LETTER Communicated by Terrence Sejnowski

Exploring the Architectural Biases of the Cortical Microcircuit

Aishwarya Balwani
abalwani6@gatech.edu
School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA

Suhee Cho
suheecho@stanford.edu
Department of Psychology, Stanford University, Stanford, CA 94305, USA

Hannah Choi
hannahch@gatech.edu
School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332, USA

The cortex plays a crucial role in various perceptual and cognitive functions, driven by its basic unit, the canonical cortical microcircuit. Yet, we remain short of a framework that definitively explains the structure-function relationships of this fundamental neuroanatomical motif. To better understand how physical substrates of cortical circuitry facilitate their neuronal dynamics, we employ a computational approach using recurrent neural networks and representational analyses. We examine the differences manifested by the inclusion and exclusion of biologically motivated interareal laminar connections on the computational roles of different neuronal populations in the microcircuit of hierarchically related areas throughout learning. Our findings show that the presence of feedback connections correlates with the functional modularization of cortical populations in different layers and provides the microcircuit with a natural inductive bias to differentiate expected and unexpected inputs at initialization, which we justify mathematically. Furthermore, when testing the effects of training the microcircuit and its variants with a predictive-coding-inspired strategy, we find that doing so helps better encode noisy stimuli in areas of the cortex that receive feedback, all of which combine to suggest evidence for a predictive-coding mechanism serving as an intrinsic operative logic in the cortex.

Aishwarya Balwani and Hannah Choi are the corresponding authors. Suhee Cho was at the Georgia Institute of Technology when this research was conducted.

Neural Computation **37, 1551–1599** (2025)
https://doi.org/10.1162/neco.a.23

1 Introduction

The brain is known to extensively implement hierarchical, predictive computations for both learning and inference. In particular, predictive coding theory (Rao & Ballard, 1999; Friston, 2005; Spratling, 2017) posits that the cortex relies on the differences arising between top-down predictions (made by areas higher in the hierarchy) and bottom-up sensory information (received by areas lower in the hierarchy) for efficient information processing and perception (Helmholtz, 1860; Gregory, 1968; Dayan et al., 1995; Rao & Sejnowski, 1999). Numerous experimental studies also support this theory; for example, when trained to perform a visual change detection task or when presented with multiple repeats of a sequence of images, for example, mice have been observed to exhibit heightened neural activity when viewing surprise inputs, with their neuronal representations varying appreciably in different cortical layers and areas depending on whether the stimuli are expected or unexpected (Garrett et al., 2020; Homann et al., 2022). Additionally, unexpected event signals have also been shown to predict subsequent changes in responses to expected and unexpected stimuli in both individual (Gillon et al., 2024) and populations of neurons (Wyrick et al., 2023), making a case for the neocortex indeed instantiating a predictive hierarchical model wherein unexpected events and error signals drive learning.

The anatomical substrate for these computations is the canonical cortical microcircuit (Douglas et al., 1989; Mountcastle, 1997; Hawkins, 2021), a structural motif that is shared across cortical areas and found in multiple mammalian species (Felleman & Van Essen, 1991; Douglas & Martin, 1991, 2004; Harris et al., 2019). Spanning the length of the cortical column, the microcircuit is defined by distinctive intra- and inter-areal layer-specific connectivity patterns; specifically, feedforward signals stemming from the shallow layers (layers 2/3) of the lower cortical area and targeting the granular layer (layer 4) of the higher area, along with feedback signals arising from the deep layers (layers 5/6) of the higher area, and projecting onto layers 2/3 and 5/6 of the lower area. In connection with predictive coding theory, these distinct layer-specific projection patterns are postulated to shape computations of expectations and surprisal along the cortical hierarchy. Furthermore, it is also suggested that layers receiving feedback from higher areas (i.e., 2/3 and 5/6) counter the driving feedforward signals and are therefore the location at which the error and difference computations occur (Friston, 2008, 2010).

Indeed, previous studies have provided indirect experimental evidence for prediction-related computations in various cortical layers and areas (De Kock et al., 2007; Sakata & Harris, 2009; Maier et al., 2010) and considered in an abstract sense the theoretical underpinnings of predictive coding (Maass et al., 2004; Nessler et al., 2013). They have also discussed their normative formulations (Bastos et al., 2012) and deliberated how they might arise in a Bayesian framework (Shipp, 2016; Sohn & Narain, 2021).

However, current understanding of how structural and computational primitives of the cortical microcircuit shape its representations to induce relevant encoding of errors and (un)expectedness of stimuli is limited (Kogo & Trengove, 2015; Bowman et al., 2023), with few works looking to explicitly map the processes of error-driven learning onto the cortical microcircuit (O'Reilly et al., 2021).

Given the current state of technology, in vivo efforts to experimentally test the underlying mechanisms of predictive coding still face enormous challenges; we therefore propose to explore these questions in silico, building on a body of work where deep neural networks have been successfully used as models of various neurobiological systems (Yamins et al., 2014; McIntosh et al., 2016; Cadena et al., 2019; Perich et al., 2020; Lindsey et al., 2019; Merel et al., 2019). However, unlike the normative approach typically taken by previous work studying predictive coding (Sacramento et al., 2018; Golkar et al., 2022), we do not start by assigning predetermined computational roles to possible anatomical structures. Instead, we focus on mapping how anatomical connectivity influences function and information representation within the cortical microcircuit.

Specifically, we look to isolate and understand the core computational effects of interareal and interlaminar connectivity and establish whether these basic anatomical patterns in themselves could give rise to predictive-coding-like behaviors under the assumption of error-driven learning, but without the added complexities of different cell types, their varied distributions, specific receptor types, or detailed dendritic morphologies that would increase model complexity while potentially obscuring the fundamental computational principles arising from connectivity patterns alone. To that end, we also restrict ourselves to the use of point neurons without spatial extent or dendritic compartmentalization: While pyramidal neurons in the cortex have extensive dendritic arbors with spatially segregated inputs (Larkum, 2013; Petreanu et al., 2009) and this segregation is thought to be important for integrating top-down predictions with bottom-up sensory information (Sacramento et al., 2018; Guerguiev et al., 2017), our simplification abstracts away these cellular details, allowing us to assess whether the computational principles of predictive coding can emerge from network-level connectivity patterns alone, before considering how cellular properties might further enhance these computations.

Employing an anatomically informed ensemble of recurrent neural networks (RNNs) to represent the canonical cortical microcircuit, we interrogate the following aspects of its role in learning: (1) Where are task-related variables encoded across the microcircuit? How do interareal feedback connections shape these neural representations? (2) Are there inherent performance advantages associated with the canonical architectural motif? and (3) Are the geometries and functionalities of layer-specific

neural representations primarily influenced by the architecture itself or more so by the training method used when learning the task?

By studying representations from multiple architectures with systematically altered structural motifs across different sequential learning tasks, we find that the inherent architectural hierarchy of the microcircuit naturally leads to differences in the geometries of representations across the network. We also observe that the combination of biologically consistent interareal connections and time delays in signal projections provides an intrinsic inductive bias for the segregation of expected and unexpected stimuli at initialization in layers that receive interareal feedback. Additionally, there is an emergence of further functional specialization of neuronal populations in terms of representing task-relevant variables over training. Finally, the inclusion of a predictive-coding-based training scheme further strengthens this modularization, specifically in the context of encoding unexpectedness in the stimuli. Taken together, these results suggest how connectivity structure and learning objectives in the cortical circuit shape representations of both expected and unexpected information, thereby facilitating predictive coding.

2 Models and Methods

In this section, we describe the canonical cortical microcircuit as observed in vivo, followed by the details of our anatomically constrained model in silico. We subsequently explain the structural modifications we make to our model to study the effects of interareal feedback on learning and performance in the microcircuit. We also provide brief descriptions of the tasks we train our models on and the methods we use to analyze the representations across the circuit over learning.

2.1 The Canonical Cortical Microcircuit: In Vivo and In Silico.

2.1.1 Standard in vivo Motif. Building on previous anatomical studies (Douglas & Martin, 1991; Felleman & Van Essen, 1991; Harris et al., 2019), following is the universal set of structural connections on which our anatomically constrained, in silico system is modeled:

- Interareal feedforward inputs are projected by superficial layers 2/3 in the lower area and received at layer 4 in the higher area (Usrey & Fitzpatrick, 1996).
- Interareal feedback is received in the lower area at layers 2/3 and 5/6 while being projected by layer 5/6 in the higher area.
- Within an area/cortical column, layer 2/3 projects to layer 5/6, layer 5/6 projects to layer 4, and layer 4 projects to layer 2/3.

2.1.2 Standard in silico Motif. Our typical model, which we call the corticalRNN (see Figure 1A), has a lower and higher processing area (or column), each of which uses an ensemble of three Elman RNNs (Elman, 1991). The internal state $\left(h^{(t)}\right)$ and output or readout $\left(y^{(t)}\right)$ of an RNN at time t are determined as

$$h^{(t)} = \sigma\left(W_{ih} \cdot x^{(t)} + W_{hh} \cdot h^{(t-1)} + b_h\right),$$
$$y^{(t)} = W_{hy} \cdot h^{(t)} + b_y. \tag{2.1}$$

where $x^{(t)}$ are the inputs received by the RNN at time t, W_{ih} is the input channel that projects $x^{(t)}$ onto the RNN's neuronal population, W_{hh} are the recurrent weights that specify the connectivity of neurons within the RNN, $h^{(t-1)}$ is the internal state of the network from the previous time step, and b_h is a bias term. W_{hy} projects the current hidden state h_t onto the appropriate output space, and b_y is another bias term. σ is a nonlinear activation function, (e.g., tanh). Every individual RNN models the dynamics of one of the cortical neuronal populations in superficial layers 2/3, granular layer 4, or deep layers 5/6.

Interareal connectivity that is, feedforward and feedback projections, as well as the intraareal connectivity (i.e., lateral projections within each area) between the various RNNs representing cortical layers are modeled by learnable projection matrices with fixed, sparse connectivity patterns sampled from a Bernoulli distribution. The directionalities of both intra- and interareal connectivity follow the descriptions laid out previously, while the relative sizes of the neuronal populations in the RNNs roughly correspond to the specifications established in the literature with respect to the mouse neuroanatomy, specifically the visual cortex (Shi et al., 2022). Additionally, interareal feedforward and feedback projections are incorporated by a recipient RNN after a delay of one and two time steps, respectively. These delays represent a simplified model of the temporal asymmetry observed in cortical processing between feedforward and feedback pathways (O'Reilly et al., 2014; Bastos et al., 2015). While our discrete time steps abstract away from precise millisecond timing, they capture the essential property that feedback signals arrive later than feedforward signals due to longer conduction paths, additional synaptic delays, and more complex processing requirements (Murray et al., 2014; Lamme & Roelfsema, 2000). All long-range interareal connections are strictly excitatory.

The data received and processed at every layer of the canonical microcircuit are described as follows:

- **Lower-area layers**
  LL4: $input \oplus proj(LL\ 5/6 \rightarrow LL\ 4)$[1]

[1] LL 4 = lower area layer 4; LL 5/6 = lowerarea layers 5 and 6. Similar shorthand used for higher-area layers, for example, HL 2/3 = higher-area layers 2 and 3. The symbol $\oplus$ denotes concatenation.

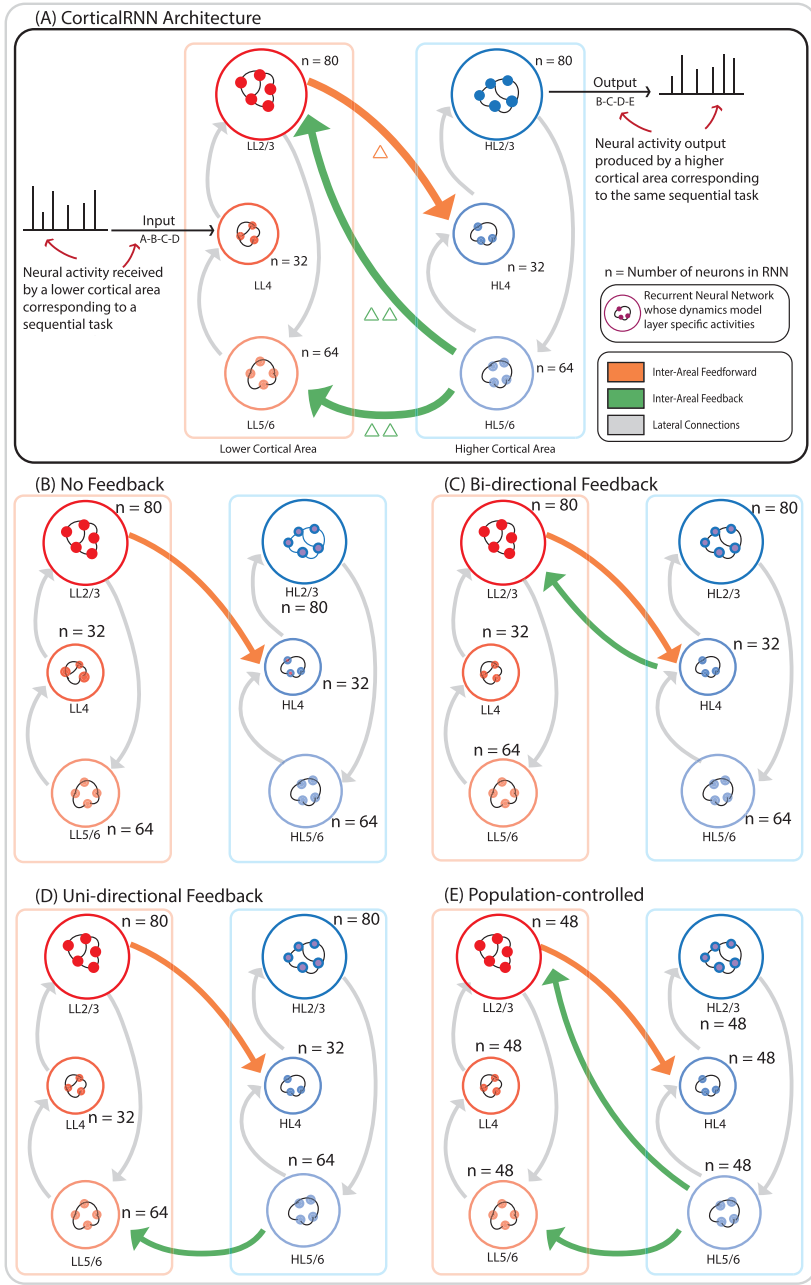

Figure 1: Experimental setup. (A) Architecture of the cortical RNN. Various arrows show the direction of connectivity across the microcircuit, while each Δ represents one time step of delay. (B–E) Structurally altered versions of the cortical RNN motif.

LL2/3: $proj(LL\ 4 \rightarrow LL\ 2/3) \oplus FB_a$
LL5/6: $proj(LL\ 2/3 \rightarrow LL\ 5/6) \oplus FB_b$

- **Higher-area layers**
  HL4: $FF \oplus proj(HL\ 5/6 \rightarrow HL\ 4)$
  HL2/3: $proj(HL\ 4 \rightarrow HL\ 2/3)$
  HL5/6: $proj(HL\ 2/3 \rightarrow HL\ 5/6)$

Here, $proj(X \rightarrow Y)$ represents the projection of activity from any layer X onto layer Y. The feedforward projection FF is defined as $proj(LL2/3 \rightarrow HL4)$. The two feedback projections FB_a, FB_b are defined as $proj(HL5/6 \rightarrow LL2/3)$ and $proj(HL5/6 \rightarrow LL5/6)$, respectively. The sign $\oplus$ is used to denote concatenation of inputs and/or features. Expressions for the data processing and computations at each of the layers are provided in appendix A.

2.1.3 Structurally Altered Motifs. To isolate the effects of the canonical interareal feedback connections on learning and performance in the microcircuit, we study the following variations of the motif:

1. **No interareal feedback** from the higher to the lower cortical area, that is, LL2/3 $\not\leftarrow$ HL5/6, and LL5/6 $\not\leftarrow$ HL5/6 (see Figure 1B),
2. A single, weighted, **bidirectional feedforward-feedback connection** between the lower and higher cortical areas (LL2/3 $\rightleftharpoons$ HL4) (see Figure 1C),
3. **Unidirectional feedback**, where the connection LL5/6 $\leftarrow$ HL5/6 is preserved while the connection from HL5/6 to LL2/3 (which in the canonical architecture could interact with the feedforward projection at LL2/3), is removed, thus; LL2/3 $\not\leftarrow$ HL5/6 (see Figure 1D),
4. The connectivity of the microcircuit is preserved, but the number of neurons across all RNNs is equalized, that is, all cortical layers have the **same-sized populations**, doing away with any physical compression and expansion of signals, within and across cortical areas (see Figure 1E)

Given that our standard model of the microcircuit comprises exactly two hierarchical areas, sensory inputs to the circuit are received at LL4 and outputs are produced at HL2/3. Following the neuroanatomy of the mouse visual cortex (Shi et al., 2022), the neuronal population ratio in layers 2/3, 4, and 5/6 is set to 5:2:4. Since our base model uses a scaling factor of 16, we assign 80, 32, and 64 neurons to these layers, respectively. All RNN weights at initialization are sampled from $\mathcal{N}(0, \frac{1}{n})$, where n is the number of neurons.

2.2 Tasks. Considering the temporal nature of the world and noting that this structure can be used to learn behaviorally relevant representations of stimuli (Recanatesi et al., 2021), all our tasks involve temporal predictions and follow the general structure of learning a sequence $\{x_i\}_{i=1}^{k}$ where $k \in \mathbb{N}$

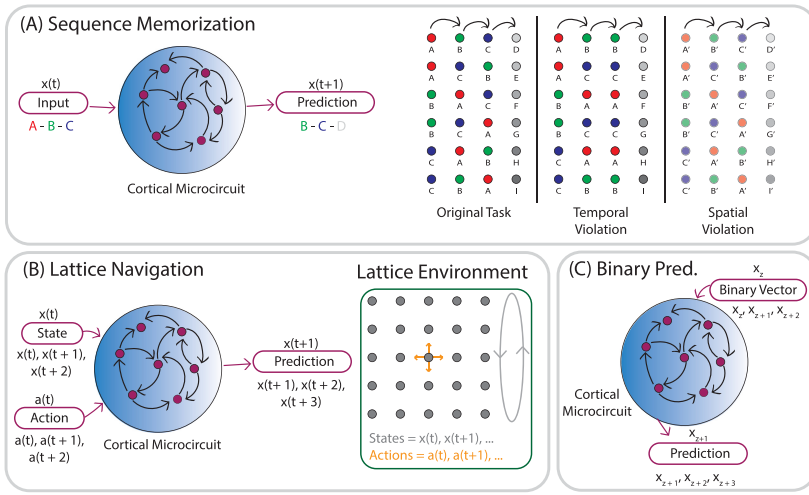

Figure 2: Task setup. (A) Schematic of sequence memorization task along with the graphical representations of our temporal and spatial perturbations. (B) Schematic of lattice navigation task. (C) Schematic of binary prediction task.

is the length of the sequence. More concretely, provided the input elements of the sequence $\{x_1, x_2, \ldots x_t \ldots, x_{k-2}, x_{k-1}\}$ one at a time, the circuit is required to predict the next element at each time step, that is, the sequence $\{x_2, x_3, \ldots x_{t+1} \ldots, x_{k-1}, x_k\}$. The training objective used is the mean squared error reconstruction loss.

$$\mathcal{L}_{recon}(x) = \sum_t ||\hat{x}_{t+1} - x_{t+1}||^2,$$

where $\hat{x}_{t+1} = y_{HL2/3,\,t} = W_{hy} \cdot h_{HL2/3,\,t} + b_{yHL2/3}$ is the output at HL2/3 at time t as per equation 2.1.

The three tasks in particular that we test on are (1) sequence memorization (see Figure 2A), where the microcircuit is trained to learn a given set of sequences, (2) lattice navigation (see Figure 2B), where the network needs to predict the next state on a grid given a current state and action; and (3) binary addition (see Figure 2C), where the microcircuit is given a binary input and is expected to increment it at every time step. Full descriptions of the tasks are provided in appendix B.

To study the microcircuit in the context of the predictive coding hypothesis and discern if any evidence of error computations can be found explicitly within the circuit, we test our networks on two types of out-of-distribution data (see Figure 2A) post training: (1) Data with temporal violations, where the sequence processed is $x_1 - x_2 - x_2$ instead of $x_1 - x_2 - x_3$, that is, the

element in the third position is unexpectedly replaced by a repetition of what it saw in the previous time step (2) data with an additive spatial noise resulting in the input sequence x'_1-x'_2-x'_3 where $x'_t = x_t + \eta, \eta \sim \mathcal{N}(0, \sigma),\ \sigma \in [0, 2]$.

All of our sequences incorporate a repetition of the input stimuli to ensure that every element of the sequence is processed for a sufficient length of time throughout the circuit. As a result, the microcircuit receives the input sequence $x_1 - x_2 - x_3 - x_4$ as $x_1 - x_1 - x_2 - x_2 - x_3 - x_3$ and is expected to predict $x_2 - x_2 - x_3 - x_3 - x_4 - x_4$ as its corresponding output sequence, where "$-$" denotes one time-step separation. Additional details regarding the specific implementations of all tasks are provided in appendix B.

2.3 Methods for Representational Analyses. This section gives a brief overview of the methods we use to analyze the geometric properties and informativeness of our representations. Additional explanations and implementation details for all our representational analyses methods are provided in appendix C.

2.3.1 Dimensionality Gain. As per the work of Recanatesi et al. (2021), for a given set of representations, we define their dimensionality gain (DG) as the ratio of their linear global dimension (L_{dim}) to their nonlinear local dimension (NL_{dim})—$\mathrm{DG} = \frac{\mathrm{L}_{dim}}{\mathrm{NL}_{dim}}$. Intuitively, the measure should capture whether the representations are encoding simple or complex task-relevant concepts and if they are doing so in an efficient manner. Moreover, DG of a complex yet structured concept would be high (see Figure 3B, left), while that of something without intrinsic structure, such as noise, would be low (see Figure 3B, right). Intuitively, this can be reasoned about as follows:

- If a complex concept is being encoded by a set of neurons, their linear, or global, dimension is high as the data occupy much of the neuronal space available to them. However, since the information in the representations is relevant to the task, there is structure reflected in them that is captured by a nonlinear transformation, resulting in a low nonlinear, or local, dimension. As a result, the DG of such a concept is high (see Figure 3B, left).
- If a concept is simple, both its linear and nonlinear dimesionalities are low, making the DG low as well.
- If the information encoded in the representations has no real structure, both the linear and nonlinear dimensionalities tend to be high as the representations seemingly occupy all of the ambient space, again making the DG low (see Figure 3B, right).

As a result, we can think of increasing DGs to be a sign of learning, with higher values corresponding to more complex phenomena, while lower

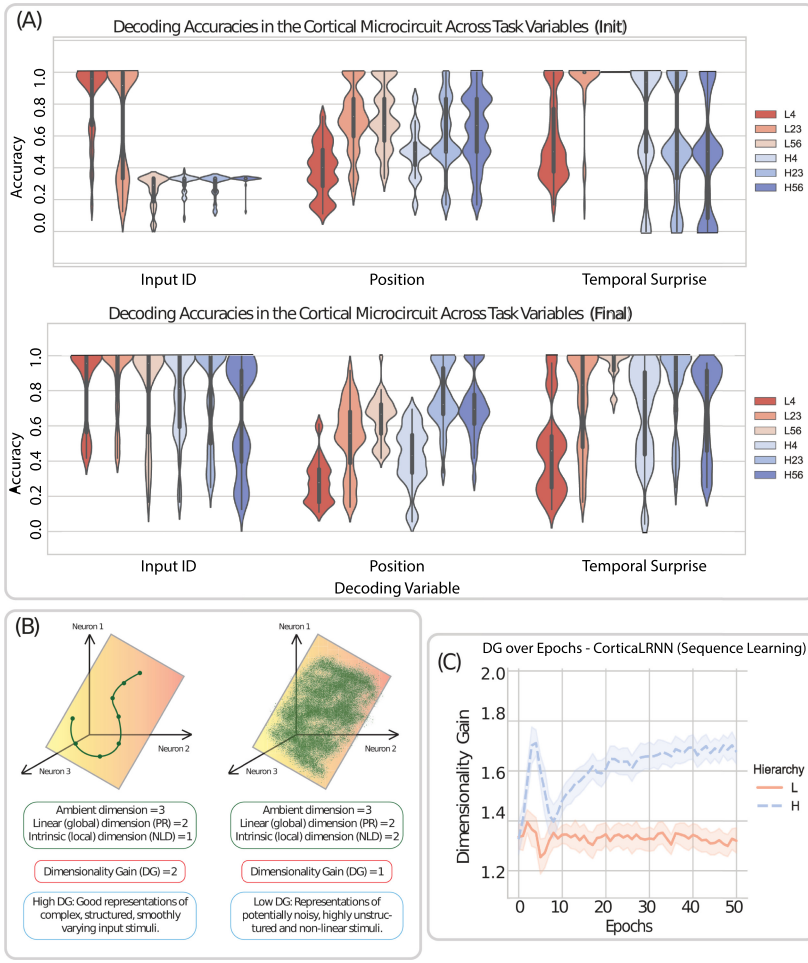

Figure 3: Architectural biases of the cortical RNN for the sequence memorization task. (A) Decoding accuracies of task-relevant variables at various layers in the cortical RNN at initialization (top), and after training (bottom). (B) Schematic showing dimensionality gain (DG) in the case of a complex yet structured (left) versus unstructured concept (right). (C) DG of the cortical RNN during learning of the sequential learning task, split by layers that form the lower cortical area (L: solid line for mean, red) and the higher cortical area (H: dashed line for mean, blue; shaded area implies variance). For the purposes of exposition, results in this figure are restricted to the case of the sequence memorization task. Results corresponding to the lattice navigation task and binary addition task are provided in appendix D.

or decreasing DGs indicate lack of structure or noise being encoded in the representations. Illustrative results on some simulated data can be found in appendix C.

We do note, however, that despite its intuitive appeal, the DG measure comes with certain challenges. Its effectiveness hinges on reliably estimating both the linear and nonlinear dimensions, and certain biases or insufficient samples in the data can lead to ambiguous estimates and incorrect interpretations of what a given DG value signifies. While higher DG values are generally interpreted as indicators of richer and more structured neural representations, this relationship is multifaceted; such increases might also reflect redundancies or distributed coding properties that do not straightforwardly translate to complexity in task-relevant information.

2.3.2 Decodability of Task Variables. Another way to measure how informative a set of representations is about a concept is by assessing their predictiveness, or equivalently, how linearly separable different "classes" relevant to the specific concept are from each other in high-dimensional space. The concepts we test decodability for are (1) input ID, that is, which input is being processed by the microcircuit (e.g., x_1 versus x_2); (2) position, that is, where in the sequence you are (e.g. the first position in the sequence versus the second position in it); and (3), surprise, whether the input being processed is expected (e.g., expected input x_2 presented as a part of a learned sequence versus unexpected input x_2 presented as a repeat that violates a learned sequence)

2.3.3 Assessing Neuronal Population Selectivity. Complementing the analyses studying dimensionality of neuronal representations, we also quantify the selectivity of different neuronal populations for a particular concept. To do so, we estimate the minimum number of neurons required to predict a concept from a given set of representations, using the L_1 regularized version of a linear classifier. Subsequently, requiring fewer neurons to predict a concept (i.e., separate classes relevant to that concept) post training with high accuracy can be identified as being more specialized, or tuned to that particular concept.

This approach is grounded in established literature demonstrating that specialized neural populations often encode relevant information using fewer, more selective neurons (DiCarlo & Cox, 2007; Olshausen & Field, 2004). Studies of the visual cortical hierarchy have shown that higher areas achieve greater selectivity for complex features using increasingly specialized neuronal subsets (Rust & DiCarlo, 2010), while work in auditory cortex has demonstrated similar principles of sparse, specialized encoding (Hromádka et al., 2008). This also aligns with findings that information about specific features becomes more concentrated in dedicated

subpopulations as it progresses through neural processing hierarchies (Rigotti et al., 2013).

3 Architectural Biases and Learning in the Canonical Cortical Microcircuit

Given the construction described in section 2.1, in this section we study the effects of the explicit anatomical structure (i.e., interareal feedforward and feedback projections) observed in the canonical cortical microcircuit. All results provided are averaged over five independent runs of each experiment, performed on a Dell Precision 7920 Tower with an Nvidia Quadro RTX 5000 GPU.

3.1 Evidence of Functional Modularization. Observing the decoding accuracies in various layers of the cortical RNN for the sequence learning task, we see that at initialization (see Figure 3A, top left), the identity of the input is highly decodable in the lower area in LL4 and LL2/3, while after training (see Figure 3A, bottom left), decodability of the input identity increases to high levels across the microcircuit. In the case of decoding the position of the received input vector in the sequence, accuracies are greater in layers higher in the hierarchy, particularly post training (see Figure 3A, bottom middle). Finally, in the case of decodability of surprise, we note that right at initialization, decodability is extremely high for layers LL2/3 and LL5/6, which both receive feedback from the higher area (see Figure 3A, top right). Post training, accuracies improve to some degree for the layers in the higher areas as well (see Figure 3A, bottom right); however, the highest accuracies observed are still at layers LL2/3 and LL5/6 where feedback projections from the higher area directly targeted, overall in agreement with the predictive coding hypothesis. These results collectively make a case for the idea of functional modularization within the microcircuit (with similar results for the other two tasks also provided in appendix D), in that they align with the notion that lower cortical areas have neuronal representations that correspond to lower-level or "simpler" concepts, that is, information that is directly related to the input itself (such as the identity of the input vector or state being received) and higher cortical areas have neuronal representations that are better tuned to more complex task-related concepts derived from combining different pieces of information corresponding to the task (such as the temporal position of an input within a sequence). These results are further supported by the dimensionality gains (see Figure 3B) that we see averaged over the populations in the lower and higher processing areas separately (see Figure 3C). Over the course of learning, we note that the average DG of the lower area remains relatively constant, implying that it encodes simpler information that has a constant ratio of ambient and latent dimensionality in neuronal space. That of the higher area increases with learning, implying that it encodes information that is higher dimensional

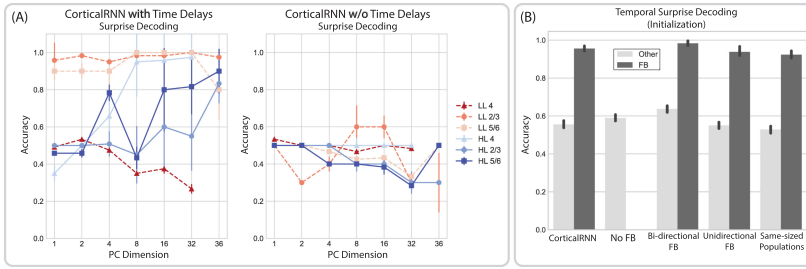

Figure 4: Temporal surprise decoding in the cortical RNN for sequence memorization. (A) Decoding of temporal surprise at different layers of the cortical RNN with (left) and without (right) time delays, using the principal components of their representations at initialization. (LL4, HL4 have 32 neurons and therefore have the maximum PC dimension of 32.) (B) Effects of interareal time delays on decodability of surprise at initialization, in layers receiving interareal feedback (dark gray) versus not (light gray), in the cortical RNN and its structural variants.

in the ambient sense while still being structured and encodable with few latent variables.

We also extended our two-area model to include an additional cortical area, yielding a model of three hierarchically related cortical areas. Preliminary experiments with the three-area model also yield similar results (see appendix J, Figures 15 and 16).

3.2 Inductive Biases due to Interareal Feedback and Time Delays in the Cortical Microcircuit at Initialization. Further delving into the results, we obtain for decoding task-related concepts, we observe that areas receiving interareal feedback: LL2/3 and LL5/6 are particularly adept at decoding surprise or unexpectedness in the sequence memorization task. Moreover, they do so with high accuracy not only post training, but at initialization as well (see Figures 4A, left, and 4B). We posit that this result is a consequence of the presence of timedelays in the projection of interareal communication and the easily separable nature of our stimuli. More concretely, we hypothesize and subsequently show that the microcircuit exploits the sequential nature of the task and separability of the input as follows:

1. The first three elements of our expected sequences are always permutations of the inputs x_1, x_2, x_3, which ensures that all inputs are seen as both expected and unexpected in the same temporal position as part of some sequence. For example, both $x_1 - x_2 - x_3 - x_4$ and $x_1 - x_3 - x_2 - x_5$ belong to the set of expected sequences. The standard sequence, $x_1 - x_2 - x_{3e} - x_4$, has the input x_{3e} as expected in the third position, but the sequence $x_1 - x_3 - x_{3v} - x_5$ is temporally

violated from its natural form $x_1 - x_3 - x_2 - x_5$ and sees the input x_{3v} unexpectedly, again in the third position. (Note that $x_3 = x_{3e} = x_{3v}$ in input space; we make the distinction between them simply for the purposes of exposition.)

2. Decoding surprise in the sequence memorization task requires the ability to find a separating hyperplane between inputs that are received as expected in a sequence versus those that are received unexpectedly, that is, being able to distinguish x_{3e} from x_{3v} in neuronal space with a separating hyperplane.
3. When interareal time delays are present in the circuit, the representations formed in neuronal populations that receive feedback involve a structured integration of information from different time points. Specifically, these populations simultaneously process current bottom-up inputs alongside top-down feedback signals that are derived from inputs presented two time steps prior. This temporal offset creates a natural mechanism for comparing predictions with current sensory data.
4. In contrast, without these explicit interareal delays, even though recurrent connections maintain some temporal information, the circuit lacks the systematic temporal separation between bottom-up and top-down processing streams. As a result, all levels of the circuit primarily integrate information from the same temporal context, preventing the natural emergence of prediction error computations that depend on comparing current inputs with predictions based on past states.
5. We provide repeated presentations of the inputs to the microcircuit to represent their duration: our input sequences are $x_1 - x_1 - x_2 - x_2 - x_{3e} - x_{3e}$ and $x_1 - x_1 - x_3 - x_3 - x_{3v} - x_{3v}$. Consequently, in the presence of time delays, the neuronal populations LL2/3 and LL5/6 process a combination of the inputs x_2, x_{3e} in the case of the standard sequence and x_3, x_{3v} in the case of the violated sequence at the third temporal position. In the absence of time delays, at the third temporal position, the microcircuit simply processes x_{3e} in the standard case and x_{3v} in the temporally violated case at all layers.
6. Noting that all the weights of the microcircuit are gaussian distributed at initialization and that there are no projections that lead to any drastic compression or expansion across layers or areas (i.e., by an order of magnitude or greater), the relative distances between inputs in the input space are largely preserved in the representational space, as per the Johnson-Lindenstrauss lemma (Dasgupta & Gupta, 2003; Arriaga & Vempala, 2006).
7. Assuming that the inputs are fairly separable in input space, this explains why populations that receive interareal feedback can distinguish an expected input from an unexpected one in the presence of time delays; these layers have representations that easily allow a sep-

arating hyperplane between representations of the form $\approx x_2 + x_3$ for the expected x_{3e} versus those of the form $\approx x_3 + x_3$ for the unexpected x_{3v}. However in the absence of time delays, this would not be the case as the representations to be separated would both correspond to the same input $\approx x_3$ for both the expected and unexpected x_3.

More formal justifications supporting our hypothesis and subsequent rationale are provided in appendix E, where for the sequence memorization task we show the following:

Theorem 1. *Decodability of Bernoulli-distributed sequential stimuli at initialization in the canonical cortical microcircuit. Let* $\mathbf{x}_t \sim \mathcal{B}(p)$ *be the input Bernoulli vectors for the sequence memorization task where p denotes the probability that any component of* $\mathbf{x}_t$ *is* 1*, and* $\mathbf{x}_t \in \{0,1\}^d$ *with d being the vector dimension. Assume that the initial weights of the cortical microcircuit are drawn from* $\mathcal{N}(0,1)$*. Then there exists a multiplicative distortion factor* $\epsilon \in (0,1)$*, representing the maximum allowable relative error in pairwise distances after projection, such that the input vectors* $\mathbf{x}_t$ *are separable in the representation space, provided the projection dimension k satisfies*

$$k \geq \frac{4}{\epsilon^2(1-\epsilon)} \ln\left(\frac{2\sqrt{\pi}}{\sqrt{\pi} - 2\sqrt{1-p}}\right).$$

Proof. Our argument proving the above statement relies on the following logic:

- *Separability:* The input vectors are separable in input space with expected distance $2dp(1-p)$. This separability is preserved for expected sequences but not for temporal violations.
- *Probability of linear separability:* For vectors separated by the distance δ, the probability that a random vector can act as a linear separator increases with the distance δ, given by $p = \frac{2\delta}{\pi} + O(\delta^3)$ as $\delta \to 0$.
- *Dimensionality constraint:* To preserve the separability in neuronal space, given an acceptable distortion ϵ, by the Johnson-Lindenstrauss lemma, the projection dimension k must satisfy

$$k \geq \frac{4}{\epsilon^2(1-\epsilon)} \ln\left(\frac{2\sqrt{\pi}}{\sqrt{\pi} - 2\sqrt{1-p}}\right)$$

The detailed proof is provided in appendix E. □

Our empirical results (see Figure 4) validate our intuitions and theoretical justifications, where we see that it is indeed the structure of the stimuli in input space, sequential nature of the memorization task, and intrinsic time delays in the microcircuit that explain our results (see Figure 4A). Furthermore, the ability to distinguish temporal violations with high accuracy is specific to layers that receive interareal feedback across different types

of architectures (see Figure 4B), making a strong case that the surprise encoding in accordance with the predictive coding hypothesis can arise from the circuit structure itself. While we make the above argument in the case where our stimuli are highly separable in input space, the idea truly is more general and would hold in whichever cortical area the representations corresponding to a stimulus are distinguishable. Hence, these results suggest a general motif of cortical computation underlying the efficient information processing in sequential tasks that is often observed across various natural environments and animals (Jiang & Rao, 2024).

Finally, we note that architectures receiving feedback are slightly more robust than the no-feedback architecture to temporal violations post training (see appendix F). When given a temporally violated input sequence, they still produce outputs that correspond to the original sequence more consistently. However, in the presence of spatial noise, we find no significant differences in the performance across the various architectures across tasks.

4 A Predictive-Coding-Inspired Training Strategy

Whereas the previous section focused on the effects of structural priors (i.e., anatomical constraints) on information processing within the canonical microcircuit, we now examine the impact of imposing a functional prior during learning. Specifically, we test whether integrating principles of predictive coding alongside the microcircuit's structure provides evidence supporting the idea that areas receiving interareal feedback are responsible for error computation, by incorporating them into our training scheme. Our training strategy inspired by hierarchical predictive coding (Rao & Ballard, 1999) involves two distinct phases, each updating the model with different loss functions. In the first phase, as before, the entire network is trained using a reconstruction-based loss of the form

$$\mathcal{L}_{recon} = ||\hat{x}_{t+1} - x_{t+1}||^2.$$

In the second phase, only parts of the network responsible for generating and transmitting prediction signals are trained. These are the higher cortical layers $HL2/3$, $HL4$, $HL5/6$, and feedback connections W_{FB_a} and W_{FB_b}. The training now uses a predictive-coding (PC) loss, calculated as the average of the prediction errors from the two feedback projections FB_a and FB_b as defined in section 2.1. Each prediction error is calculated as the difference between the signal conveyed through the feedback connection and the neuronal representation of the layer to which the signal is directed. Specifically, we minimize

$$\mathcal{L}_{PC} = \frac{1}{2}||W_{FB_a} \cdot h_{HL5/6,t} - h_{LL2/3,t}||^2 + \frac{1}{2}||W_{FB_b} \cdot h_{HL5/6,t} - h_{LL5/6,t}||^2.$$

We then alternate between the two phases so that parameters in the higher areas are encouraged to be predictive of the activity in the lower area. The explicit training algorithm is provided in appendix G.

Furthermore, during the second phase of training, where the lower cortical activities are fixed, we assume that they are conditionally independent given the higher cortical activity and their respective feedback weights. Under this assumption,[2] the PC loss objective can be derived by maximizing the likelihood of the lower cortical activity with respect to the higher cortical activity and feedback weights, using the same principles as normative predictive coding (see a appendix H).

Finally, we note that in the absence of feedback connections, the predictive loss simply acts as a Tikhonov regularizer on the activities of the relevant lower-area populations, effectively promoting sparse coding (Olshausen & Field, 1997). In other words, in the no-feedback case, we set both W_{FBa}, W_{FBb} to zero, which simplifies the loss to $\mathcal{L}_{PC} = \frac{1}{2}||h_{L2/3,t}||^2 + \frac{1}{2}||h_{L5/6,t}||^2$.

4.1 Effects of Predictive-Coding-Inspired Training on Representational Dimensionality across Layers and Areas. Studying DG across training in various layers of the microcircuit when using only a reconstruction-based loss shows no appreciable differences in DGs across various architectures (see Figure 5A). The DGs of layers in the higher processing area are greater than those in the lower area, and this trend is maintained across all architectures. Moreover, addition of the PC loss does not affect the general trend of DGs across areas for architectures that have no feedback connection from the higher area to LL5/6 either (Figure 5B, Top - (ii), (iii)). However, addition of the PC loss significantly changes the DG of LL5/6 in architectures where the layer does receive feedback (Figure 5B, top, (i), (iv), (v)). In particular, we note that in these cases, the DG of LL5/6 drops with training, implying that the representations in the layer in these cases have linear and nonlinear dimensionalities that grow in tandem. This consequently points to the idea that the phenomenon being encoded in LL5/6 lacks smooth structure and predictability. We therefore hypothesize that LL5/6 in these cases encodes "surprise" as posited by predictive coding theory, which by definition is unstructured and unpredictable.

Certain experimental studies have also found evidence supporting such a hypothesis by noting that there is an intermixed representation of predictive neurons in layers 2/3 and 5/6 that would make surprise linearly decodable from both these populations (Audette et al., 2022; Gillon et al., 2024;

[2] While this assumption simplifies our derivation, in practice, the shared higher cortical activity and the use of backpropagation introduce some dependencies between the updates of the feedback weights. Despite this, treating the lower cortical activities as conditionally independent provides a reasonable approximation that aligns with the theoretical framework of predictive coding.

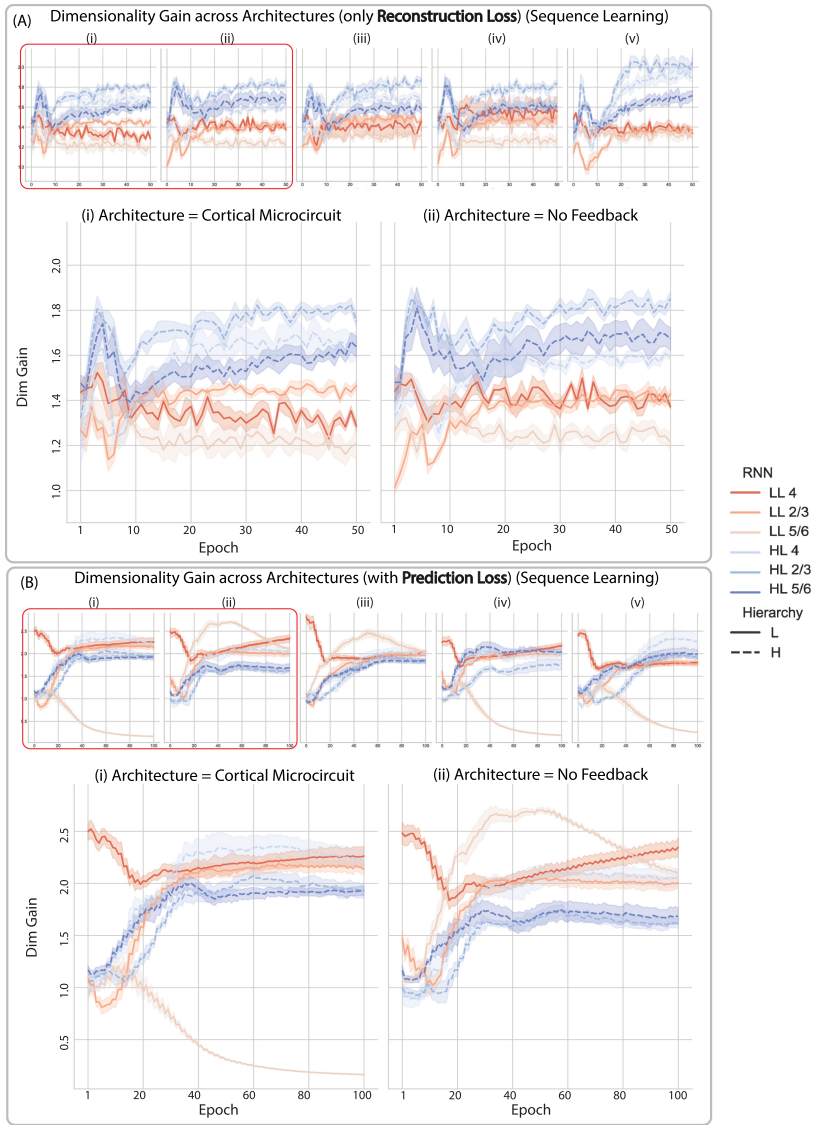

Figure 5: Effects of Using Predictive-Coding-Based Training Objective on Dimensionality Gain. Panels A and B are juxtaposed to show the differences that arise in the DG of various layers across learning (A) without and (B) with the inclusion of a predictive-coding-based objective, respectively. In both, the top row shows DG across layers for architectural variations in the following order: (i) canonical cortical microcircuit, (ii) no feedback, (iii) bidirectional feedback, (iv) unidirectional feedback, and (v) population-controlled microcircuit. The lower panels zoom into the DGs of the (i) cortical microcircuit and (ii) no-feedback architectures.

Wyrick et al., 2023), with it sometimes being primarily decodable from layer 5/6 (Audette & Schneider, 2023). Likewise, other computational works have also drawn similar conclusions, suggesting layers 5 and 6 of the lower cortical area as the site encoding surprise (Cain et al., 2016; Barry & Gerstner, 2024; Kermani Nejad et al., 2024). In particular, work by O'Reilly et al. (2021) proposes a related but distinct mechanism for predictive learning in the neocortical hierarchy, wherein they suggest that the pulvinar nucleus of the thalamus serves as a crucial substrate for computing prediction errors. In this model, the error signal emerges as a temporal difference between prediction and outcome states in the pulvinar, with layer 6 corticothalamic (6CT) neurons generating top-down predictions and layer 5 intrinsic bursting (5IB) neurons providing the actual outcome signal via driver inputs. The 5IB neurons exhibit rhythmic bursting at alpha frequency (∼10 Hz), creating a natural temporal difference between prediction and outcome phases that drives learning. This temporal difference mechanism differs from our approach, which proposes a direct encoding of surprise in deep layers (particularly 5/6) of the cortical microcircuit itself. While our model emphasizes the functional specialization of cortical areas and laminar-specific connections for error computation, the framework proposed by O'Reilly et al. (2021) highlights the thalamocortical circuit's role in generating a temporally structured error signal. Both approaches, however, converge on the importance of feedback projections and the potential role of layer 5/6 neurons in predictive coding computations, suggesting complementary mechanisms that may operate in parallel within the cortical hierarchy.

Furthermore, this encoding of surprise in the area is facilitated by both the physical feedback connection as well as the training objective, and therefore remains conspicuously absent in the dimensionalities of representations in LL5/6 of the architecture without any feedback connections (see Figure 5, top: (ii) and (iii)). While Figure 5 shows these trends in the sequence learning task, we find that these results hold consistently across other tasks as well (see appendix I). Results with a three-area model follow a similar trend (see appendix J), where adding a PC-based loss leads to a drop in DG for layer 5/6 in the intermediate and lower areas (see Figure 19) compared to those observed with just the reconstruction loss (see Figure 18).

4.2 Effects of Predictive-Coding-Inspired Training on Neuronal Specialization. Following our observations of how the DG changes in LL5/6 when trained with the PC loss, we looked to verify our hypothesis that this was indeed caused due to the neurons in the population encoding surprise. To do so, we checked the neuronal selectivity for temporal surprise in the neurons. If a sparse set of neurons can effectively distinguish surprising elements from expected ones, it can be inferred that the concept of surprise is efficiently represented in that neural population and is a primary driver of their activity. We find that our results support the hypothesis, and that

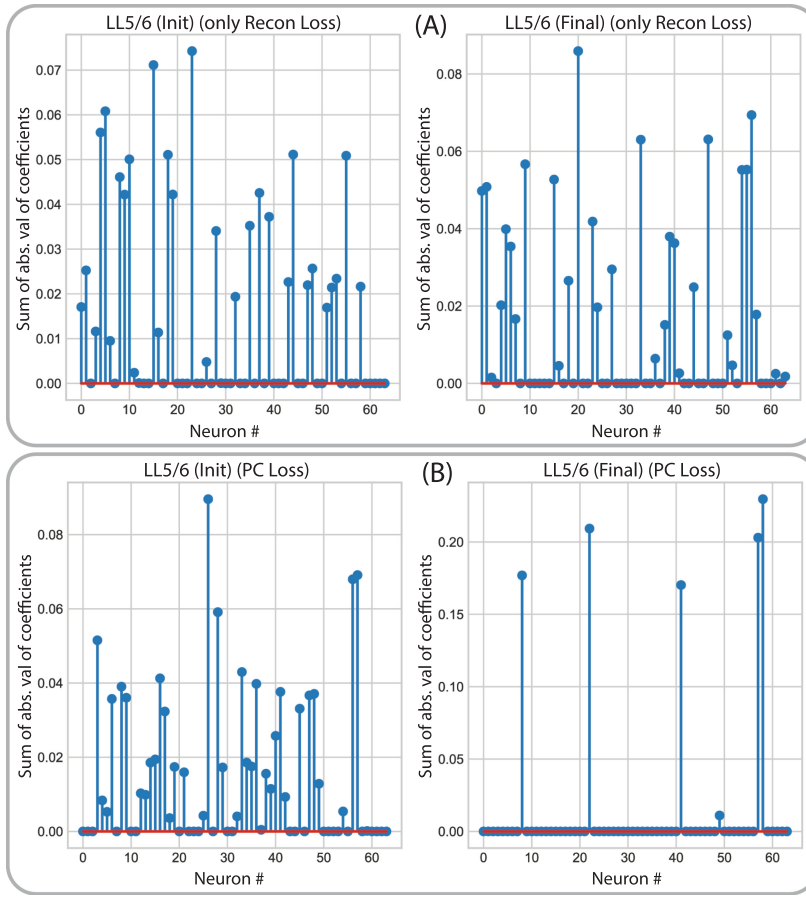

Figure 6: Effects of using the predictive-coding based training objective on neuronal selectivity. Examples of neurons used to decode surprise in LL5/6 (A) without and (B) with the PC loss, both before (left) and after (right) training.

the addition of the PC loss results in fewer neurons being required to robustly encode surprise in LL5/6 (see Figure 6B, right) as compared to when not using the loss (see Figure 6A, right), but approximately the same number of neurons are needed to do so at initialization in both cases (left, Figures 6A and B). The results hold over multiple runs and instantiations (see appendix K).

Discussion and Conclusion

Our work provides compelling evidence that while the architectural constraints of the canonical cortical microcircuit primarily shape information

representation within it, functional priors, in terms of how the network is trained, play a key role in its learning and the geometry of its neuronal representations. Specifically, we find that (1) the structural primitives (i.e., feedback projections) of the network and the nature of sequential tasks, combined with the inherent time delay typical of interareal cortical projections, can strongly influence which areas best represent certain task-relevant concepts, and (2) both physical structure and training objective can independently influence the dimensionalities of representations across the microcircuit. While the physical structure induces an architectural bias toward error encoding via feedback projections (in accordance with hypotheses made by hierarchical predictive coding), it is the training objective and procedure that imposes the functional prior and further strengthens the ability of neurons in layers of the lower cortical area that receive direct feedback projections to encode surprise.

That said, there remain a number of open questions and avenues for further investigation. For one, our work points out the importance of time delays, an aspect that has traditionally been understudied in the context of predictive coding, in the microcircuit's ability to identify violations and encode surprise, especially at initialization. Inquiry along these lines could potentially lead to deeper insights into how temporal dynamics contribute to error propagation and prediction updates in cortical circuits.

We also note that the nature of our violations might play a role in how the predictive errors they manifest are represented across the microcircuit; our violations correspond to global oddballs in that the violating stimuli are not novel because the network has never seen them before, but unexpected because they defy expectations while repeating the local context. Recent work by Westerberg et al. (2024) has found that unlike local oddballs (i.e., violations that are completely new to the network), which are largely compliant with conventional predictive coding theory and emerge early in superficial layers 2/3, global oddballs emerged in nongranular layers, a finding that is in alignment with ours. These results therefore warrant further exploration along the lines of stimulus dependency in error encoding, a notion also supported by the work of Furutachi et al. (2024), and rethinking of the normative predictive coding computations themselves by now taking the type of predictive error into account.

Architecturally, we recognize several limitations of our current model. First, we note that we primarily study the microcircuit with two hierarchically related areas, with only a preliminary expansion to three areas. How these results scale with increasing hierarchical depth, such as in the extensive hierarchies observed in primate visual cortex (Felleman & Van Essen, 1991) and mouse visual cortex (Harris et al., 2019), remains an open question that merits further investigation.

Second, our model uses simplified point neurons without dendritic compartmentalization or spatial extent. Experimental evidence demonstrates that pyramidal neurons integrate feedforward and feedback inputs in

specialized dendritic compartments, with apical dendrites receiving feedback and basal dendrites receiving feedforward signals (Mikulasch et al., 2023). These segregated inputs enable unique computational properties through active dendritic processing (Takahashi et al., 2020) that may be crucial for implementing predictive coding operations at the cellular level. Computational models have suggested that such dendritic segregation could support error-driven learning algorithms similar to backpropagation, offering a potential biological implementation of prediction error computation that our current model cannot capture. Specifically, work by Sacramento et al. (2018) demonstrated how apical dendrites of pyramidal neurons could compute prediction errors locally by comparing top-down predictions with bottom-up activity, while Guerguiev et al. (2017) and Richards & Lillicrap (2019) showed how this dendritic compartmentalization allows neurons to effectively implement credit assignment during learning in hierarchical networks. While our network-level approach reveals important computational properties emerging from interlaminar connectivity patterns, future models incorporating dendritic computation could provide additional insights into how error signals are computed and propagated at the cellular level.

Third, incorporating additional biological constraints such as distinct excitatory and inhibitory cell types, Dale's law, more realistic synaptic dynamics, or biologically plausible local learning rules would further enhance model realism. The addition of these constraints could significantly affect network dynamics and neuronal interactions, potentially affecting the sparsity and compositional nature of the learned representations. For instance, distinct inhibitory interneuron classes may play specialized roles in predictive coding computations (Keller & Mrsic-Flogel, 2018; Spratling, 2017; Hertäg & Clopath, 2022), helping to compute prediction errors or gate prediction signals.

Nevertheless, the promising results from our simplified model suggest that many fundamental aspects of hierarchical prediction can be understood from interareal- and interlaminar-level connectivity patterns alone. Future work that systematically incorporates these additional biological details will undoubtedly provide richer insights into how cellular and network properties interact to implement efficient predictive processing in cortical circuits and also perhaps reveal how biological constraints enhance the robustness and efficiency of predictive coding implementations beyond what is possible with our current simplified architecture.

Finally, we also note that our work assumed the canonical microcircuit structure and studied learning in it subsequently, but it will be interesting for future expansions to identify physiological phenomena (e.g., energy efficiency, Hebbian plasticity, critical developmental periods) that might constrain development in the cortex and reverse-engineer (through, e.g., sparsity penalties, unsupervised learning rules, pruning schemes) how this structure might emerge organically. Answers to these questions would go a

long way in not only understanding the brain, but could also pave the way for more efficient, robust, and interpretable artificial intelligence systems by leveraging the appropriate architectural and functional priors.

Appendix A: Layerwise Data Processing and Computations at an Arbitrary Time Step

A.1 Notations. We define the variables used in our computations:

- $x^{(t)}$: Sensory input to the microcircuit at step t
- $u^{(t)}$: The total input to layer X at time t after input processing through W_{iX}
- W_{iX}: Input projection matrix at layer X
- $h_X^{(t)}$: Neural activity at layer X at time t
- W_X: Recurrent weights at layer X
- W_{iX}: Input projection weights at layer X
- b_X: Bias term at layer X
- W_{BB}: Weights of the intra-areal, interlayer backbone (weights connecting layers within the same cortical area)
- W_{FF}: Weights corresponding to the feedforward projection from the lower cortical area to the higher cortical area
- W_{FBa}: Weights for the feedback projection from higher area layer HL5/6 to lower area layer LL2/3
- W_{FBb}: Weights for the feedback projection from higher area layer HL5/6 to lower area layer LL5/6
- $\sigma(\cdot)$: Hyperbolic tangent activation function, $\tanh(\cdot)$
- $(\cdot)_+$: Rectified linear unit activation function, $\mathrm{ReLU}(\cdot)$
- $proj(X \to Y)$: Projection from layer X to layer Y

The network consists of two cortical areas, each with multiple layers: **lower area layers** LL4, LL2/3, LL5/6 and **higher area layers** HL4, HL2/3, HL5/6. Thus, in the notations above, layers $X, Y \in$ {LL4, LL2/3, LL5/6, HL4, HL2/3, HL5/6}. The microcircuit incorporates both feedforward and feedback pathways, as well as intra-areal connections. At each time step t, the microcircuit processes inputs and computes the neural activities for each layer. The sensory input $x^{(t)}$ is projected into the microcircuit through the matrix W_{iLL4} at layer LL4. Information flows from lower layers to higher layers via feedforward projections, particularly from LL2/3 to HL4 through the feedforward weights W_{FF} and a delay of one time step. Higher area layer HL5/6 sends feedback signals to lower area layers LL2/3 and LL5/6 with a delay of two time steps, through feedback weights W_{FBa} and W_{FBb}, respectively. The backbone weights W_{BB} enable communication between layers within the same cortical area. All inputs to a layer X are processed through the layer's input projection matrix W_{iX}.

A.2 Overall Network Flow.

- **Lower layer 4 (LL4)**
  Receives sensory input $x^{(t)}$ and projections from LL5/6 via the intra-areal connection W_{BB}.
  Processes combined inputs through its input projection matrix W_{iLL4}
- **Lower layer 2/3 (LL2/3)**
  Receives delayed feedback from HL5/6 and projections from LL4
  Processes combined inputs through $W_{iLL2/3}$
- **Lower layer 5/6 (LL5/6)**
  Receives delayed feedback from HL5/6 and projections from LL2/3
  Processes combined inputs through $W_{iLL5/6}$
- **Higher layer 4 (HL4)**
  Receives feedforward input from LL2/3 and projections from HL5/6
  Processes combined inputs through W_{iHL4}
- **Higher layer 2/3 (HL2/3)**
  Receives projections from HL4
  Processes inputs through $W_{iHL2/3}$
- **Higher layer 5/6 (HL5/6)**
  Receives projections from HL2/3
  Processes inputs through $W_{iHL5/6}$

A.3 Computations at Various Layers at Time Step t. The computations carried out at each layer at time step t are as follows:

Lower Layer 4 (LL4)

Input processing:

$$\begin{aligned} u_{LL4}^{(t)} &= W_{iLL4} \cdot \left(x^{(t)} \oplus proj(LL5/6 \rightarrow LL4) \right) \\ &= W_{iLL4} \cdot \left(x^{(t)} \oplus W_{BB} \cdot h_{LL5/6}^{(t)} \right) \end{aligned}$$

Activation computation:

$$h_{LL4}^{(t)} = \sigma \left(u_{LL4}^{(t)} + W_{LL4} \cdot h_{LL4}^{(t-1)} + b_{LL4} \right)$$

Lower Layer 2/3 (LL2/3)

Input processing:

$$\begin{aligned} u_{LL2/3}^{(t)} = W_{iLL2/3} \cdot (&proj(HL5/6 \rightarrow LL2/3) \\ &\oplus proj(LL4 \rightarrow LL2/3)) \end{aligned}$$

$$= W_{iLL2/3} \cdot \left(\left(W_{FBa} \cdot h_{HL5/6}^{(t-2)} \right)_+ \oplus W_{BB} \cdot h_{LL4}^{(t)} \right)$$

Activation computation:

$$h_{LL2/3}^{(t)} = \sigma \left(u_{LL2/3}^{(t)} + W_{LL2/3} \cdot h_{LL2/3}^{(t-1)} + b_{LL2/3} \right)$$

Lower Layer 5/6 (LL5/6)

Input processing:

$$\begin{aligned} u_{LL5/6}^{(t)} &= W_{iLL5/6} \cdot (proj(HL5/6 \rightarrow LL5/6) \\ &\quad \oplus proj(LL2/3 \rightarrow LL5/6)) \\ &= W_{iLL5/6} \cdot \left(\left(W_{FBb} \cdot h_{HL5/6}^{(t-2)} \right)_+ \oplus W_{BB} \cdot h_{LL2/3}^{(t)} \right) \end{aligned}$$

Activation computation:

$$h_{LL5/6}^{(t)} = \sigma \left(u_{LL5/6}^{(t)} + W_{LL5/6} \cdot h_{LL5/6}^{(t-1)} + b_{LL5/6} \right)$$

Higher Layer 4 (HL4)

Input processing:

$$\begin{aligned} u_{HL4}^{(t)} &= W_{iHL4} \cdot (proj(LL2/3 \rightarrow HL4) \oplus proj(HL5/6 \rightarrow HL4)) \\ &= W_{iHL4} \cdot \left(\left(W_{FF} \cdot h_{LL2/3}^{(t-1)} \right)_+ \oplus W_{BB} \cdot h_{HL5/6}^{(t)} \right) \end{aligned}$$

Activation computation:

$$h_{HL4}^{(t)} = \sigma \left(u_{HL4}^{(t)} + W_{HL4} \cdot h_{HL4}^{(t-1)} + b_{HL4} \right)$$

Higher Layer 2/3 (HL2/3)

Input processing:

$$\begin{aligned} u_{HL2/3}^{(t)} &= W_{iHL2/3} \cdot proj(HL4 \rightarrow HL2/3) \\ &= W_{iHL2/3} \cdot W_{BB} \cdot h_{HL4}^{(t)} \end{aligned}$$

Activation computation:

$$h_{HL2/3}^{(t)} = \sigma \left(u_{HL2/3}^{(t)} + W_{HL2/3} \cdot h_{HL2/3}^{(t-1)} + b_{HL2/3} \right)$$

Table 1: Expected and Unexpected Sequences for the Memorization Task.

Expected sequence	Temporally violated sequence
x_1-x_2-x_3-x_4	x_1-x_2-x_2-x_4
x_1-x_3-x_2-x_5	x_1-x_3-x_3-x_5
x_2-x_1-x_3-x_6	x_2-x_1-x_1-x_6
x_2-x_3-x_1-x_7	x_2-x_3-x_3-x_7
x_3-x_1-x_2-x_8	x_3-x_1-x_1-x_8
x_3-x_2-x_1-x_9	x_3-x_2-x_2-x_9

Higher Layer 5/6 (HL5/6)

Input processing:

$$u^{(t)}_{HL5/6} = W_{iHL5/6} \cdot proj(HL2/3 \rightarrow HL5/6)$$
$$= W_{iHL5/6} \cdot W_{BB} \cdot h^{(t)}_{HL2/3}$$

Activation computation:

$$h^{(t)}_{HL5/6} = \sigma\left(u^{(t)}_{HL5/6} + W_{HL5/6} \cdot h^{(t-1)}_{HL5/6} + b_{HL5/6}\right)$$

Appendix B: Task Details

B.1 Sequence Memorization. Our exemplary task (see Figure 2A) entails memorizing a set $\mathcal{S}$ with elements that are sequences of length k. In particular, $\mathcal{S}$ is defined as the collection of all $(k-1)!$ possible sequences where the first $k-1$ elements of the sequence are permutations of $\{x_1, x_2, \ldots x_t, \ldots x_{k-1}\}$, and each x_t is a sparse vector such that $x_t \in \{0,1\}^D$ where $x^d_t \sim \text{Bernoulli}(p), d = \{1, \ldots D\}$ and $t = \{1, \ldots k\}$. The kth element is a sequence identifier or label drawn from the same distribution as the other $\{x_i\}_{i=1}^{k-1}$—for example, when $k = 4$, $\mathcal{S} = \{(x_1, x_2, x_3, x_4), (x_1, x_3, x_2, x_5), (x_2, x_1, x_3, x_6) \ldots, (x_3, x_2, x_1, x_9)\}$ and $|\mathcal{S}| = 6$. The inputs are sampled such that $\{x_i\}_{i=1}^{9} \sim \text{Bernoulli}_4(p),\ p \leq \frac{1}{2}$. The training and test sets subsequently consist of sequences $\overset{\text{iid}}{\sim} \mathcal{S}$. We use $n_{train} = n_{test} = 1000$. With $p = 0.25,\ k = 4,$ and $D = 64$, the sets of expected and unexpected sequences take the form in Table 1.

B.2 Lattice Navigation. Our second task is similar to that of Recanatesi et al. (2021), requiring the network to predict the next state x_{t+1} it would traverse to on an $n \times n$ grid, given a sequence of the previous states $(x_0, x_1, \ldots x_t)$ along with the actions $(a_0, a_1, \ldots a_t)$ taken at each of those states (see Figure 2B). Each position on the grid corresponds to a fixed

high-dimensional state $x_t \in \{0,1\}^D$, and the actions $a_t \in \mathbb{1}_i$, $i = \{1,2,3,4\}$ are one-hot vectors corresponding to the four cardinal directions.

In the case of temporal violation at a time step t, we provide the microcircuit with the appropriate action but repeat the state vector x_{t-1} as the state input x_t. For our purposes, we use $D = 64$, $n_{train} = 1600$, $n_{test} = 400$, $n_{epochs} = 50$, $p = 0.75$.

B.3 Binary Addition. Our last task requires the microcircuit to interpret a binary input in D bits and increment it by one at every time step, also in binary (see Figure 2C). Specifically, given the binary representations $(x_z, x_{z+1}, x_{z+2}, \ldots)$ of the numbers $(z, z+1, z+2, \ldots)$ where $x_z \in \{0,1\}^D$ and $z \in \mathbb{R}$, the expected output is $(x_{z+1}, x_{z+2}, x_{z+3}, \ldots)$ that is, the corresponding D bit binary representations of the numbers $(z+1, z+2, z+3)$ at each corresponding time step.

In the case of temporal violation at time step t, which should receive the input x_z, we repeat the input vector x_{z-1} instead of presenting the microcircuit with the appropriate input x_z. For our purposes, we have $D = 16$, $n_{train} = 4000$, $n_{test} = 1000$, and $n_{epochs} = 50$.

For training on all our tasks, we use the Adam optimizer (Kingma & Ba, 2014) with a step size of 0.001, in its standard PyTorch (Paszke et al., 2019) implementation.

Appendix C: Representational Analyses Details

C.1 Dimensionality Gain. DG captures whether representations encode simple or complex task-relevant concepts, and if they do so in an efficient manner, as described in section 2.3 (see Figure 7).

Our measure of linear dimensionality (L_{dim}) is the participation ratio (PR) (Abbott & Dayan, 1999; Litwin-Kumar et al., 2017) where

$$PR = \frac{(\sum_i \lambda_i)^2}{\sum_i (\lambda_i^2)}$$

and λ_i are the eigenvalues of the covariance matrix of the neuronal representations.

Nonlinear measures of dimensionality, however, are far less standardized, and therefore we take the average of four different measures: CorrDim, MLE, DANCo, and $MiND_{ML}$ (Grassberger & Procaccia, 1983; Levina & Bickel, 2004; Lombardi et al., 2011; Ceruti et al., 2012), as our measure for nonlinear dimension (NL_{dim}) from the `scikit-dimension` python package (Bac et al., 2021).

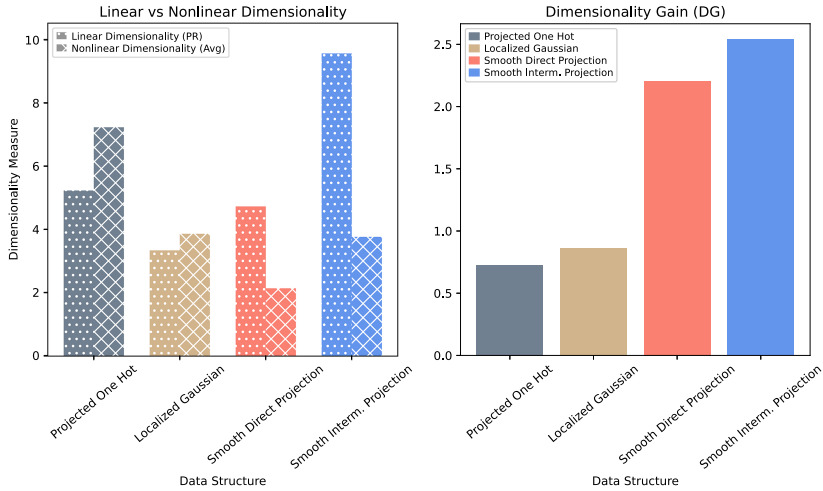

Figure 7: Dimensionality gain for different data sets: Left: Linear and nonlinear dimensionalities of different types of simulated data sets. Right: Resulting DGs of the data.

Consequently,

$$DG = \frac{\mathrm{L}_{dim}}{\mathrm{NL}_{dim}} = \frac{4 \cdot PR}{\mathrm{CorrDim} + \mathrm{MLE} + \mathrm{DANCo} + \mathrm{MiND}_{\mathrm{ML}}}.$$

Examples of how DG scales for some simulated data sets follow these:

- Projected one-hot: Simple one-hot encoding of five discrete states projected via a sparse matrix (30% nonzero elements)
- Localized gaussian: A single latent variable in 50D ambient space with localized gaussian tuning curves and preferred stimulus centers, creating finite-width tuning
- Smooth with direct projection: Three latent variables where each dimension is a sum of sine/cosine functions with random frequency/phase, creating smooth structure. The data are then projected directly into an ambient dimension (50)
- Smooth with intermediate projection: Three latent variables expanded via multiple sine/cosine transformations and then projected to an intermediate dimension (15), before being projected to a final space (50), creating complex structure from simple latents

C.2 Linear Decoding of Task Variables. A greater degree of linear separability among classes of a concept when using fewer dimensions implies

the representations are highly tuned to the particular concept, and therefore more informative of the concept under consideration. We thus compute the principal components (PCs) of the neuronal representations from the different layers and subsequently fit separating hyperplanes between the PCs corresponding to different classes pertinent to the various concepts. The separating hyperplanes are found using a standard linear support vector machine, without any modifications from the `scikit-learn` Python package (Pedregosa et al., 2011).

C.3 Neuronal Population Selectivity. To quantify how tuned the neuronal activity in different cortical layers is to specific concepts, we find the sparseet subset within a given population that allows us to draw separating hyperplanes between different classes relevant to the concept (e.g., different input identities in the case of decoding input states and expected versus unexpected inputs when decoding surprise) as accurately as the full population itself. This allows us to find the neurons that dominate the activity relevant to a particular concept while also accounting for redundancies within the activities of the neurons themselves. Assuming a sparse multinomial logistic regression model trained with cross-entropy, our objective is

$$\min ||\beta||_1 \text{ s.t. } \frac{-1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_{i,k}\log \hat{p}_{i,k} \leq \varepsilon,$$

where ε is the error of the classifier when using all neurons. y is the true label, K is the number of classes relevant to the concept being tested, and N is the total number of samples. As with typical logistic regression, $\hat{p}_{i,k}$ is the predicted probability that the ith sample belongs to class k, given by

$$\hat{p}_{i,k} = \frac{e^{\beta_k \cdot x_i}}{\sum_{j=1}^{K} e^{\beta_j \cdot x_i}}.$$

$\beta \in \mathbb{R}^d$ is the coefficient vector, and $x \in \mathbb{R}^d$ the firing rates of the d neurons in a neuronal population.

Appendix D: Decoding Task-Relevant Variables

D.1 Binary Prediction Task. As before in section 3.1, we notice that input identity is more decodable in layers in the lower area at initialization (see Figure 8, left), and remains so even after training (see Figure 9, left). At the same time, the input ID becomes highly decodable in the higher area too with training (see Figure 9, left). On the other hand, position of an input in the sequence is most decodable in layers higher in the hierarchy, along

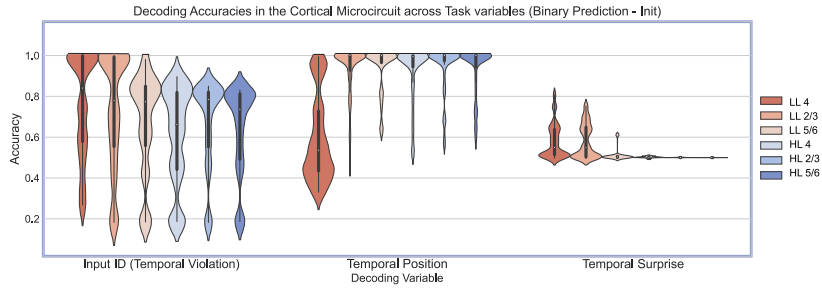

Figure 8: Binary prediction (initialization): Decoding of task-relevant variables.

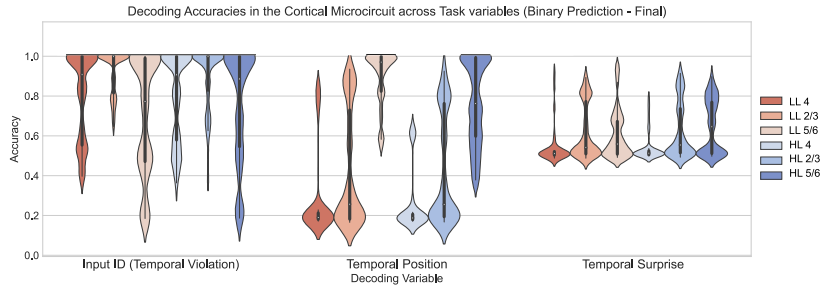

Figure 9: Binary prediction (final): Decoding of task-relevant variables.

with those receiving feedback at initialization (see Figure 8, middle), while post training LL5/6 and HL5/6 seem to do best (see Figure 9, middle). At initialization, all layers seem to only decode surprise at chance[3] (see Figure 8, right), but post training accuracies improve across the microcircuit, with LL5/6 reaching the highest accuracies (Figure 9, right). Overall, the trend that simpler concepts are better encoded in the lower area while more complex ones are better encoded in layers belonging to the higher cortical area holds.

D.2 Lattice Navigation Task. Functional modularization is extremely evident in the case of input decoding (see Figures 10 and 11, left) and position decoding (see Figures 10 and 11, middle). Surprise decoding at initialization (Figure 10, right) was best performed by LL2/3, with slight reduction in decodability across the microcircuit post training (Figure 11, right).

[3] Since the classifier is trained to distinguish between elements that are either categorized as expected or surprise, the probability of randomly decoding correctly is $\frac{1}{2}$.

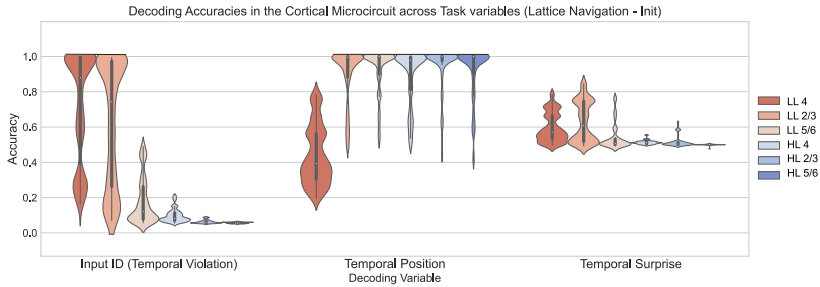

Figure 10: Lattice navigation (initialization): Decoding of task-relevant variables.

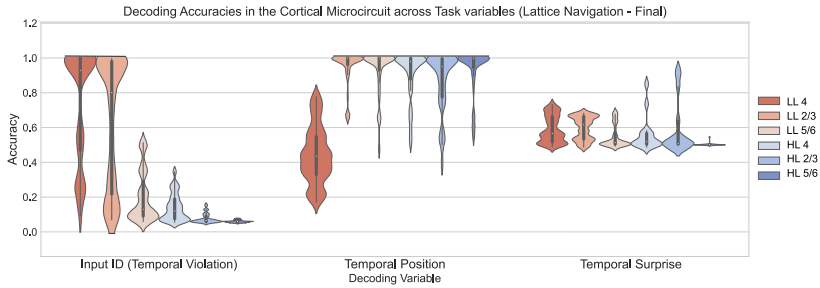

Figure 11: Lattice navigation (final): Decoding of task-relevant variables.

Appendix E: Surprise Decoding: Effects of Feedback and Interareal Time Delay at Initialization

In this section we provide mathematically grounded justifications for the intuitive explanations provided in section 3.2 regarding the decodability of surprise at initialization in LL2/3, LL5/6 for the sequence memorization task. Our analysis also explains why there is no significant separability in the representations of these layers in the absence of any such time delays. In particular, we show more rigorously that

1. Our input stimuli are separable in input space, and this separability is maintained in the case of expected sequence memorization but not in the case of temporal violations.
2. The probability of a random vector being able to act as a linear separator (i.e., decoder) increases with vectors that are farther apart and therefore easily separable.
3. The separability among stimuli in the input space largely carries over to neuronal space (specifically in LL2/3, LL5/6) at initialization.

E.1 Separability of Bernoulli Vectors in Input Space. Given $x, y \sim \mathcal{B}(p)$, where p is the probability that any element x_i or $y_i = 1$, and $x, y \in \{0, 1\}^d$ with d being the length of the vector, the expected distance between the two vectors x, y is

$$\begin{aligned}
\mathbb{E}\left[||x - y||^2\right] &= \mathbb{E}\left[(x - y)^\top (x - y)\right] \\
&= \mathbb{E}\left[||x||^2 + ||y||^2 - 2\langle x, y\rangle\right] \\
&= 2\mathbb{E}\left[||x||^2\right] - 2\mathbb{E}\left[\langle x, y\rangle\right] \\
&= \underbrace{2\sum_{k=0}^{d} k\binom{d}{k} p^k (1-p)^{d-k}}_{\text{Term (1)}} - \underbrace{2\mathbb{E}\left[\sum_{i=1}^{d} x_i \cdot y_i\right]}_{\text{Term (2)}} \\
&= 2dp - 2dp^2 \\
&= 2dp(1-p).
\end{aligned}$$

Term 1 is obtained by first making the substitution $||y||_2^2 = ||x||_2^2$ as their expected values are the same given that they are both drawn from the same distribution, and then summing them. Next, we note that the expected norm squared of the vector x is the exact same quantity as the number of ones in the arbitrary vector x. This is the same quantity as the expected value of successes in a binomial random variable with parameters (d, p), which in turn is equivalent to the sum of d independent Bernoulli trials X_i with the parameter (i.e., mean) p, resulting in the following simplifications:

$$\sum_{k=0}^{d} k\binom{d}{k} p^k (1-p)^{d-k} = \mathbb{E}\left[\sum_{i=1}^{d} X_i\right] = \sum_{i=1}^{d} \mathbb{E}[X_i] = \sum_{i=1}^{d} p = dp.$$

Term 1 consequently can be written as $2dp$. Term 2 is the expected value of the sum of the element-wise multiplication of the vectors x, y. This is the same as counting the number of matching elements between the two vectors, while noting that every element is drawn independently from $\mathcal{B}(p)$. Therefore, we have

$$\mathbb{E}\left[\sum_{i=1}^{d} x_i \cdot y_i\right] = \sum_{i=1}^{d} \mathbb{E}[x_i \cdot y_i] = \sum_{i=1}^{d} 1 \cdot \mathbb{P}(x_i = y_i = 1) = \sum_{i=1}^{d} p^2 = dp^2.$$

which simplifies term 2 to $2dp^2$.

In the sequence memorization task, substituting $x = x_2 + x_3$ and $y = x_3 + x_3$ for the case of the expected sequences as discussed in section 3.2,

we see that the same squared distance $2dp(1-p)$ is maintained by any two arbitrary vectors x_2 and x_3 in input space[4] as $\mathbb{E}[||x-y||^2] = \mathbb{E}[||(x_2+x_3)-(x_3+x_3)||^2] = \mathbb{E}[||x_2-x_3||^2]$, and their eventual representations in neuronal space in LL2/3 and LL5/6, assuming that the same distance is largely preserved in representational space.

Before showing this, we first confirm that an increased distance between vectors implies a higher probability of being able to draw a separating hyperplane between them.

E.2 Probability of Finding a Separating Hyperplane between $\delta-$ Separated Vectors. Without loss of generality, assume vectors $x, y \in \mathbb{R}^d$ are both unit-norm and the distance between them is δ. The probability then, that a random vector v can separate the two, is

$$p := \mathbb{P}(x \cdot v < 0 < y \cdot v) + \mathbb{P}(x \cdot v > 0 > y \cdot v) = 2\mathbb{P}(x \cdot v < 0 < y \cdot v)$$

Assuming uniformly sampled $v \in \mathbb{S}^{d-1}$, v is equal in distribution to the vector $\frac{(Z_1, Z_2, \ldots, Z_d)}{\sqrt{\sum_{i=1}^d Z_i^2}}$ where Z_i are independent standard normal random variables. Additionally, we note that the dot product of a Bernoulli vector $\sim \mathcal{B}(p_b)$ with a random vector v sampled from the unit sphere follows the same distribution as v, as the dot product simply samples a subset of the vector v. Consequently,

$$p = 2\mathbb{P}(X < 0 < Y),$$

where $X := \sum_{i=1}^d x_i Z_i$ and $Y := \sum_{i=1}^d y_i Z_i$. The random variables X and Y are jointly normal with zero means, variances $\frac{1}{p_b}$, and correlation r such that

$$r = \mathbb{E}[XY] = \mathbb{E}[x \cdot y] = \frac{1}{2}\left(||x||^2 + ||y||^2 - ||x-y||^2\right) = 1 - \frac{\delta^2}{2}.$$

The pair (X, Y) equals $\left(X, rX + \sqrt{1-r^2}Z\right)$ in distribution, where Z is a standard normal random variable independent of X. So,

$$\begin{aligned} p &= 2\mathbb{P}\left(X < 0 < rX + \sqrt{1-r^2}Z\right) = 2\mathbb{P}(X < 0, Z > -kX) \\ &= 2\mathbb{P}((X, Z) \in A), \end{aligned}$$

[4] In the case where the the vectors x, y themselves are normalized to be unit length, the squared distance $\mathbb{E}[||\frac{x}{||x||} - \frac{y}{||y||}||^2] = 2(1-p)$ since $\mathbb{E}[||x||^2] = \mathbb{E}[||y||^2] = dp$ acts as the common denominator and cancels out with part of the numerator for $\mathbb{E}[||x-y||^2] = 2dp(1-p)$.

where $k := \frac{r}{\sqrt{1-r^2}}$ and A is the angle between $\{(x, 0) : x \leq 0\}$ and $\{(x, -kx) : x \leq 0\}$.

Since the distribution of the random vector (X, Z) in $\mathbb{R}^2$ is rotation invariant, we conclude that the probability in question is $p = \frac{\theta}{\pi}$, where the measure of the angle A is

$$\theta := \text{arccot} k = \arccos r = \arccos\left(1 - \frac{\delta^2}{2}\right).$$

In particular, it follows that $p = \frac{2\delta}{\pi} + O\left(\delta^3\right) \sim \frac{2\delta}{\pi}$ as $\delta << 1$.

E.3 Johnson-Lindenstrauss Lemma. To make our final point that the distances between vectors in input space are largely preserved in representational space at initialization, we invoke a modified version of the Johnson-Lindenstrauss lemma as stated by theorem 2 (neuronal RP) in (Arriaga & Vempala, 2006).

Lemma. *Let $x, y \in \mathbb{R}^d$ and let x', y' be their projections onto $\mathbb{R}^k$ via a random matrix whose entries are chosen independently from $\mathcal{N}(0, 1)$. Then,*

$$\mathbb{P}[(1-\epsilon)||x-y||^2 \leq ||x'-y'||^2 \leq (1+\epsilon)||x-y||^2] \geq 1 - 2e^{-(\epsilon^2-\epsilon^3)\frac{k}{4}}.$$

The above statement guarantees that a standard normal random gaussian projection can preserve pairwise distances $||x-y||^2$ up to an arbitrarily small precision ϵ with probability $1 - 2e^{-(\epsilon^2-\epsilon^3)\frac{k}{4}}$ as long as we project onto a minimum dimension k. We therefore can use the above result to guarantee that inputs that are appreciably far apart in input space will maintain similar distances (and hence separability) in representational space, given that they are not projected onto a dimension that is significantly smaller than what k should be.

We note, however, that there are two interareal projections that occur to the input before it is processed by either LL2/3 or LL5/6 as part of the feedback signal, since it is first projected forward to HL4 and then projected back from HL5/6. Therefore, allowing for the representation to change by a multiplicative factor of ϵ at each step, we have

$$\mathbb{P}[(1-\epsilon)^2\delta^2 \leq \delta_k^2 \leq (1+\epsilon)^2\delta^2] \geq \left(1 - 2e^{-(\epsilon^2-\epsilon^3)\frac{k}{4}}\right)^2$$

where $\delta = ||x-y||$ is the original distance between the two vectors x and y and δ_k is the distance between the projected vectors x' and y' in $k-$dimensional space. If we let $\bar{p} = (1 - 2e^{-(\epsilon^2-\epsilon^3)\frac{k}{4}})^2$ we find that the

minimum dimension k that we need to project to to maintain pairwise distances is

$$k \geq \frac{4}{\epsilon^2(1-\epsilon)} \ln\left(\frac{2}{1-\sqrt{\bar{p}}}\right). \tag{E.1}$$

Substituting $\bar{p} = \frac{2\delta}{\pi}$ from the result in section E.2, and $\delta = 2(1-p)$ from the calculation shown in note 4 into equation E.1,

$$k \geq \frac{4}{\epsilon^2(1-\epsilon)} \ln\left(\frac{2\sqrt{\pi}}{\sqrt{\pi} - 2\sqrt{(1-p)}}\right) \approx \frac{17.9}{\epsilon^2(1-\epsilon)}$$

for $p = 0.25$ as mentioned in appendix B.

While these results hold precisely for weights that are sampled from $\mathcal{N}(0,1)$, the nature of the results extends to other gaussian distributions too, with changes in the constants. All weights at initialization in the microcircuit are sampled from the gaussian distribution $\mathcal{N}(0, \sigma_n^2)$, where n is the number of neurons and the variance $\sigma_n^2 \propto n^{-1}$. These include recurrent weights within a population in the same layer (e.g., W_{LL4}, $W_{LL5/6}$, $W_{HL2/3}$), weights of the interareal connections (i.e, W_{FF}, W_{FBa}, W_{FBb}) and intra-areal, interlayer lateral connections (i.e., W_{BB}).

Furthermore, while we impose sparsity in the connectivity across and within layers by multiplying all nonrecurrent unit weights with a binary mask sampled from a Bernoulli distribution $\mathcal{B}(p)$, where p is a predetermined probability of connection, the sparsified weights are simply a subset sampled from the original gaussian distribution and still follow a gaussian distribution, $\mathcal{N}\left(0, \frac{\sigma_n^2}{p}\right)$.

We note that our subsequent analyses rely on the assumption that the pairwise distances between inputs are approximately preserved when processed through the microcircuit up to LL2/3. This requires analyzing two successive projections: first from the input space (16 dimensions) to LL4 (32 neurons) and then from LL4 to LL2/3 (80 neurons). This can be justified using the neuronal RP theorem. For each projection of vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ to $\mathbf{x}', \mathbf{y}' \in \mathbb{R}^k$ via a random matrix with entries from $\mathcal{N}(0,1)$, the probability of distance preservation is bounded by

$$P[(1-\epsilon)||\mathbf{x}-\mathbf{y}||^2 \leq ||\mathbf{x}'-\mathbf{y}'||^2 \leq (1+\epsilon)||\mathbf{x}-\mathbf{y}||^2] \geq 1 - 2e^{-(\epsilon^2-\epsilon^3)k/4}.$$

Given that the expected distance between our Bernoulli inputs is $\delta = 2p(1-p)$ and the probability of finding a separating hyperplane is approximately $2\delta/\pi$, the minimum dimension k required for preserving distances with distortion at most ϵ is:

$$k \geq \frac{4}{\epsilon^2(1-\epsilon)} \ln\left(\frac{2}{1-\sqrt{4p(1-p)/\pi}}\right). \tag{E.2}$$

For the first projection from input to LL4, $k = 32$ satisfies this bound with $\epsilon \geq 0.452$. For the subsequent projection to LL2/3 with $k = 80$, we get $\epsilon \geq 0.386$. The composite effect of these two projections means that distances are preserved within a multiplicative factor of approximately $(1 \pm 0.452) \cdot (1 \pm 0.386)$. While this indicates nonnegligible distortion through the circuit, the separability of inputs is still maintained with high probability due to the expansion of dimensionality at each step ($16 \rightarrow 32 \rightarrow 80$), allowing us to proceed with our subsequent analyses of surprise encoding in these layers.

In totality, the results and analyses in this section provide a mathematically grounded explanation as to why it is possible for the neuronal representations in LL2/3 and LL5/6 to distinguish between expected and surprising elements of a sequence in a simple sequential memorization task at the initialization. We note that these results require the tasks to be both "simple" and sequential in nature, as if the inputs are not significantly apart in the input/representation space before reaching the lower cortical area, or if there is no explicit temporal structure that the microcircuit can exploit, this argument fails.

Appendix F: Reconstruction Performance across Architectures and Tasks in the Presence of Expectation Violations

We note that the red bars in Figure 12 (test accuracies for producing the correct final output given temporal violations in the input sequence) for architectures that receive feedback (columns a, c, d, e) are appreciably higher than that of the no-feedback architecture (column b) for the binary addition task (row 2) and slightly higher for the sequence memorization task (row 1). Following is the p-value annotation legend:

- ns: 5.00e-02 $< p \leq$ 1.00e+00
- * 1.00e-02 $< p \leq$ 5.00e-02
- ** 1.00e-03 $< p \leq$ 1.00e-02
- *** 1.00e-04 $< p \leq$ 1.00e-03
- **** $p \leq$ 1.00e-04

Appendix G: Predictive Coding-Based Training Algorithm

Algorithm 1: Predictive-Coding Based Training.

procedure TWO-PHASE-TRAINING(Input data, corticalRNN, max_epochs)
  Initialize corticalRNN model parameters randomly
  $epoch \leftarrow 1$
  while $epoch \leq max\_epochs$ **do**
    if $epoch$ is odd **then**
      Phase 1 (Reconstruction-based loss):
      for all mini-batches in training data **do**
        $\mathcal{L}_{recon} \leftarrow \Sigma_t ||\hat{x}_{t+1} - x_{t+1}||_2^2$
        Update all learnable parameters with backprop
      end for
    else
      Phase 2 (Predictive loss):
      for all mini-batches in training data **do**
        $\mathcal{L}_{PE} \leftarrow \frac{1}{2}\Sigma_t \left(||W_{FBa} \cdot h_{HL5/6,t} - h_{LL2/3,t}||_2^2 + ||W_{FBa} \cdot h_{HL5/6,t} - h_{LL5/6,t}||_2^2 \right)$
        Update parameters of $HL2/3, HL4, HL5/6$ and W_{FB_a}, W_{FB_b} with backprop
      end for
    end if
    $epoch \leftarrow epoch + 1$
  end while
end procedure

Appendix H: Implicit Bayesian Nature of Predictive-Coding-Based Loss

While our predictive coding loss is motivated from a mechanistic viewpoint of wanting to minimize the discrepancies between the predictions in the higher area and the activities in the lower area, it can also be derived by maximizing the likelihood of the activity in the lower area (LL2/3, *LL*5/6) with respect to the activity in the higher area (HL5/6) and the feedback weights (W_{FBa}, W_{FBb}) (Rao & Ballard (1999)).

Note that during the second phase of our training strategy, we assume the activities of the two lower layers are fixed and thus conditionally independent given the higher layer activity $h_{HL5/6}$ and the feedback weights W_{FBa}, W_{FBb}. This is because only the higher cortical layers and feedback connections are updated, while the lower layer activities remain unchanged. Therefore, the feedback weights learned corresponding to each of the two lower layers are considered independent of each other during this phase. Hence, without loss of generality, we derive the loss corresponding to LL2/3 and W_{FBa}, and replicate the same derivation for LL5/6 and W_{FBb}.

Let $h_{LL2/3}$ and $h_{HL5/6}$ denote the activities of LL2/3 and HL5/6, respectively. Let W_{FBa} be the feedback connection that projects from HL5/6 to LL2/3. Assuming that $h_{LL2/3} \in \mathbb{R}^n$ (where n is the size of the neuronal

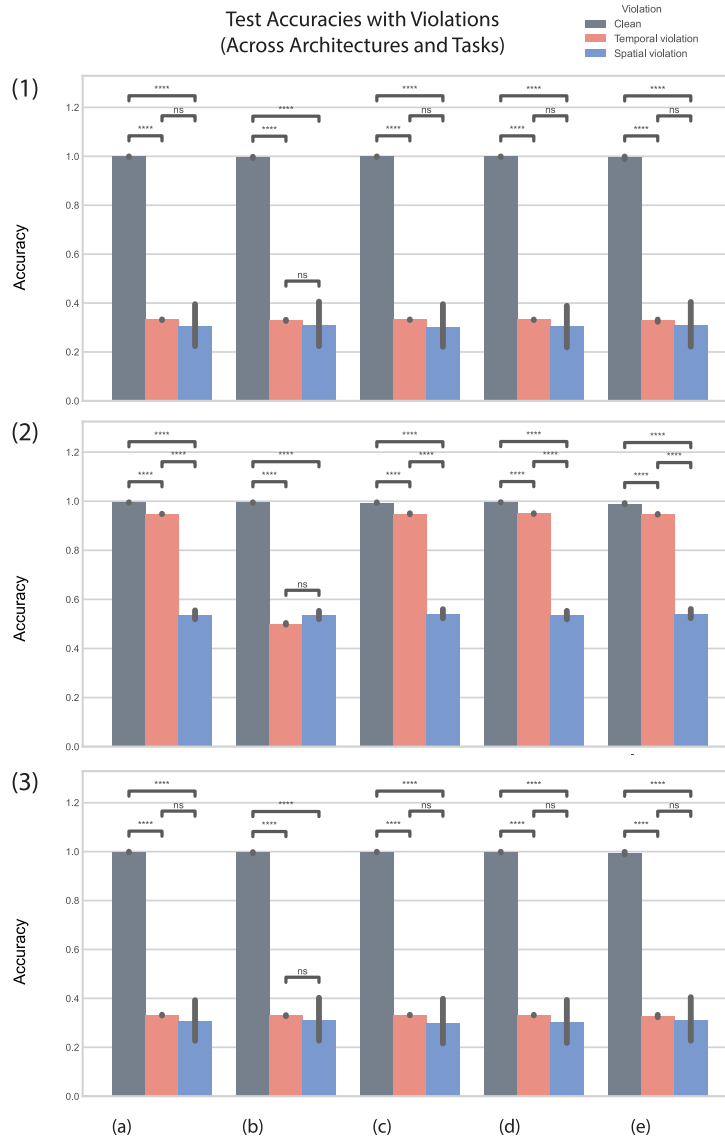

Figure 12: Performance of architectures across all tasks and violations. Rows denote different tasks, and columns denote different architectural modifications of the cortical microcircuit. Task 1 = Sequence memorization. Task 2 = binary addition. Task 3 = lattice navigation. Architecture a = cortical RNN. Architecture b = no feedback. Architecture c = bi-directional feedback. Architecture d = unidirectional feedback. Architecture e = population controlled.

population in LL2/3) is a multivariate gaussian centered at $W_{FBa} \cdot h_{HL5/6}$ with variance $\sigma^2 I_n$, the likelihood can be written as

$$p\left(h_{LL2/3}|h_{HL5/6}, W_{FBa}\right) = \left(2\pi\sigma^2\right)^{-\frac{n}{2}} e^{-\frac{\left(h_{LL2/3}-W_{FBa}\cdot h_{HL5/6}\right)^{\top}\left(h_{LL2/3}-W_{FBa}\cdot h_{HL5/6}\right)}{2\sigma^2}}.$$

The negative log likelihood is consequently given by

$$-\log\left(p\left(h_{LL2/3}|h_{HL5/6}, W_{FBa}\right)\right) = \frac{n}{2}\log\left(2\pi\sigma^2\right) + \frac{1}{2\sigma^2}||h_{LL2/3} - W_{FBa}\cdot h_{HL5/6}||^2.$$

Since $\frac{n}{2}\log\left(2\pi\sigma^2\right)$ remains unaffected by any changes in W_{FBa} and $h_{HL5/6}$, we have

$$-\log\left(p\left(y_{LL2/3}|h_{HL5/6}, W_{FBa}\right)\right) \propto \frac{1}{2\sigma^2}||h_{LL2/3} - W_{FBa}\cdot h_{HL5/6}||^2.$$

Therefore, the only term we would need to minimize for the likelihood maximization of $h_{LL2/3}$ (or, alternatively, its negative log-likelihood minimization) is $||W_{FBa}\cdot h_{HL5/6} - h_{LL2/3}||^2$.

Likewise, the term needed to maximize the likelihood of $h_{LL5/6}$ is $||W_{FBb}\cdot h_{HL5/6} - h_{LL5/6}||^2$. Taking the average of the two, we arrive at our final top-down prediction-error-based loss

$$\mathcal{L}_{PC} = \frac{1}{2}||W_{FB_a}\cdot h_{HL5/6,t} - h_{LL2/3,t}||^2 + \frac{1}{2}||W_{FB_b}\cdot h_{HL5/6,t} - h_{LL5/6,t}||^2.$$

Appendix I: Dimensionality Gain with Predictive-Coding (PC) Loss Across Binary Prediction and Lattice Navigation Tasks

We note that in the case of all our tasks, for architectures that do not receive any feedback, the trend in the DG of LL5/6 changes from that compared to when the architectures receive feedback. Additionally, we note that the two-phase training causes oscillations that are particularly exacerbated in the LL5/6 DG in the absence of feedback.

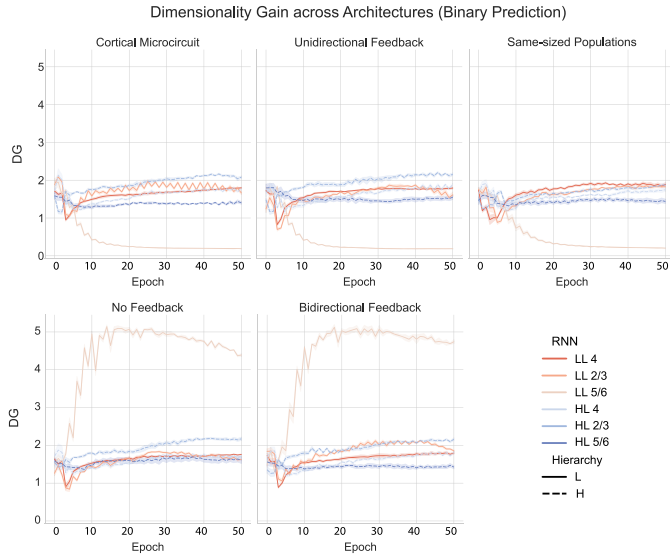

Figure 13: Binary prediction: Dimensionality gain across architectures.

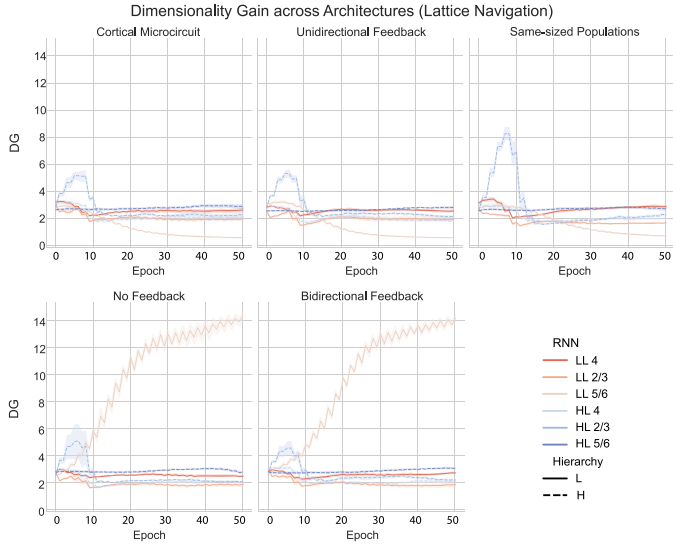

Figure 14: Lattice navigation: Dimensionality gain across architectures.

Appendix J: Three-Area Model Experiments

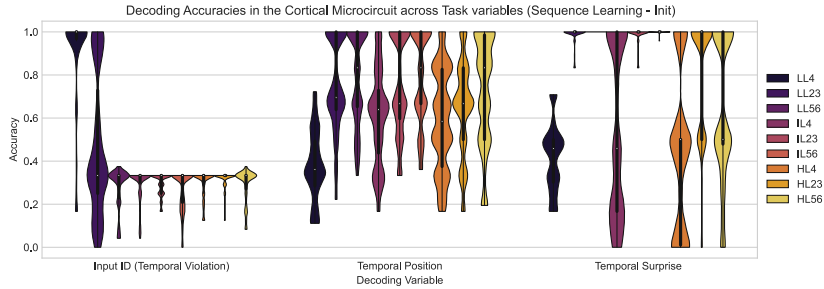

Figure 15: Decoding of task-relevant variables in the sequential memorization task at the initialization.

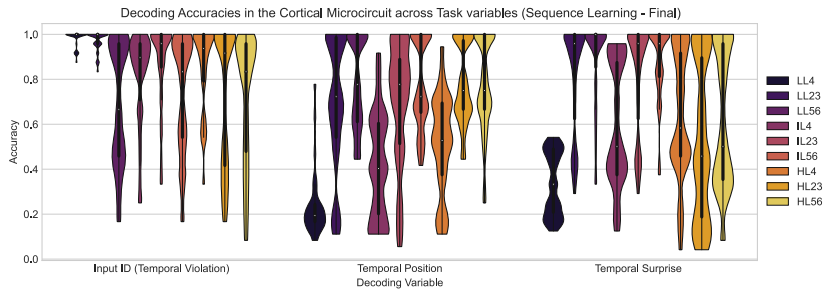

Figure 16: Decoding of task-relevant variables in the sequential memorization task post training.

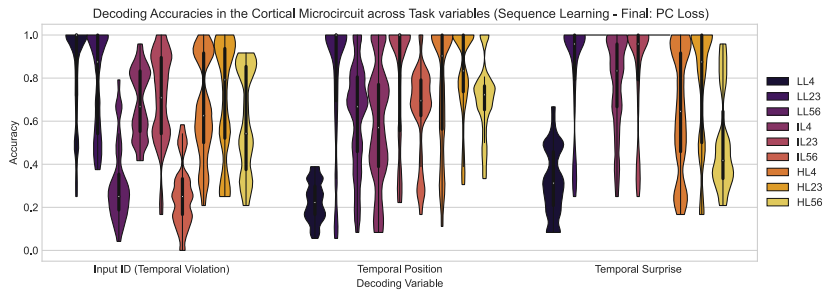

Figure 17: Decoding of task-relevant variables in the sequential memorization task *post training, trained with the predictive-coding-based loss*.

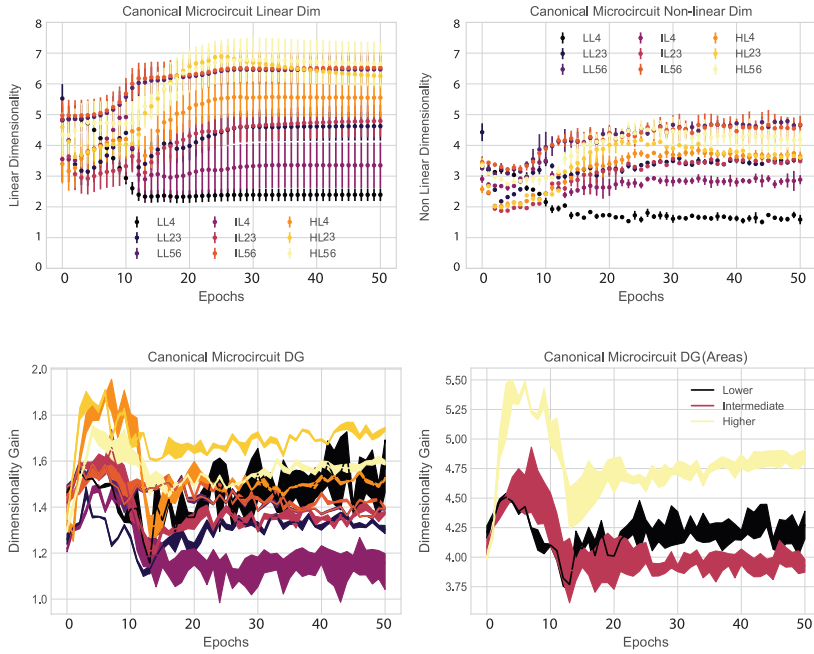

Figure 18: Sequential memorization (trained only with the reconstruction loss): dimensionality gain plots. Top left: Linear dimensionality of representations. Top right: Nonlinear dimensionality of representations. Bottom left: DGs of individual layers. Bottom right: DGs averaged across areas.

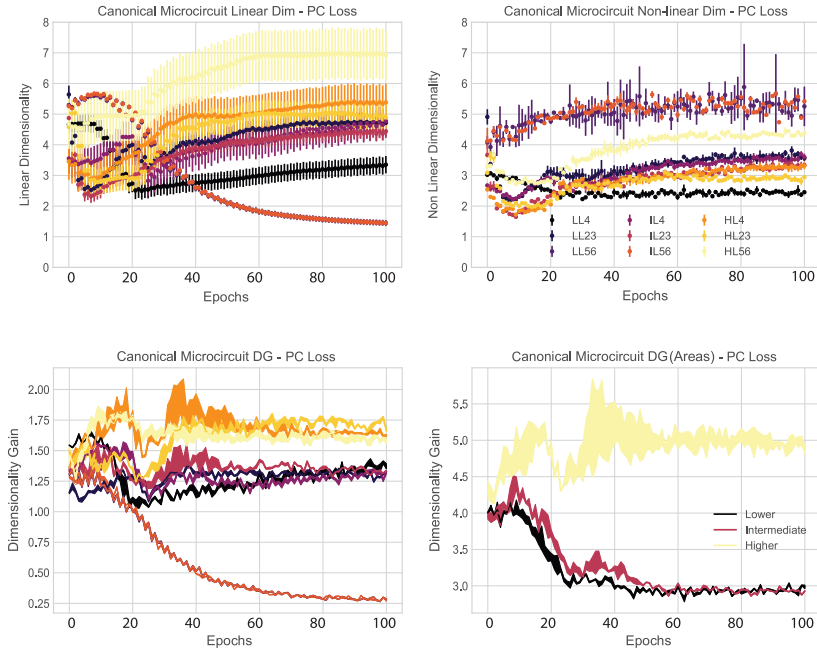

Figure 19: Sequential memorization (trained with the predictive-coding-based loss): dimensionality gain plots. Top left: Linear dimensionality of representations. Top right: Nonlinear dimensionality of representations. Bottom left: DGs of individual layers, Bottom right: DGs averaged across areas.

Appendix K: Neuronal Selectivity for Surprise in LL5/6 with Predictive-Coding (PC) Loss

We find that when averaged over multiple runs, the fraction of neurons needed to decode expected versus surprise is consistently lower post training with the predictive-coding-based (PC) loss compared to initialization, as well as when not using the PC loss (see Figure 20). Moreover, the drop in the median fraction of neurons used is higher when the microcircuit is trained with the PC loss (right) than without (left).

The box and whisker plots follow standard convention, where the lower edge of the box extends to the first bottom ($Q1$) and the upper edge extends to the top of the third quartiles ($Q3$). The lower whisker extends to $Q1 - 1.5 \cdot (Q3 - Q1)$ and the upper whisker extends to $Q3 + 1.5 \cdot (Q3 - Q1)$. The line inside the box plots represents the median ($Q2$).

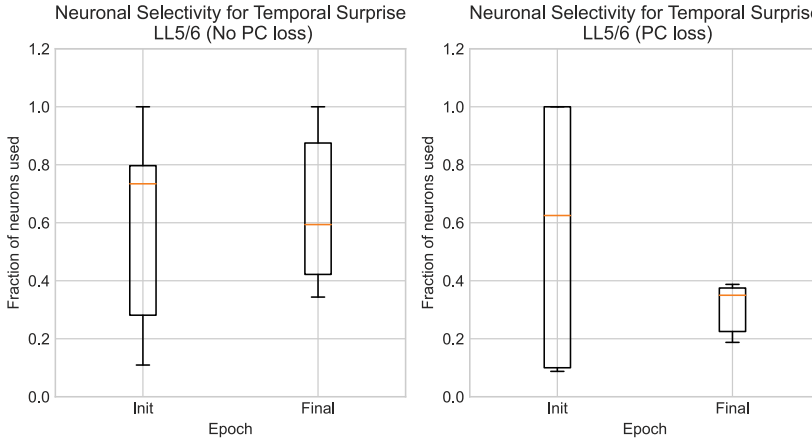

Figure 20: Effect of PC loss on neuronal selectivity for surprise in LL5/6. Left: Fraction of neurons needed to decode surprise without PC loss at initialization and post training. Right: Fraction of neurons needed to decode surprise with PC loss at initialization and post training.

Acknowledgments

This work was supported by the National Eye Institute of the National Institutes of Health under award R00 EY030840 and an Alfred P. Sloan Research Fellowship in Neuroscience to H.C. The content is solely our responsibility and does not necessarily represent the official views of the National Institutes of Health.

References

Abbott, L. F., & Dayan, P. (1999). The effect of correlated variability on the accuracy of a population code. *Neural Computation*, *11*(1), 91–101. 10.1162/089976699300016827

Arriaga, R. I., & Vempala, S. (2006). An algorithmic theory of learning: Robust concepts and random projection. *Machine Learning*, *63*, 161–182. 10.1007/s10994-006-6265-7

Audette, N. J., & Schneider, D. M. (2023). Stimulus-specific prediction error neurons in mouse auditory cortex. *Journal of Neuroscience*, *43*(43), 7119–7129. 10.1523/JNEUROSCI.0512-23.2023

Audette, N. J., Zhou, W., La Chioma, A., & Schneider, D. M. (2022). Precise movement-based predictions in the mouse auditory cortex. *Current Biology*, *32*(22), 4925–4940. 10.1016/j.cub.2022.09.064

Bac, J., Mirkes, E. M., Gorban, A. N., Tyukin, I., & Zinovyev, A. (2021). Scikit-dimension: A Python package for intrinsic dimension estimation. *Entropy*, *23*(10), 1368. 10.3390/e23101368

Barry, M. L., & Gerstner, W. (2024). Fast adaptation to rule switching using neuronal surprise. *PLOS Computational Biology*, *20*(2), e1011839. 10.1371/journal.pcbi.1011839

Bastos, A. M., Litvak, V., Moran, R., Bosman, C. A., Fries, P., & Friston, K. J. (2015). A DCM study of spectral asymmetries in feedforward and feedback connections between visual areas V1 and V4 in the monkey. *NeuroImage*, *108*, 460–475. 10.1016/j.neuroimage.2014.12.081

Bastos, A. M., Usrey, W. M., Adams, R. A., Mangun, G. R., Fries, P., & Friston, K. J. (2012). Canonical microcircuits for predictive coding. *Neuron*, *76*(4), 695–711. 10.1016/j.neuron.2012.10.038

Bowman, H., Collins, D., Nayak, A., & Cruse, D. (2023). Is predictive coding falsifiable? *Neuroscience and Biobehavioral Reviews*, 105404.

Cadena, S. A., Denfield, G. H., Walker, E. Y., Gatys, L. A., Tolias, A. S., Bethge, M., & Ecker, A. S. (2019). Deep convolutional models improve predictions of macaque V1 responses to natural images. *PLOS Computational Biology*, *15*(4), e1006897. 10.1371/journal.pcbi.1006897

Cain, N., Iyer, R., Koch, C., & Mihalas, S. (2016). The computational properties of a simplified cortical column model. *PLOS Computational Biology*, *12*(9), e1005045. 10.1371/journal.pcbi.1005045

Ceruti, C., Bassis, S., Rozza, A., Lombardi, G., Casiraghi, E., & Campadelli, P. (2012). *Danco: Dimensionality from angle and norm concentration*. arXiv:1206.3881.

Dasgupta, S., & Gupta, A. (2003). An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures and Algorithms*, *22*(1), 60–65. 10.1002/rsa.10073

Dayan, P., Hinton, G. E., Neal, R. M., & Zemel, R. S. (1995). The Helmholtz machine. *Neural Computation*, *7*(5), 889–904. 10.1162/neco.1995.7.5.889

De Kock, C., Bruno, R. M., Spors, H., & Sakmann, B. (2007). Layer-and cell-type-specific suprathreshold stimulus representation in rat primary somatosensory cortex. *Journal of Physiology*, *581*(1), 139–154. 10.1113/jphysiol.2006.124321

DiCarlo, J. J., & Cox, D. D. (2007). Untangling invariant object recognition. *Trends in Cognitive Sciences*, *11*(8), 333–341. 10.1016/j.tics.2007.06.010

Douglas, R. J., & Martin, K. (1991). A functional microcircuit for cat visual cortex. *Journal of Physiology*, *440*(1), 735–769. 10.1113/jphysiol.1991.sp018733

Douglas, R. J., & Martin, K. A. (2004). Neuronal circuits of the neocortex. *Annual Review of Neuroscience*, *27*(1), 419–451. 10.1146/annurev.neuro.27.070203.144152

Douglas, R. J., Martin, K. A., & Whitteridge, D. (1989). A canonical microcircuit for neocortex. *Neural Computation*, *1*(4), 480–488. 10.1162/neco.1989.1.4.480

Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, *7*, 195–225. 10.1023/A:1022699029236

Felleman, D. J., & Van Essen, D. C. (1991). Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex*, *1*(1), 1–47. 10.1093/cercor/1.1.1

Friston, K. (2005). A theory of cortical responses. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *360*(1456), 815–836. 10.1098/rstb.2005.1622

Friston, K. (2008). Hierarchical models in the brain. *PLOS Computational Biology*, *4*(11), e1000211.

Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, *11*(2), 127–138. 10.1038/nrn2787

Furutachi, S., Franklin, A. D., Aldea, A. M., Mrsic-Flogel, T., & Hofer, S. B. (2024). Cooperative thalamocortical circuit mechanism for sensory prediction errors. *Nature*, *633*, 398–406. 10.1038/s41586-024-07851-w

Garrett, M., Manavi, S., Roll, K., Ollerenshaw, D. R., Groblewski, P. A., Ponvert, N. D., . . . Olsen, S. (2020). Experience shapes activity dynamics and stimulus coding of VIP inhibitory cells. *eLife*, *9*, e50340.

Gillon, C. J., Pina, J. E., Lecoq, J. A., Ahmed, R., Billeh, Y. N., Caldejon, S., . . . Gold, J. (2024). Responses to pattern-violating visual stimuli evolve differently over days in somata and distal apical dendrites. *Journal of Neuroscience*, *44*(5). 10.1523/JNEUROSCI.1009-23.2023

Golkar, S., Tesileanu, T., Bahroun, Y., Sengupta, A., & Chklovskii, D. (2022). Constrained predictive coding as a biologically plausible model of the cortical hierarchy. In *Advances in neural information processing systems*, *35* (pp. 14155–14169). Curran.

Grassberger, P., & Procaccia, I. (1983). Measuring the strangeness of strange attractors. *Physica D: Nonlinear Phenomena*, *9*(1–2), 189–208. 10.1016/0167-2789(83)90298-1

Gregory, R. L. (1968). Perceptual illusions and brain models. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, *171*(1024), 279–296.

Guerguiev, J., Lillicrap, T. P., & Richards, B. A. (2017). Towards deep learning with segregated dendrites. *eLife*, *6*, e22901.

Harris, J. A., Mihalas, S., Hirokawa, K. E., Whitesell, J. D., Choi, H., Bernard, A., . . . Zeng, H. (2019). Hierarchical organization of cortical and thalamic connectivity. *Nature*, *575*(7781), 195–202. 10.1038/s41586-019-1716-z

Hawkins, J. (2021). *A thousand brains: A new theory of intelligence*. Basic Books.

Helmholtz, H. v. (1860). *Handbuch der physiologischen optik* [English translation]. New York.

Hertäg, L., & Clopath, C. (2022). Prediction-error neurons in circuits with multiple neuron types: Formation, refinement, and functional implications. *Proceedings of the National Academy of Sciences*, *119*(13), e2115699119.

Homann, J., Koay, S. A., Chen, K. S., Tank, D. W., & Berry, M. J. (2022). Novel stimuli evoke excess activity in the mouse primary visual cortex. *Proceedings of the National Academy of Sciences*, *119*(5), e2108882119. 10.1073/pnas.2108882119

Hromádka, T., DeWeese, M. R., & Zador, A. M. (2008). Sparse representation of sounds in the unanesthetized auditory cortex. *PLOS Biology*, *6*(1), e16.

Jiang, L. P., & Rao, R. P. (2024). Dynamic predictive coding: A model of hierarchical sequence learning and prediction in the neocortex. *PLOS Computational Biology*, *20*(2), e1011801.

Keller, G. B., & Mrsic-Flogel, T. D. (2018). Predictive processing: A canonical cortical computation. *Neuron*, *100*(2), 424–435. 10.1016/j.neuron.2018.10.003

Kermani Nejad, K., Anastasiades, P., Hertäg, L., & Costa, R. P. (2024). *Self- supervised predictive learning accounts for layer-specific cortical observations*. bioRxiv:2024–04.

Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization.* arXiv:1412.6980.

Kogo, N., & Trengove, C. (2015). Is predictive coding theory articulated enough to be testable? *Frontiers in Computational Neuroscience*, *111*.

Lamme, V. A., & Roelfsema, P. R. (2000). The distinct modes of vision offered by feedforward and recurrent processing. *Trends in Neurosciences*, *23*(11), 571–579. 10.1016/S0166-2236(00)01657-X

Larkum, M. (2013). A cellular mechanism for cortical associations: An organizing principle for the cerebral cortex. *Trends in Neurosciences*, *36*(3), 141–151. 10.1016/j.tins.2012.11.006

Levina, E., & Bickel, P. (2004). Maximum likelihood estimation of intrinsic dimension. In L. Saul, Y. Weiss, & L. Bottou (Eds.), *Advances in neural information processing systems*, *17*. MIT Press.

Lindsey, J., Ocko, S. A., Ganguli, S., & Deny, S. (2019). A unified theory of early visual representations from retina to cortex through anatomically constrained deep CNNs. In *Proceedings of the International Conference on Learning Representations*.

Litwin-Kumar, A., Harris, K., Axel, R., Sompolinsky, H., & Abbott, L. (2017). Optimal degrees of synaptic connectivity. *Neuron*, *93*(5), 1153–1164. 10.1016/j.neuron.2017.01.030

Lombardi, G., Rozza, A., Ceruti, C., Casiraghi, E., & Campadelli, P. (2011). Minimum neighbor distance estimators of intrinsic dimension. In *Proceedings of the Machine Learning and Knowledge Discovery in Databases: European Conference* (pp. 374–389).

Maass, W., Natschläger, T., & Markram, H. (2004). Computational models for generic cortical microcircuits. *Computational Neuroscience: A Comprehensive Approach*, *18*, 575–605.

Maier, A., Adams, G. K., Aura, C., & Leopold, D. A. (2010). Distinct superficial and deep laminar domains of activity in the visual cortex during rest and stimulation. *Frontiers in Systems Neuroscience*, *4*, 31.

McIntosh, L., Maheswaranathan, N., Nayebi, A., Ganguli, S., & Baccus, S. (2016). Deep learning models of the retinal response to natural scenes. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, *29*. Curran.

Merel, J., Aldarondo, D., Marshall, J., Tassa, Y., Wayne, G., & Ölveczky, B. (2019). *Deep neuroethology of a virtual rodent*. arXiv:1911.09451.

Mikulasch, F. A., Rudelt, L., Wibral, M., & Priesemann, V. (2023). Where is the error? Hierarchical predictive coding through dendritic error computation. *Trends in Neurosciences*, *46*(1), 45–59. 10.1016/j.tins.2022.09.007

Mountcastle, V. B. (1997). The columnar organization of the neocortex. *Brain*, *120*(4), 701–722. 10.1093/brain/120.4.701

Murray, J. D., Bernacchia, A., Freedman, D. J., Romo, R., Wallis, J. D., Cai, X., . . . Wang, X.-J. (2014). A hierarchy of intrinsic timescales across primate cortex. *Nature Neuroscience*, *17*(12), 1661–1663. 10.1038/nn.3862

Nessler, B., Pfeiffer, M., Buesing, L., & Maass, W. (2013). Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity. *PLOS Computational Biology*, *9*(4), e1003037. 10.1371/journal.pcbi.1003037

O'Reilly, R. C., Russin, J. L., Zolfaghar, M., & Rohrlich, J. (2021). Deep predictive learning in neocortex and pulvinar. *Journal of Cognitive Neuroscience*, *33*(6), 1158–1196.

O'Reilly, R. C., Wyatte, D., & Rohrlich, J. (2014). *Learning through time in the thalamocortical loops*. arXiv:1407.3432.

Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, *37*(23), 3311–3325. 10.1016/S0042-6989(97)00169-7

Olshausen, B. A., & Field, D. J. (2004). Sparse coding of sensory inputs. *Current Opinion in Neurobiology*, *14*(4), 481–487. 10.1016/j.conb.2004.07.007

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., . . . Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems*, *32*. Curran.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Cournapeau, D. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Perich, M. G., Arlt, C., Soares, S., Young, M. E., Mosher, C. P., Minxha, J., . . . Rajan, K. (2020). *Inferring brain-wide interactions using data-constrained recurrent neural network models*. bioRxiv:2020–12.

Petreanu, L., Mao, T., Sternson, S. M., & Svoboda, K. (2009). The subcellular organization of neocortical excitatory connections. *Nature*, *457*(7233), 1142–1145. 10.1038/nature07709

Rao, R. P., & Ballard, D. H. (1999). Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, *2*(1), 79–87. 10.1038/4580

Rao, R., & Sejnowski, T. J. (1999). Predictive sequence learning in recurrent neocortical circuits. In S. Solla, T. Leen, & K. Müller (Eds.), *Advances in neural information processing systems*, *12*. MIT Press.

Recanatesi, S., Farrell, M., Lajoie, G., Deneve, S., Rigotti, M., & Shea-Brown, E. (2021). Predictive learning as a network mechanism for extracting low-dimensional latent space representations. *Nature Communications*, *12*(1), 1417. 10.1038/s41467-021-21696-1

Richards, B. A., & Lillicrap, T. P. (2019). Dendritic solutions to the credit assignment problem. *Current Opinion in Neurobiology*, *54*, 28–36. 10.1016/j.conb.2018.08.003

Rigotti, M., Barak, O., Warden, M. R., Wang, X.-J., Daw, N. D., Miller, E. K., & Fusi, S. (2013). The importance of mixed selectivity in complex cognitive tasks. *Nature*, *497*(7451), 585–590. 10.1038/nature12160

Rust, N. C., & DiCarlo, J. J. (2010). Selectivity and tolerance ("invariance") both increase as visual information propagates from cortical area V4 to IT. *Journal of Neuroscience*, *30*(39), 12978–12995. 10.1523/JNEUROSCI.0179-10.2010

Sacramento, J., Ponte Costa, R., Bengio, Y., & Senn, W. (2018). Dendritic cortical microcircuits approximate the backpropagation algorithm. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems*, *31*. Curran.

Sakata, S., & Harris, K. D. (2009). Laminar structure of spontaneous and sensory-evoked population activity in auditory cortex. *Neuron*, *64*(3), 404–418. 10.1016/j.neuron.2009.09.020

Shi, J., Tripp, B., Shea-Brown, E., Mihalas, S., & Buice, M. (2022). Mousenet: A biologically constrained convolutional neural network model for the mouse visual cortex. *PLOS Computational Biology*, *18*(9), e1010427. 10.1371/journal.pcbi.1010427

Shipp, S. (2016). Neural elements for predictive coding. *Frontiers in Psychology*, *7*, 1792. 10.3389/fpsyg.2016.01792

Sohn, H., & Narain, D. (2021). Neural implementations of Bayesian inference. *Current Opinion in Neurobiology*, *70*, 121–129. 10.1016/j.conb.2021.09.008

Spratling, M. W. (2017). A review of predictive coding algorithms. *Brain and Cognition*, *112*, 92–97. 10.1016/j.bandc.2015.11.003

Takahashi, N., Ebner, C., Sigl-Glöckner, J., Moberg, S., Nierwetberg, S., & Larkum, M. E. (2020). Active dendritic currents gate descending cortical outputs in perception. *Nature Neuroscience*, *23*(10), 1277–1285. 10.1038/s41593-020-0677-8

Usrey, W. M., & Fitzpatrick, D. (1996). Specificity in the axonal connections of layer VI neurons in tree shrew striate cortex: Evidence for distinct granular and supragranular systems. *Journal of Neuroscience*, *16*(3), 1203–1218. 10.1523/JNEUROSCI.16-03-01203.1996

Westerberg, J. A., Xiong, Y. S., Nejat, H., Sennesh, E., Durand, S., Cabasco, H., . . . Bastos, A. M. (2024). *Stimulus history, not expectation, drives sensory prediction errors in mammalian cortex*. bioRxiv.

Wyrick, D. G., Cain, N., Larsen, R. S., Lecoq, J., Valley, M., Ahmed, R., . . . Mazzucato (2023). *Differential encoding of temporal context and expectation under representational drift across hierarchically connected areas*. biorxiv.org/content/10.1101/2023.06.02.543483v2

Yamins, D. L., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., & DiCarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, *111*(23), 8619–8624. 10.1073/pnas.1403112111

Received November 10, 2024; accepted May 6, 2025.