemblies/attractors

Autoassociative dynamics in the generation of sequences of hippocampal place cells

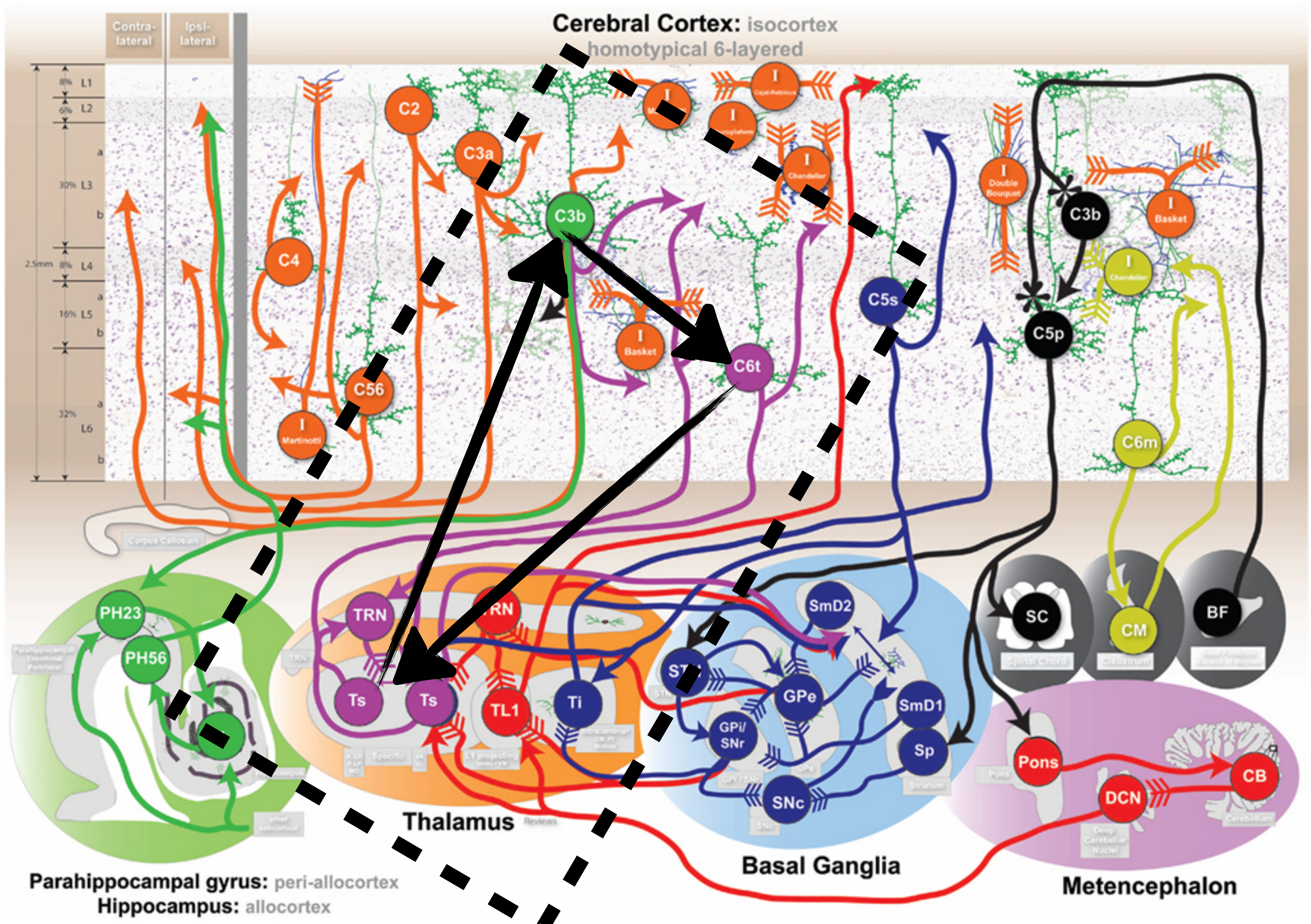
Brad E. Pfeiffer* and David J. Foster†

Neuronal circuits produce self-sustaining sequences of activity patterns, but the precise mechanisms remain unknown. Here we provide evidence for autoassociative dynamics in sequence generation. During sharp-wave ripple (SWR) events, hippocampal neurons express sequenced reactivations, which we show are composed of discrete attractors. Each attractor corresponds to a single location, the representation of which sharpens over the course of several milliseconds, as the reactivation focuses at that location. Subsequently, the reactivation transitions rapidly to a spatially discontiguous location. This alternation between sharpening and transition occurs repeatedly within individual SWRs and is locked to the slow-gamma (25 to 50 hertz) rhythm. These findings support theoretical notions of neural network function and reveal a fundamental discretization in the retrieval of memory in the hippocampus, together with a function for gamma oscillations in the control of attractor dynamics.

See also: "Imprinting and recalling cortical ensembles" by Yuste lab

Differences with today's deep learning

The attractors may be in cortico-thalamo-cortical loops

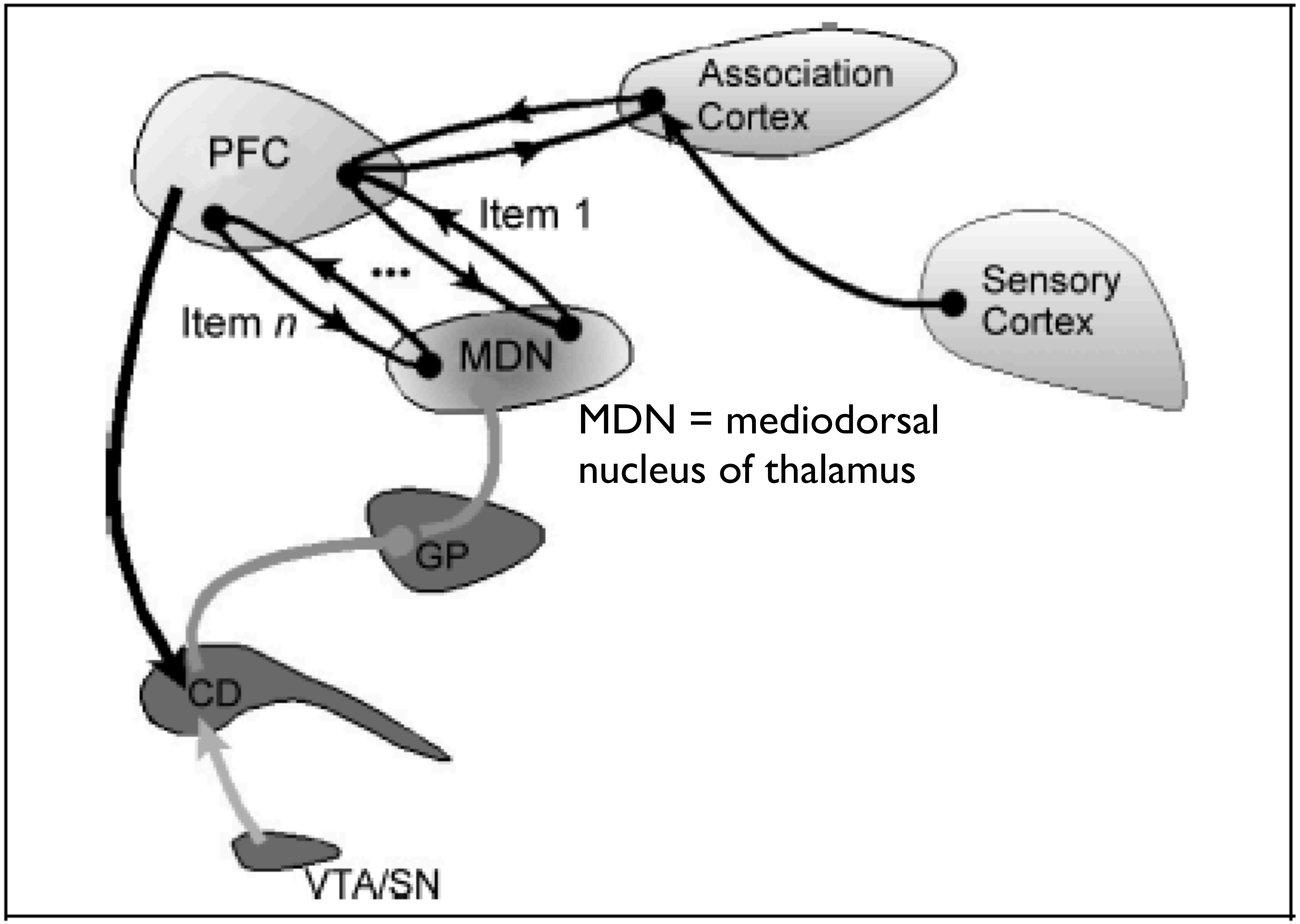


Cognitive consilience: primate non-primary neuroanatomical circuits underlying cognition

Soren Van Hout Solari[1,2]* and Rich Stoner[3]*

Differences with today's deep learning

The attractors may be in cortico-thalamo-cortical loops



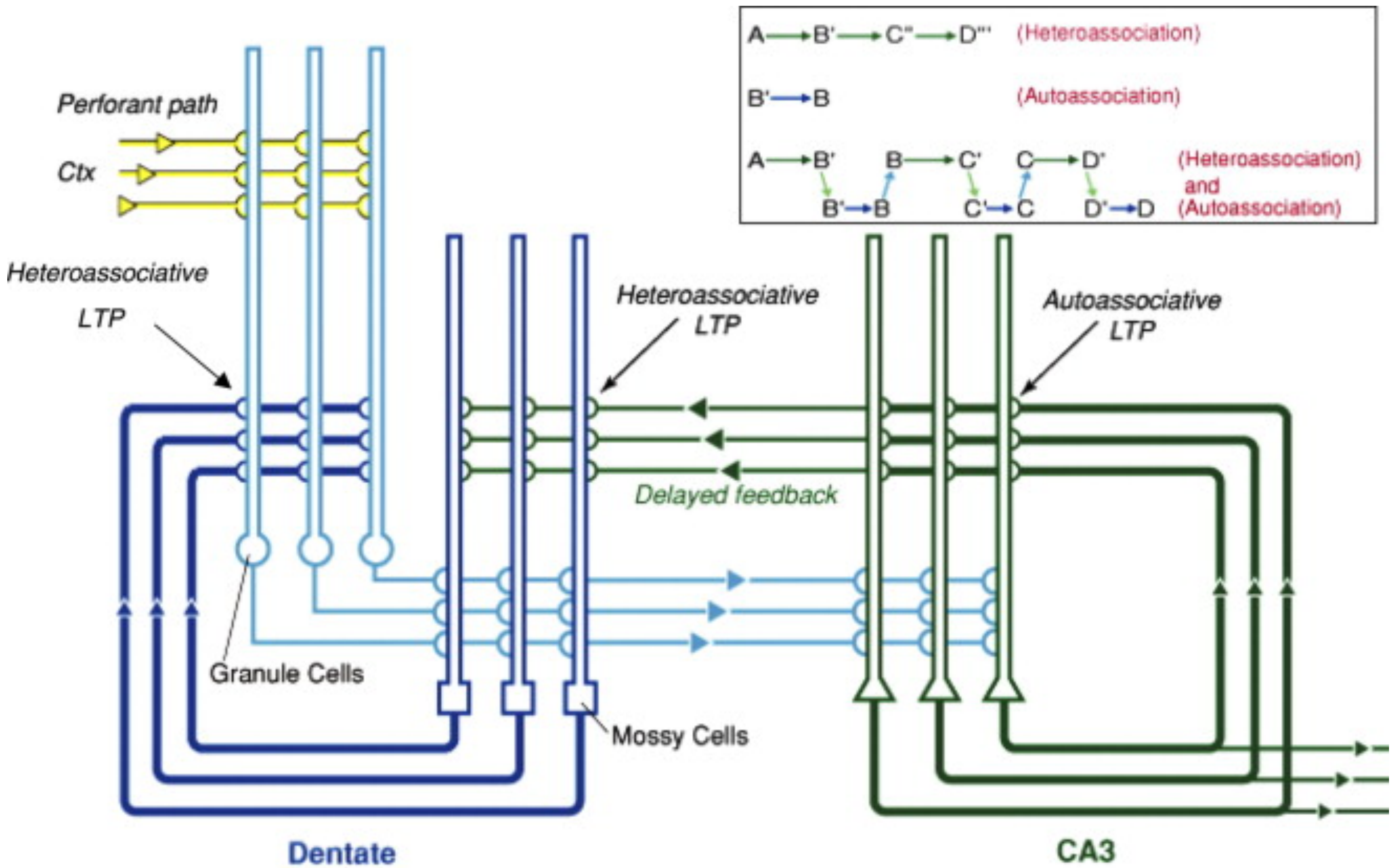
FROST: A Distributed Neurocomputational Model of Working Memory Maintenance

F. Gregory Ashby[1], Shawn W. Ell[2], Vivian V. Valentin[1], and Michael B. Casale[1]

Basal ganglia gated cortico-thalamo-cortical loops in working memory...

Differences with today's deep learning

Auto-associative and hetero-associative memories



Recall of memory sequences by interaction of the dentate and CA3: A revised model of the phase precession

John E. Lisman[a], Lucia M. Talamini[b, 1], Antonino Raffone[c, 2]

Differences with today's deep learning

Coordinating communication via oscillations?

Thalamus sets up synchronous oscillations in donor and recipient cortical areas, and this synchrony gates direct cortico-cortical information transfer between them

(adapted from [6]). Information is transmitted via the cortico–cortical connections to the next cortical region or regions, while the HO thalamic nuclei selectively activate the appropriate downstream cortical area that will be engaged in the next level of processing. Building from

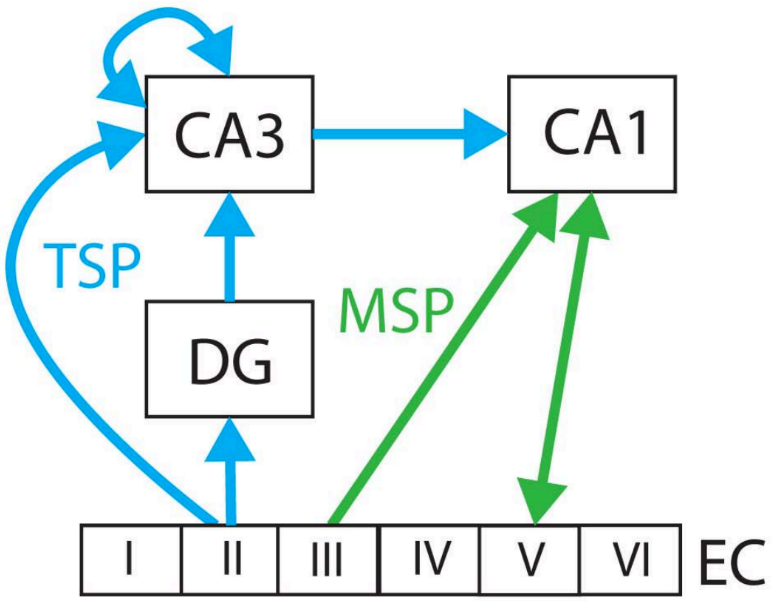
Thalamic pathways underlying prefrontal cortex–medial temporal lobe oscillatory interactions

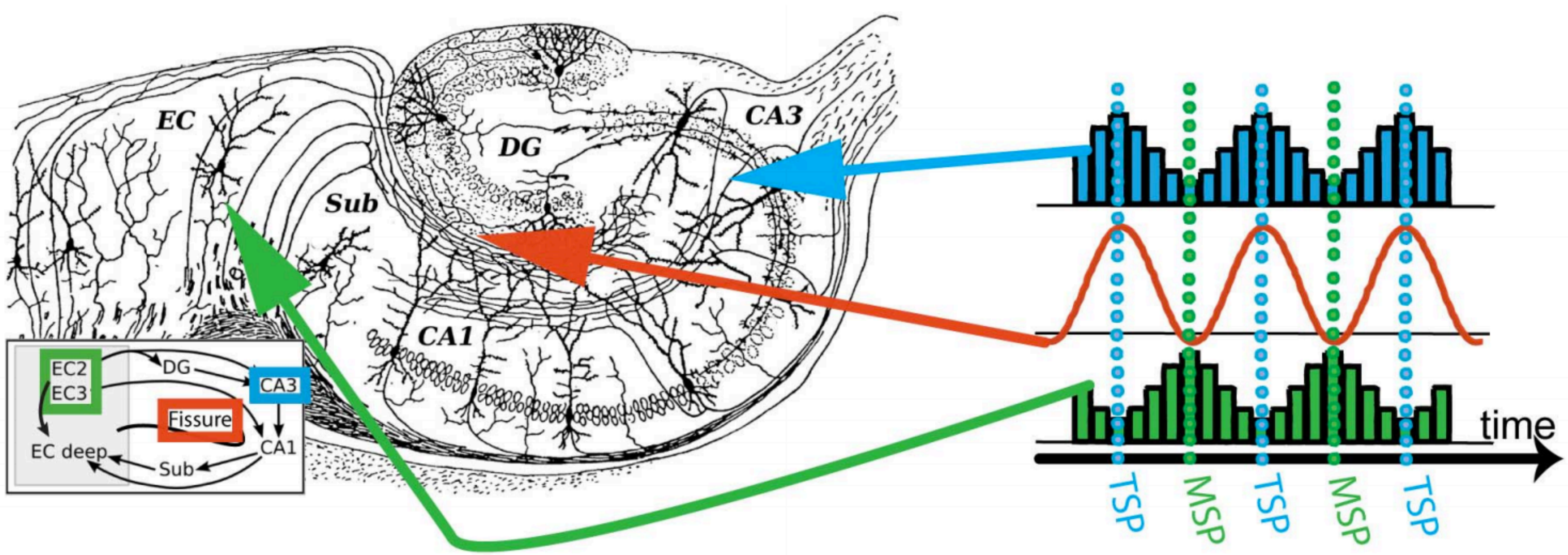
Nicholas A. Ketz, Ole Jensen, Randall C. O'Reilly

DOI: http://dx.doi.org/10.1016/j.tins.2014.09.007 | CrossMark

Differences with today's deep learning

Coordinating learning via oscillations?





Theta Coordinated Error-Driven Learning in the Hippocampus

Nicholas Ketz, Srinimisha G. Morkonda, Randall C. O'Reilly

TAKE HOME MESSAGES

We have no idea if the brain "can do backdrop", but also no reason to think it cannot

The end of the "representations + transformations" program?

- Neural representations are complex
  - You can find any almost any "tuning" (see Marius's lecture…)
- Neural computations are diverse

What if "understanding" should mean identifying:

- Architecture
- Cost Functions (as a function of area and time)
- Means of optimization

…rather than directly modeling how representations are transformed, i.e., rather than listing "atoms of computation"

But: <u>need</u> to understand the significance of key elements like

- Attractors, Oscillations, Diversity of Neurons/Synapses

Look to mesoscale anatomy for clues to architecture

- Patterns in mesoscale anatomy should have functional roles/explanations

with Konrad Kording & Greg Wayne

Thank You