Predictive Sequence Learning in the Hippocampal Formation

Yusi Chen,[1,2*] Huanqiu Zhang,[1,3] Terrrence Sejnowski [1,2*]
[1]Computational Neurobiology Laboratory Salk Institute for Biological Sciences, La Jolla, CA
[2] Division of Biological Sciences, University of California, San Diego, La Jolla, CA
[3] Department of Neurosciences, University of California, San Diego, La Jolla, CA
*To whom correspondence should be addressed
Email: chenyusi151201@gmail.com (Y.C); terry@salk.edu (T.S.)

Summary: We linked the temporal precision of neural encoding to the implementation of cognitive functions through predictive sequence learning.

The hippocampus of rodents receives sequences of sensory inputs from the cortex during exploration and then encodes the sequences with millisecond precision despite inter-regional transmission delays. Our study linked such temporal precision to the cognitive functions of hippocampus in a self-supervised recurrent neural network that was trained to predict its next input. The model exhibited localized place cells and experimentally observed features such as one-shot learning, replay and phase precession. We tested and confirmed the assumption that area CA3 is a predictive recurrent autoencoder by analyzing the spike coupling between simultaneously recorded neurons in hippocampal subregions. These results imply that the place field activity of neurons in area CA1 report temporal prediction error, which decays with familiarity.

Introduction

The spatiotemporal patterns of spikes in cortical structures is used for both representing and processing information (*1, 2*). During locomotion, the timing of spikes precesses relative to the phase of the traveling wave of activity in the hippocampus (*3*). Spike timing is also precisely regulated at the millisecond level for spike-timing dependent plasticity (STDP) (*4, 5*). This regulation must take into account time delays for both conduction of spikes between neurons and transmission delays at synapses. We will focus here on the functional implications of this precision for how temporal sequences of spikes are shaped by neural circuits. We model the hippocampal formation to demonstrate how the temporal precision of spike timing coupled with anatomical wiring could support cognitive functions.

The hippocampal circuit encodes a cognitive and predictive map, with place cells in rodents responding not only to locomotion signals (*6*) but also to other sensory stimuli, such as reward (*7*), auditory tones (*8*), odors and time (*9*). These stimuli are high-dimensional and highly redundant, yet only a few hippocampal neurons are reliably and repetitively activated in a short time interval, forming a low-dimensional dynamical trajectory in activity space (*10*). The hippocampus therefore learns how to encode high-dimensional sensory and motor signals at the apex of cortical hierarchies into low-dimensional, latent, non-redundant, sequential representations that ultimately support abstract representational learning. After learning sequences of events, the hippocampus then replays them during sleep and immobility when external inputs to the cerebral cortex are suppressed (*11*).

Existing computational frameworks (*12–15*) have successfully modeled cognitive functions of the hippocampus and reproduced the statistics of place cell under various task conditions. However, these models do not provide a mechanistic implementation of these cognitive functions or account for the distinct encoding and firing properties of neurons in subregions CA3,

CA1, and the dentate gyrus (DG) (*16–20*). For example, CA1 neurons respond more than neurons in other regions to unexpected signals (*21–23*) and their activity decays in familiar environments (Fig.S1). In contrast, the recurrent CA3 subregion stores an internal representation of sequences, generating replay (*24*) and preplay (*25*) and neural representations of space are generally more stable in CA3 than in CA1 (*17, 19*).

Predictive coding of visual inputs efficiently encodes features in lower cortical layers, allowing higher layers to encode more abstract representations (*26, 27*). In this study, we extend predictive coding to the temporal domain to model the interactions between subregions in the hippocampal formation. We show that this framework accurately captures the functional computational hierarchy in these subregions, which is not perfectly aligned with anatomical connectivity. Using our proposed model, we are able to simulate qualitative place cell statistics across various cognitive tasks and make quantitative predictions that match neural recordings. We also propose a novel loss function for recurrent neural networks and demonstrate its universality for learning and generating sequences.

Temporal Prediction Hypothesis

Figure 1A summarizes the major connectivity in the hippocampal formation. The entorhinal cortex (EC) is the major cortical input to and output from the hippocampus, which contains subregions DG, CA3 and CA1. Among them, recurrently connected CA3 is ideal for storing internal states in the form of attractor dynamics (*28*). Interestingly, there are two pathways projecting from the EC to CA1: an indirect pathway via DG and CA3 and a direct pathway from the EC. Moreover, the two pathways are delayed to different extents because there are more synaptic delays in the indirect path through CA3 (*29*). Assuming a synaptic transmission delay $\tau > 0$, signals transmitted through the indirect pathway to CA1 are delayed by 3τ while those going through the direct pathway are only delayed by τ. The function of this seemingly

redundant and asynchronous transmission suggests that *CA3 may be making predictions about future inputs*, which can then compared at CA1 with the less delayed teacher signal from the direct pathway. From the viewpoint of CA1, the prediction made by CA3 can be compared with future inputs from the direct pathway. This comparison is similar to a Bayes filter (*30*) where future predictions based on currently available information are compared with future observations to update the model. Prediction errors are computed at CA1 and used to refine the internal model stored at CA3. In this way, interactions between the cortex and the hippocampus form a self-supervised loop, which enables the circuit motif to learn and remember the latent variables represented in CA3 as sequences.

Neural evidence for transmission delay and prediction

To verify the above hypothesis, we analyzed simultaneously recorded neural activities from these subregions for evidence of transmission delay and predicting ahead. Assuming that neural signal propagation strictly follows the anatomical organization of the hippocampal formation in Fig. 1A, then signals encoded by a region should be correlated with the upstream signal shifted by an interregional delay. Ideally, if a location-sensitive neuron in EC has a bell-shaped response curve $f(x)$, where x represents any arbitrary physical variable such as location, its direct downstream DG neuron should exhibit a response curve of $f(x - \tau)$ where τ refers to the uniform interregional delay (Fig. 2A). Similarly, the response curves of their downstream neurons in CA3 and CA1 should be $f(x - 2\tau)$ and $f(x - 3\tau)$, respectively (dashed lines in Fig. 2A). Alternatively, if, according to our hypothesis, CA3 is predicting future signals to match the signal arrived from the direct pathway, CA3 and CA1 would have response curves of $f(x)$ and $f(x - \tau)$, respectively (solid lines in Fig. 2A), given similar interregional delays. Although it is unlikely to record from directly coupled neurons, a cross-correlation type analysis should reflect interregional spike coupling properties (Fig. 2BCD, upper panel). According

to our hypothesis, 1) similarity measures should peak at zero for CA1 and DG spike trains, indicating information synchrony in these two regions and the dominance of signals delayed by τ in CA1 (Fig. 2B, upper panel); 2) CA3 spike trains should couple tightly with leftward-shifted DG spike train (Fig. 2C, upper panel), indicating that CA3 firing leads DG. This suggests that CA3 is predicting ahead since it is anatomically downstream from the DG. For both the prediction and no-prediction scenarios, CA3 should always leads CA1 by one synaptic delay (Fig. 2D, upper panel), a measure of interregional delay.

Since the existence of transmission delay is intrinsic to the circuit, independent of the behavior state, we used the visual encoding neuropixel dataset from the Allen Brain Observatory (*31*). This dataset contains simultaneous recordings of neural spikes at a sampled at 30 kHz in DG, CA3 and CA1 from mice performing passive visual perception tasks. The high temporal precision enabled us to investigate spiking timing accuracy on the time scale of milliseconds.

Following the methods in Siegle et al. (*31*), we calculated the jitter-corrected cross correlagram (CCG) of spike trains between pairs of subregions over all stimulus conditions and plotted the distribution of optimal shifts where CCG peaks (Fig. 2BCD) (Methods). To access higher-order statistical relationship, we also calculated the mutual information between the shifted spike trains since we are interested in the amount of delay in the information transmitted by the spike trains. In Fig. 2B, we compared the unshifted CA1 spike train with shifted DG spike trains. From cross-correlogram analysis (Fig. 2B, middle row), the distribution of optimal shifts was bimodal, indicating that there may exist two heterogeneous neural populations in CA1 receiving inputs from two pathways. However, from the mutual information analysis, the distribution of optimal shifts is approximately a normal distribution with median value of zero. This means that neurons in CA1, despite some randomness, are synchronized with those in DG assessed by mutual information. They were both delayed by one synapse with respect to EC. Similar comparisons were made in Fig. 2C between shifted DG activity and unshifted CA3

activity. We found that both similarity measures peaked when DG shifted significantly towards the left for a median of 2 millisecond. Both observations supported the hypothesis. Finally, when we used shifted CA3 spike trains to predict unshifted CA1 spike train in Fig. 2D, they coupled most strongly when CA1 was shifted to the right. Thus, unsurprisingly, CA3 was ahead of CA1 activity by 2 millisecond, which matched the previously reported synaptic delay (*29*). Taken together, neural encoding in CA3 is ahead of DG while CA1 is synchronized with DG, confirming the hypothesis that CA3 is predicting ahead.

Learning sequences explains place cell response properties

We next constructed a model of the hippocampal circuit that could replicate a wide range of experimental observations across tasks based on a predictive recurrent autoencoder (PredRAE) that learned to match its output (o) to its input (x) (Fig. 1B, Eq. 1). The input and recurrent layer models EC and CA3. The CA1 response is computed as the difference between output prediction from CA3 and actual input signal (ReLU($x - o$)). Predicting ahead was incorporated through a predictive loss function (Fig. 1C): the mean square error (MSE) between future predictions ($o_{t+1|t}$) and future teacher signals (x_{t+1}). To find a network with the desired functionality, we trained the parameters W, U, V and b in Eq. 1 using Back Propagation Though Time (BPTT) (*32, 33*).

$$
\begin{aligned}
h_t &= \tanh(Wh_{t-1} + Ux_t + b) \\
h_{t+1} &= \tanh(Wh_t + b) \\
o_{t+1|t} &= \text{Sigmoid}(Vh_{t+1}) \\
L &= \sum_t L_t = ||o_{t+1|t} - x_{t+1}||^2
\end{aligned}
\tag{1}
$$

We first simulated a rat running along a circular track with constant velocity (where time

and location are equivalent). All neurons in the recurrent layer (CA3) receive location specific bell-shaped input activity (Ux_t in Eq. 1) representing their place fields (Fig. 3A, upper panel). A new environment was modeled as a random shuffle of place fields (Fig. 3A, lower panel). The network was trained on the first environment using the loss function in Eq. 1 PredRAE successfully remembered the sequence of inputs from the first environment. It was then trained on a second environment, a different sequence. At the beginning of training, the responses of CA1 error-encoding neurons were high, but gradually became weaker as the training continued since the CA1 activity represented sequence prediction error. The gradual decay of the response (Fig. 3B, upper) resembled neural recordings from CA1 neurons extending over one month (Fig. 3B, lower) (*34*). A separate analysis also showed that recorded CA1 unit activity decreased with familiarity of the environment (Fig. S1). The rapid neural response upon exposure to a novel environment, pointed by the red arrow in Fig. 3B, Modifying synaptic weights abruptly (*35*) might be dangerous for maintaining system stability if loosely regulated. In this study, we offer an alternative explanation for apparent one-shot learning: neural activity was immediately high in a new environment because CA1 reflects one-shot error, not one-shot learning.

After being trained to remember the two environments (Fig.3A), PredRAE exhibited replay and prediction, a consequence of attractor dynamics (Fig. 3C). Replay refers to the re-activation of place cells in the same order as they would during active exploring. Typically, this occurs when the animal is in a state of sleep or immobility, meaning that the simulated agent is not receiving any external sensory inputs. When low magnitude random noise was used to drive the trained network, it randomly reproduced one of the learned sequences (Fig. 3C, upper). Place cells also show predictive activities: cells that code for possible future locations are activated before making turns (*36*). This is a consequence of pattern completion by the recurrent network model. To demonstrate this, we gave the network partial input, then it completed the remaining sequences (3C, lower).

These two experiments are strong evidence for attractor dynamics in the network: state-space analysis shows that the system state converges robustly to a fixed sequential pattern, determined by network topology (*37*). We reordered the learned recurrent weight matrix based on the activation order of the hidden units (3D, upper). The reordered matrix has a Toeplitz form, which enables a convolution operation after matrix multiplication in the recurrent layer. The approximate kernel (3D, lower) resembles a Mexican hat, with positive values in the middle and negative values padding both sides.

The weights in the trained network are also precise enough to produce phase precession: the timing of place cell firing with respect to the phase of the oscillatory population activity becomes progressively earlier when traversing a place field (*1*). To generate biologically relevant action potentials, we transferred the learned recurrent weight to a network of leaky-integrate-fire (LIF) neurons following the procedure described in (*38*) and recorded the emitted spikes (Fig. 3E). Oscillatory activity was artificially enforced by injecting 8 Hz inhibitory currents, mimicking oscillatory inputs onto inhibitory neurons originating in the septal nucleus. Spike phases were calculated and plotted against their relative location to place field centers. We observed precession of the spike timing (Fig. 3F) similar to that in in vivo recordings (Fig. 1 in (*1*)).

We next simulated a rat in a random foraging task consisting of straight trajectories and random turns in a square arena (Fig. 4A). Input was handcrafted as a nonlinear mixture of body direction, world direction, path integrated distance and distance to the closest wall (Methods) (*39*). The selection of these input dimensions was based on the existence of head direction cells, path-integration properties and border cells in EC. Any one of these inputs alone is not able to determine the agent's current location, as evidenced by the low mutual information per second (MI) between input unit activity and location (Fig. 4 C). To enforce a sparse representation in the hidden layer, we added a regularization penalty of unit activity to the MSE loss function

(Methods).

After training using the predictive loss function (Eq. 1), the latent variables learned by PredRAE (i.e. hidden unit activity) were spatially localized representations similar to place fields. (Fig. 4B). In comparison to networks trained with the current loss function control, hidden units in networks trained with the predictive loss function displayed significantly higher MI (Fig. 4C). Both networks contributed to the extraction of locations as hidden units have much higher MI compared to the original input signal. This indicates predicting ahead is beneficial to extracting location information from upstream inputs. The same results were found when the regularization strength was varied (Fig. S4), suggesting that a temporal predictive loss function consistently aids in forming a localized representation.

Computation advantages of learning future sequences

To explore whether the predictive loss function could achieve representational learning, supporting downstream processing such as classification, memorization or prediction (*40*), we used inputs from MNIST handwritten digits, action sequences of images and rotated images.

We first trained PredRAE to recall sequences of increasing MNIST digits (Fig. 5). Multiple batches of randomly sampled images were temporally ordered based on their labels. We extracted the first 68 principal components of the images, which account for 86% variance, as the input to keep the minimal network structure. Image reconstruction was based on the network output and inverse transformation of PCA. After training, PredRAE output generic MNIST digits (Fig. 5A, middle) since the different realizations of hand-written digits between batches are difficult to predict. After training, PredRAE continued to complete the sequences when the input was stopped after digit 3 (Fig. 5A, bottom). This predictive completion was not observed in a control models trained with the current loss function (Fig. S2).

Interestingly, PredRAE not only constructed a generative model to predict sequences but

also automatically clustered the digits according to their labels. In Fig. 5D, we plotted the independent components (IC) of the hidden unit activity for each digit and colored the digits according to their labels. For the top 10 ICs (sorted by the contribution of demixing matrix), approximately each component represents one group of digits. This interpretable representation was achieved without explicitly defined labels. The sequential activation of ICs along the time axis (Fig. 5D, bottom panel) reflects the *attractor dynamics* in the linear transformed space, which can separated downstream by a single layer of weights.

PredRAE can also learn to classify action sequences (Fig.6), a cognitive function of the hippocampus. In this task the goal is to cluster similar memories together and then name the group with a shorter "codename". For example, if someone receives a message "biking", he might recall memories of biking in the mountain, on the road, etc rather than swimming. We used the sprites dataset (*41*), which contains 1,000 different characters performing nine actions (Fig. 6A). We equipped PredRAE with one convolutional layer before the RNN input and one deconvolutional layer after the RNN output, a limited form of visual image processing. It was then trained on 9,000 sequences (Methods) to reproduce the next image in each sequence. We found that sequences performing the same actions clustered together in the IC space of hidden unit activity (Fig. 6B). To quantify clustering performance, we calculated the action group variability within the three top ICs (Fig. 6C). A lower variability means a tighter clustering behavior. Within group variability is significantly lower in networks trained using predictive loss compared to those of current loss function controls (Fig. 6D). This task is a form of hash coding and the predictive loss function is effective at hashing memories.

PredRAE can also learn to perform geometric operations on images (Fig. 7). PredRAE was trained to reconstruct a sequence of MNIST digits rotating 30 degree counterclockwise on each time step (Fig. 7A). As before, it was able to rotate the given image after the input was stopped (Fig. 7B). Moreover, it generalized the rotation operation to test images although the

reconstruction performance was not as good because PredRAE was never trained to reconstruct the test images (Fig. 7C). The rotation dynamics could be recovered in the first three principal components (PC) of the hidden unit activity (Fig.7D). This was not possible in control models with the current loss function (Fig.S3).

Predictive Autoencoders in the Cortex

The superficial and deep layers of the six-layered neocortex can be identified with areas CA3 and CA1 of the hippocampus, respectively. This is illustrated in Fig. 8, which shows parallels between the indirect pathway through the DG in the hippocampus projecting to CA3 corresponding with the thalamic inputs to layer 4 of of the neocortex projecting to layers 2/3, and direct pathway from the cortex to area CA1 in the hippocampus corresponding in the cortex to direct inputs from the thalamus to layer 5.

We propose that in the cortex, the recurrent network in layers 2/3 is trained as a predictive autoencoder to remember sequences of inputs arising from the thalamus. The EC in Fig. 8 is at the top of a hierarchy of cortical areas, each recapitulating the same architecture, forming a stack of predictive autoencoders. There are also feedforward projections from layers 2/3 to layer 4 in the cortical hierarchy. The latent sequences from the upstream cortical areas are further compressed in the downstream areas, becoming more abstract while ascending the hierarchy. This generalization of our hippocampal model could be tested by recording simultaneously from the thalamus, layers 2/3 and layer 5 and analyzing the spikes the same way we analyzed recordings from the corresponding areas of the hippocampus, DG, CA3 and CA1.

Discussion

The temporal predictive coding framework proposed here to account for sequence memory and representation learning was inspired by the anatomy of the hippocampus, validated by neural

recordings and successfully replicated a variety of experimental observations. We also found that multiple sequences could be stored in the same recurrent network in a sparse Toeplitz form. This kernel underlies the moving-bump line attractor dynamics in recurrent neural networks that has been used to model compass cells in rodents (*42*) and similar attractor dynamics in other neural systems (*43, 44*). Connectomic analysis of the rodent area CA3 could confirm the predicted Toeplitz connectivity.

Previous studies have also explored the anatomy and firing properties of hippocampal neurons (*45, 46*), based on a wide range of theoretical and computation principles (*12–14, 39*). Stachenfeld et al. (*12*) modeled the activities of place cells as the successor representation in reinforcement learning; Whittington et al. (*13*) proposed that the hippocampus is a statistical machine for inferring structural properties from observations. These studies are complementary to ours. Our approach to how the hippocampus functions is more mechanistic, based on the encoding and statistical properties from neural recordings in hippocampal subregions. The simplicity and minimal realization of our model could also serve as critical building blocks to achieve successor learning and statistical inference.

Predictive loss functions are routinely used in state-space models, such as model-based control and Bayesian filtering. By predicting the future observations, generative models are continually updated to make more accurate predictions (*47*). This approach has recently been incorporated in several model-based artificial intelligence systems (*48, 49*). The performance of these systems is more robust and has superior generalizability when there is an internal model of the system. Our model-based approach to the hippocampus has similar advantages and is supported by evidence from neural recordings.

Researchers who routinely decode the location of freely moving animals from CA1 place cell activities may find it counterintuitive that CA1 neurons encode prediction error. The decay of neural activity in place cells shown in Fig. 3B occurred over a month, which is not apparent

in most studies that follow the same neurons for only a few days when substantial location information may still be present in residual prediction errors. Our hypothesis is a computational explanation for the observed decay of place fields. Neural encoding is more coherent and place fields are longer lasting in CA3 than in CA1 (*18*). This suggests that once a sequence is accurately predicted by CA3, and error signals are no longer found in CA1, the memory of the sequence continues to be retained in CA3 for replay.

Not all neurons with place fields in CA1 decay and neurons with new place fields are formed over many weeks (*34*). This slow tunover of place fields could be explained by nonstationarity in sensory input from multiple modalities, which is highly probabilistic and differs from trial to trial. For example, auditory or olfactory input could vary over time even as a rat continues to explore the same environment. The continual learning of new cues could both slow down the decay of place fields in CA1 and create new ones. Another sign of variability among CA1 neurons was found in the heterogeneity of the peaks in the cross-correlogram analysis of neuron populations in CA1 in Fig. 2B.

Representing prediction errors in some CA1 neurons could not only help improve the internal model stored in the recurrent CA3 network, but could also regulate the release of dopamine in novelty-dependent firing of cells in ventral tegmental area (VTA) through subiculum, accumbens, and ventral pallidum (*50*). This could explain why novelty detection is essential in the hippocampus.

Although we used back-propagation through time as a way to construct networks that learn sequences, it can be potentially replaced by local learning rules (*27*) because prediction error is local in both time and space. It might nonetheless be difficult to accumulate gradients over a long sequence. In this scenario, CA3 would behavior like a reservoir (*51*), generating a wide range of time varying signals and the prediction error signal could be used to select inputs and weight them to reduce the prediction error. There are also increasing efforts to develop

online training algorithms for recurrent neural networks through the approximation of target gradients (*52, 53*).

The above circuit, featured by dual pathways and predicting ahead, is not unique to hippocampus. In cortical layers, we could observe similar structure (Fig. 8). Inputs from thalamus propagate to the recurrent layers 2/3, via sparsely firing layer 4 granule neurons, for prediction; Predicted output is being compared in layer 5 with the direct input it receives from thalamus. This conserved circuit and predictive computation is very likely to underlie the robustness and flexibility of human intelligence.

Self-supervised models, such as variational autoencoders (*54*), unsupervised Boltzmann Machines (*55, 56*) and their many variants, have avoided labor-intensive supervised input labeling. The recent success of transformers like GPT were trained by predicting the next word in a sentence (*57*). These sophisticated models implicitly learn semantic representations. In the same way, PredRAE achieves representation learning of sequences without needing sophisticated statistical priors or explicitly defined representation modules. Its minimal model structure will facilitate mechanistic investigation and interpretation. Future work could focus on scaling up the network or adding preprocessing modules to handle more realistic problems such as the semantic segmentation of video clips.

Temporal prediction error has already been established for learning sequences of actions in the basal ganglia to obtain future rewards (*58*). If temporal predictive coding principles for learning sequences are also found in the hippocampus and the cortex, as suggested in Fig. 8, then predicting the next input in every cortical area may be as important a design principle for human cortical function as predicting the next word in a sentence is for the transformers in large language models (*57*). Inspired by brains, PredRAE has potential for further improvements in robustly disentangling representations in artificial intelligence and approaching human levels of performance.

References

1. M. V. Tsodyks, W. E. Skaggs, T. J. Sejnowski, B. L. McNaughton, *Hippocampus* **6**, 271 (1996).
2. C. D. Harvey, F. Collman, D. A. Dombeck, D. W. Tank, *Nature* **461**, 941 (2009).
3. J. Patel, S. Fujisawa, A. Berényi, S. Royer, G. Buzsáki, *Neuron* **75**, 410 (2012).
4. G.-q. Bi, M.-m. Poo, *Annual review of neuroscience* **24**, 139 (2001).
5. S. Song, L. F. Abbott, *Neuron* **32**, 339 (2001).
6. J. O'keefe, L. Nadel, *The hippocampus as a cognitive map* (Oxford university press, 1978).
7. J. L. Gauthier, D. W. Tank, *Neuron* **99**, 179 (2018).
8. D. Aronov, R. Nevers, D. W. Tank, *Nature* **543**, 719 (2017).
9. G. Buzsáki, D. Tingley, *Trends in cognitive sciences* **22**, 853 (2018).
10. E. H. Nieh, *et al.*, *Nature* **595**, 80 (2021).
11. L. R. Squire, *Neuron* **61**, 6 (2009).
12. K. L. Stachenfeld, M. M. Botvinick, S. J. Gershman, *Nature neuroscience* **20**, 1643 (2017).
13. J. C. Whittington, *et al.*, *Cell* **183**, 1249 (2020).
14. D. C. McNamee, K. L. Stachenfeld, M. M. Botvinick, S. J. Gershman, *Nature Neuroscience* **24**, 851 (2021).
15. D. George, *et al.*, *Nature communications* **12**, 2392 (2021).
16. S. Leutgeb, J. K. Leutgeb, A. Treves, M.-B. Moser, E. I. Moser, *Science* **305**, 1295 (2004).
17. J. Shin, H.-W. Lee, S.-W. Jin, I. Lee, *Cell Reports* **41**, 111823 (2022).
18. I. Lee, D. Yoganarasimha, G. Rao, J. J. Knierim, *Nature* **430**, 456 (2004).
19. I. Lee, G. Rao, J. J. Knierim, *Neuron* **42**, 803 (2004).
20. A. C. Schapiro, N. B. Turk-Browne, M. M. Botvinick, K. A. Norman, *Philosophical Transactions of the Royal Society B: Biological Sciences* **372**, 20160049 (2017).
21. D. Kumaran, E. A. Maguire, *PLoS biology* **4** (2006).
22. R. T. Knight, *Nature* **383**, 256 (1996).
23. K. Duncan, N. Ketz, S. J. Inati, L. Davachi, *Hippocampus* **22**, 389 (2012).
24. H. R. Joo, L. M. Frank, *Nature Reviews Neuroscience* **19**, 744 (2018).

25. G. Dragoi, S. Tonegawa, *Nature* **469**, 397 (2011).

26. W. Lotter, G. Kreiman, D. Cox, *Nature Machine Intelligence* **2**, 210 (2020).

27. R. P. Rao, D. H. Ballard, *Nature neuroscience* **2**, 79 (1999).

28. J. J. Hopfield, *Proceedings of the national academy of sciences* **79**, 2554 (1982).

29. B. Sabatini, W. Regehr, *Annual review of physiology* **61**, 521 (1999).

30. Wikipedia contributors, Recursive bayesian estimation — Wikipedia, the free encyclopedia (2021). [Online; accessed 8-May-2022].

31. J. H. Siegle, *et al.*, *Nature* **592**, 86 (2021).

32. A. E. Bryson, *Proceedings of the Harvard Univ. Symposium on digital computers and their applications* (1961).

33. P. J. Werbos, *Proceedings of the IEEE* **78**, 1550 (1990).

34. Y. Ziv, *et al.*, *Nature neuroscience* **16**, 264 (2013).

35. K. C. Bittner, A. D. Milstein, C. Grienberger, S. Romani, J. C. Magee, *Science* **357**, 1033 (2017).

36. K. Kay, *et al.*, *Cell* **180**, 552 (2020).

37. C. Eliasmith, *Scholarpedia* **2**, 1380 (2007). Revision #91016.

38. R. Kim, Y. Li, T. J. Sejnowski, *Proceedings of the national academy of sciences* **116**, 22811 (2019).

39. M. K. Benna, S. Fusi, *Proceedings of the National Academy of Sciences* **118**, e2018422118 (2021).

40. Y. Bengio, A. Courville, P. Vincent, *IEEE transactions on pattern analysis and machine intelligence* **35**, 1798 (2013).

41. Y. Li, S. Mandt, *International Conference on Machine Learning* (2018).

42. K. Zhang, *Journal of Neuroscience* **16**, 2112 (1996).

43. R. J. Gardner, *et al.*, *Nature* **602**, 123 (2022).

44. M. Khona, I. R. Fiete, *Nature Reviews Neuroscience* **23**, 1 (2022).

45. J. Basu, S. A. Siegelbaum, *Cold Spring Harbor perspectives in biology* **7**, a021733 (2015).

46. T. Klausberger, *European Journal of Neuroscience* **30**, 947 (2009).

47. T. Isomura, T. Toyoizumi, *Nature Machine Intelligence* **3**, 434 (2021).

48. S. Recanatesi, *et al.*, *Nature communications* **12**, 1 (2021).

49. W. Lotter, G. Kreiman, D. Cox, *arXiv preprint arXiv:1605.08104* (2016).

50. J. E. Lisman, A. A. Grace, *Neuron* **46**, 703 (2005).

51. H. Jaeger, H. Haas, *science* **304**, 78 (2004).

52. J. M. Murray, *Elife* **8**, e43299 (2019).

53. G. Bellec, *et al.*, *Nature communications* **11**, 1 (2020).

54. D. P. Kingma, M. Welling, *arXiv preprint arXiv:1906.02691* (2019).

55. R. Salakhutdinov, A. Mnih, G. Hinton, *Proceedings of the 24th International Conference on Machine Learning* (2007), pp. 791–798.

56. G. E. Hinton, T. J. Sejnowski, *et al.*, *Parallel distributed processing: Explorations in the microstructure of cognition* **1**, 2 (1986).

57. A. Vaswani, *et al.*, *Advances in Neural Information Processing Systems*, I. Guyon, *et al.*, eds. (Curran Associates, Inc., 2017), vol. 30.

58. M. Watabe-Uchida, N. Eshel, N. Uchida, *Annual Review of Neuroscience* **40**, 373– (2017).

59. O. Gökçe, T. Bonhoeffer, V. Scheuss, *Elife* **5**, e09222 (2016).

60. N. T. Markov, *et al.*, *Science* **342**, 1238406 (2013).

61. K. Mizuseki, A. Sirota, E. Pastalkova, K. Diba, G. Buzsáki, *CRCNS org* (2013).

62. K. Mizuseki, *et al.*, *F1000Research* **3**, 98 (2014).

63. W. Skaggs, B. Mcnaughton, K. Gothard, *Advances in neural information processing systems* **5** (1992).

Acknowledgements

Funding: DARPA W911NF1820, ONR N00014-23-1-2069. **Authors contributions:** Y.C., H.Z., T.S. conceptualized the study, wrote and revised the manuscript; Y.C, H.Z performed experiments. **Competing interests:** No competing interests to be declared. **Data and materials availability:** The Allen NeuroPixel CRCNS dataset used in the analysis is publicly available through their website. All scripts used in this study are available in https://github.com/yschen13/HCPrediction.

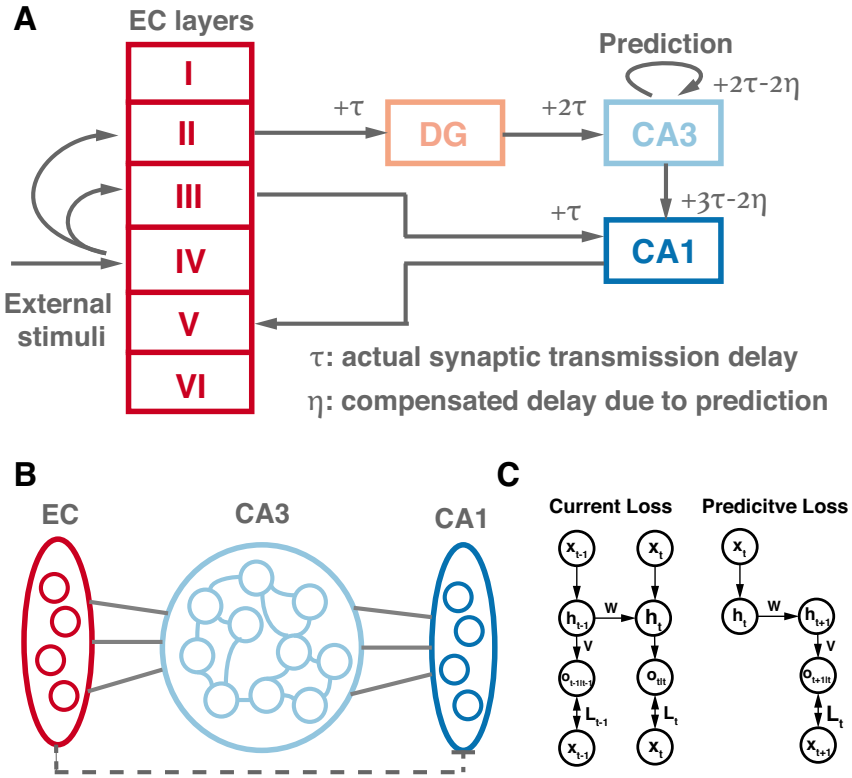

Figure 1: **The circuit within the hippocampal formation.** (A) Anatomical wiring and interregional delays. External sensory stimuli start from different cortical layers in EC and reach CA1 through two pathways, forming a self-supervised structure. Assume that spikes are delayed by τ after one synaptic transmission, they will be delayed by 3τ and τ at CA1 through the indirect and direction pathway, respectively. We hypothesize that CA3 predicts the future (-2μ) to compensate for the accumulated transmission time difference (+2τ). (B) Network model and its correspondence to the neural circuit. The input layer (EC) supplies the same input and target signal (x). The output (o) of the recurrent layer (CA3) was trained to resemble the input signal. CA1 neural response was modeled as output signal suppressed by target signal (i.e. prediction error). (C) Left: the computation graph of a conventional recurrent neural network whose loss function is Mean Square Error (MSE) between output and teacher signal at current time; Right: the computation graph of a predictive loss - MSE between future prediction and future teacher signal. $o_{t+1|t}$ means the prediction of time step $t+1$ given information at time step t. EC: Entorhinal cortex; DG: Dentate Gyrus

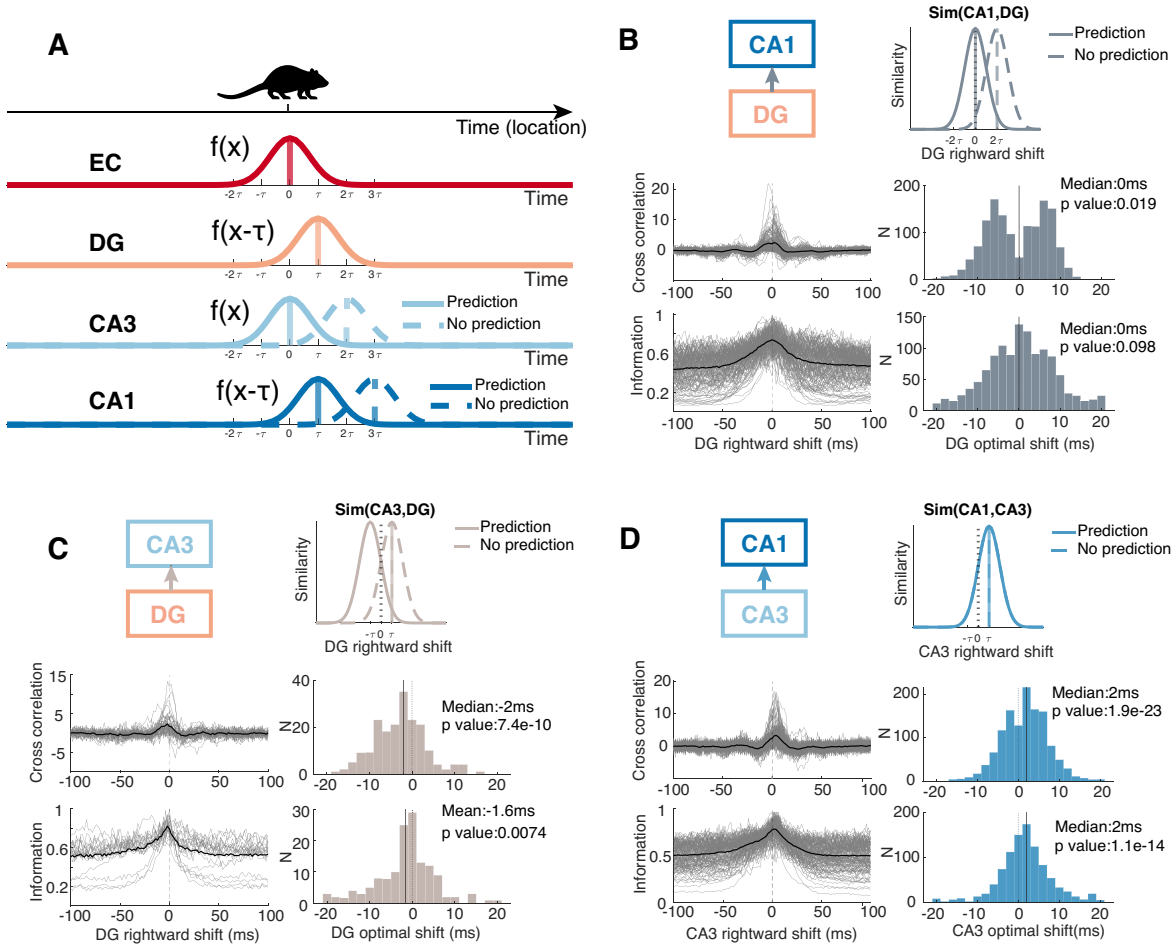

Figure 2: **Neural evidence of transmission delay and predicting ahead** (A) Schematics of delayed neural response and hypothesized predicting effect. Assume a rat running with constant velocity (time=location), one representative location sensitive neuron in EC exhibits bell-shaped response curve peaked at $t = 0$. Given there's no prediction, its direct downstream DG and CA3 neuron will peak at $t = \tau$ and $t = 2\tau$, respectively. Meanwhile, CA1 would receive mixed signals, delayed by τ and 3τ, from dual pathways. If there is prediction ahead, CA3 would instead peak at $t = 0$ and CA1 would only respond to signals peaked at $t = \tau$. (B) Spike coupling from DG to CA1. Top: schematics of spike train similarity with respect to DG neural activity shifts. Positive shift means shifting DG spike train towards the right and then computing its similarity with the unshifted CA1 spike train. Middle (Bottom): Left: Traces of corrected cross correlogram (Mutual information) from an example session. Each gray trace represents the prediction from a population of DG neurons to one CA1 neuron. The solid black trace is the average across all CA1 neurons in the session. Right: Histogram of optimal shift, where similarity measure peaks, pooled across 12 recording sessions. (p-value: t-test of population mean equals to zero) (C)(D) Spike coupling from DG (CA3) to CA3 (CA1). DG is synchronized with CA1 while CA3 leads DG by 2ms. This matches the hypothesis of CA3 predicting ahead.

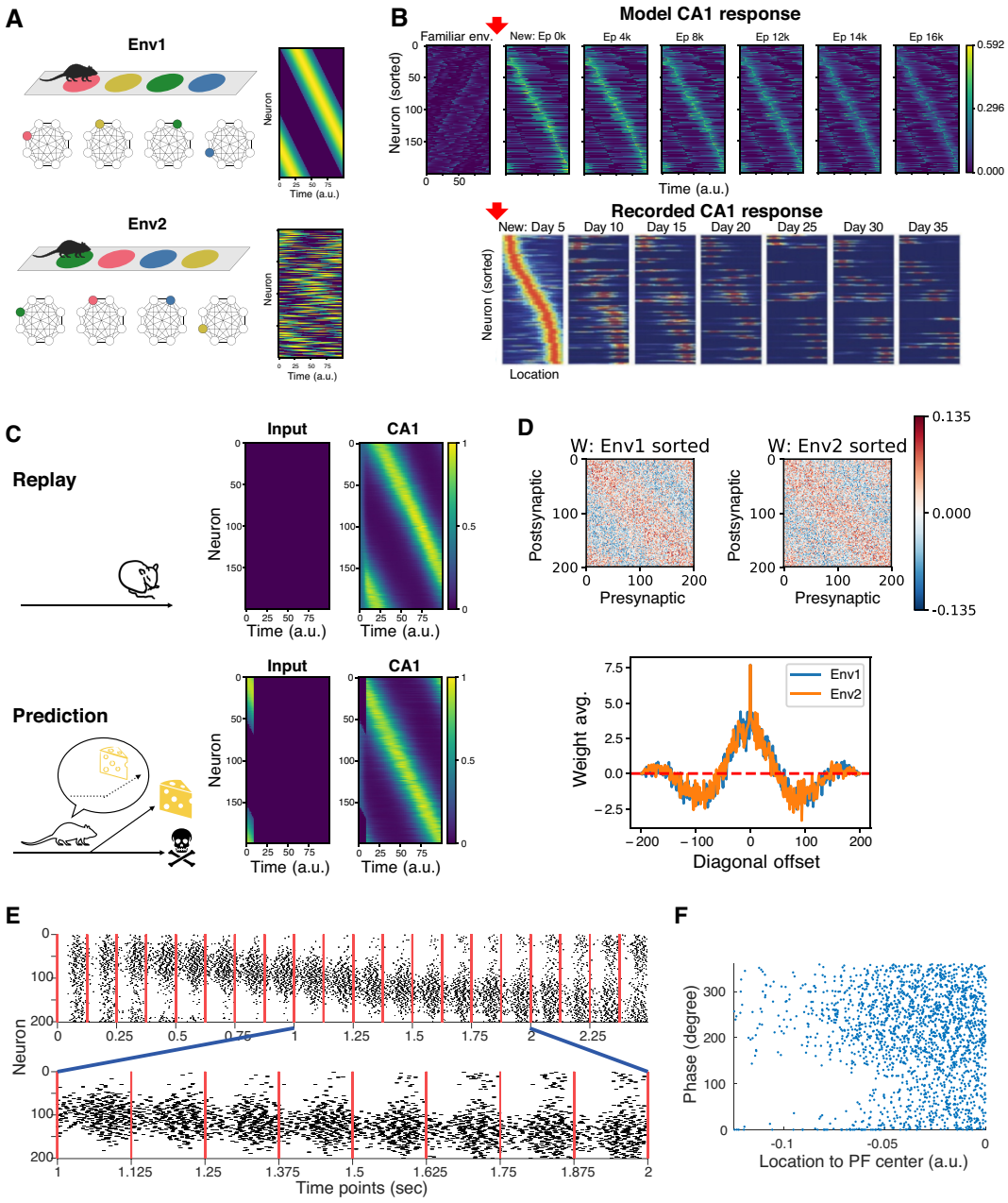

Figure 3: **PredRAE matches key place cell features, including one-shot plasticity, replay, prediction and phase precession.** (A) Input matrices (x) simulating different environments. With the equivalence of time and location, each row represents a bell-shaped location specific input current. Env: environment. (B) Long term CA1 dynamics from the model (upper) and experimental recordings (*34*) (lower). Red arrow marks the initial exposure to a second environment. The plotted neural response was sorted by their activation order. Ep: epochs; (C) Replay: Given low magnitude random input simulating spontaneous activities, PredRAE output its previously remembered pattern as CA1 neuron response. Prediction: given input of the first 10 time steps, PredRAE performed pattern completion. (D) Upper: The trained recurrent weight matrix sorted by the activation order of Env1 or Env2. Note similar entries on the diagonals; Lower: Average value of the matrix diagonals offset by the index shown on the horizontal axis. The approximate kernel here looks like a Mexican hat, pointing to the existence of line attractor dynamics. (E) Spike train recorded from a network of LIF neurons. Red vertical lines mark the trough of 8Hz population activity (i.e. phase=180 degree) (F) Spike phases (relative to 8Hz population activity) of all CA3 neurons plotted against their relative location in a place field.

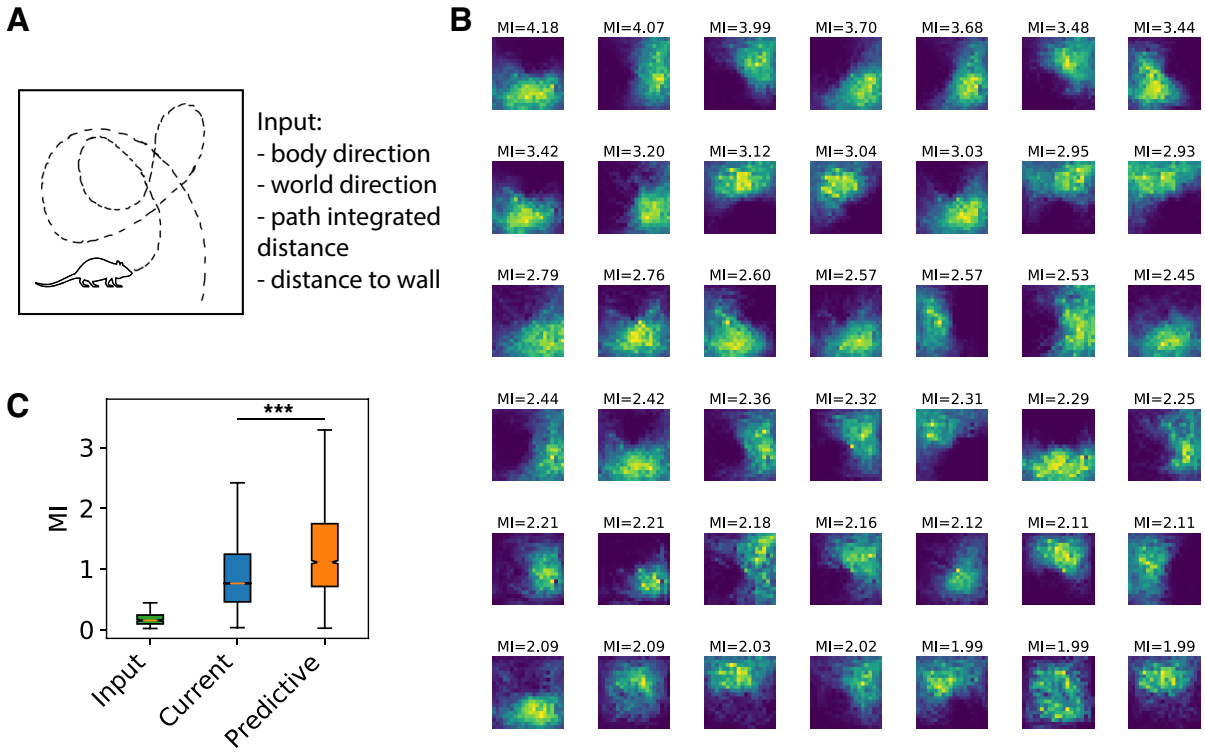

Figure 4: **Predictive networks led to more localized representation in a random foraging task.** (A) Left: Simulated rat trajectory in an open arena; Right: network input defined as a random nonlinear mixture of body direction, world direction, path integrated distance and distance to the wall. (B) Place fields of the recurrent units (CA3 neurons) after training. Extent of localization was quantified by mutual information (MI) rate per second between firing rate and location. (C) MI of the input units and recurrent units in 10 networks trained by either the current loss function as a control or the predictive loss function. Recurrent units trained by the predictive loss function showed significantly higher MI (t-test, $p < 0.001$).

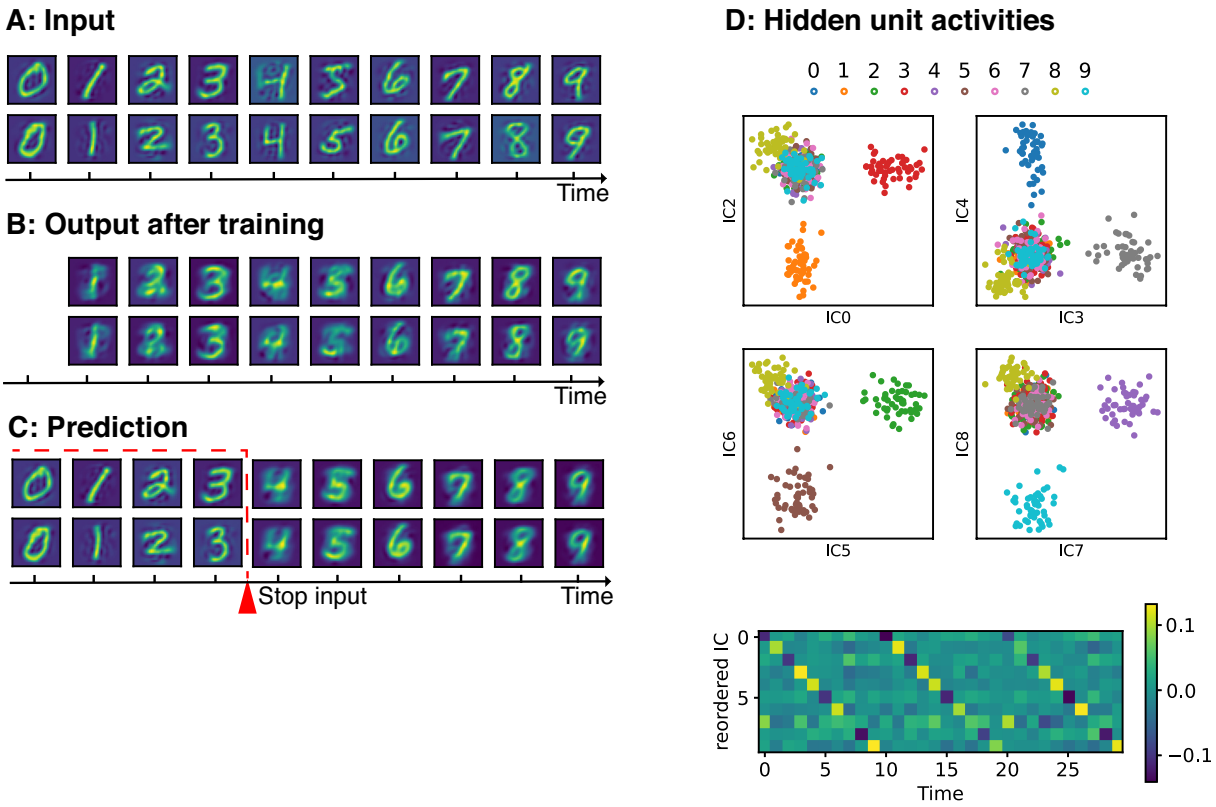

Figure 5: **Interpretable latent representation using a predictive loss function** (A) Top: Input organized as repetitive sequences of MNIST images from 0 to 9. (B) Trained network could reconstruct the given input with generic digits. (C) It could continue to predict the sequence even when the input was stopped after digit 3. Input digits were plotted before the red vertical line while predictions were plotted afterwards. (D) Top: Independent components (IC) of hidden unit activity. Most ICs represent one class (color-coded); Bottom: Reordered IC activities over time. Note the sequential activation in 10 time steps (one cycle).

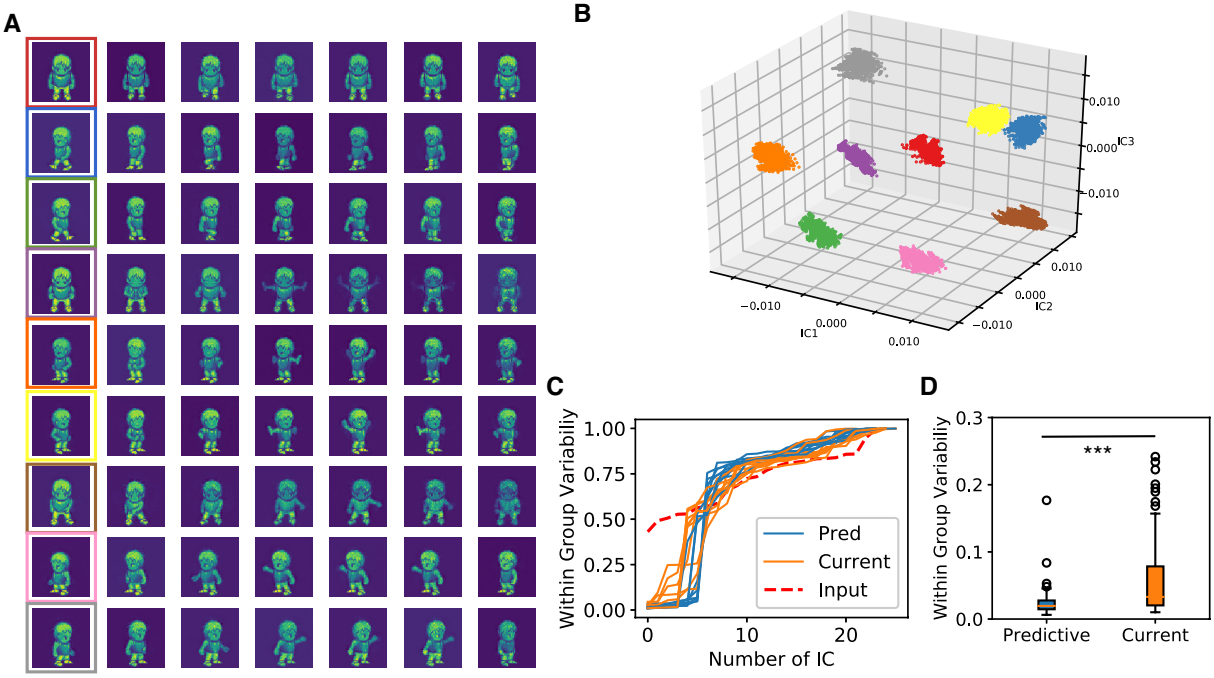

Figure 6: **PredRAE could classify action sequences.** (A) Given the initial frame in the colored square, reconstruction of future frames. (B) Independent components of hidden unit activities colored according to different action groups. Each point is one (out of one thousand) character performing one action. (C) Within action group variability as the increase of IC numbers. Each solid line is one converged network using different loss functions. The original input space was not clustered in its IC space (dashed red line). After training, the hidden unit activity of the networks trained using predictive loss function is more clustered in their IC space (blue solid lines v.s. orange solid lines). (D) Within action group variability of the top three ICs for hidden unit activities in 100 repetitive networks trained using current control or predictive loss functions (***: t-test, $p<0.001$).

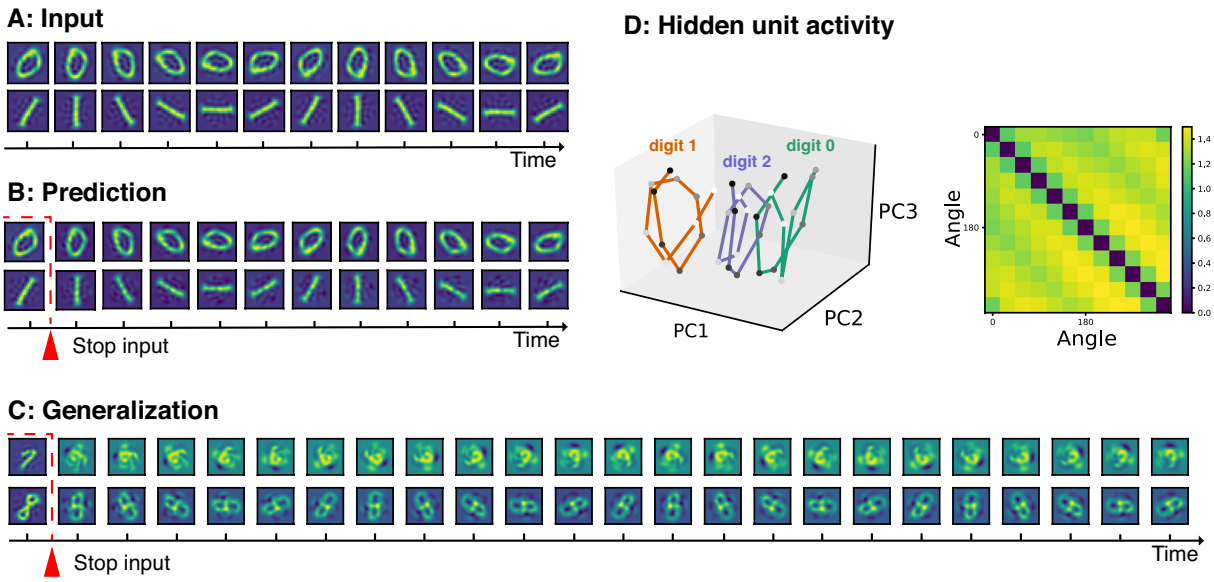

Figure 7: **PredRAE could generalize rotational dynamics.** (A) Input as a sequence of MNIST digits rotating 30 degree counterclockwise. (B) PredRAE learnt the rotation operation and kept rotating the digits after the input stops at the red vertical line. (C) Such operation worked well on novel digits (7 and 8 in this case). The entire class of 7 and 8 has never been shown to the network before. The reconstruction was not ideal because PredRAE has never learnt to reconstruct 7 or 8. It could also rotate the given image more than one cycle (two cycle shown here). (D) Left: Cycled dynamics of hidden unit activity shown in PC space. Right: the Euclidean distance matrix between different time steps averaged over all trained digits/batches. The +-1 off-diagonal stripe means cycled dynamics in the full space. The +- 6 off-diagonal stripe appeared because some digits are symmetric (like 0,1 and 8), therefore rotating 180 degree is the same as rotation 360 degree.

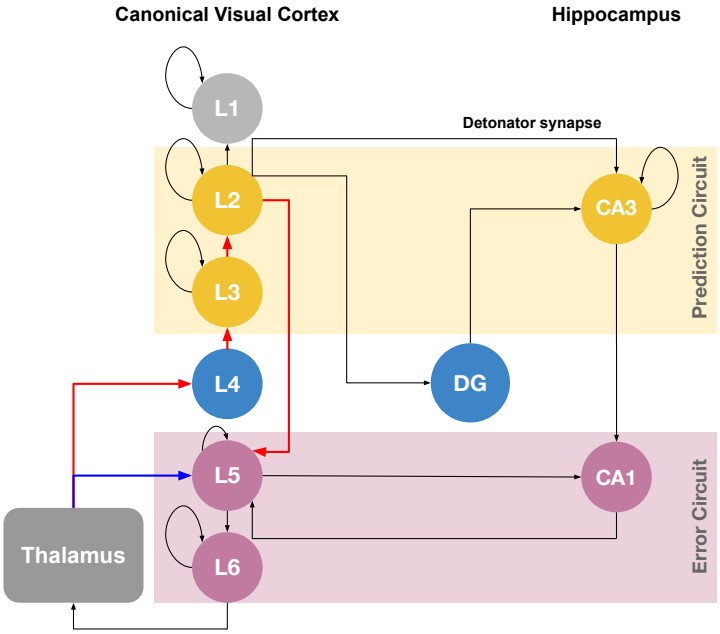

Figure 8: **Universality of the circuit in the brain.** Left: cortical layers; Right: hippocampus. Signals from thalamus reach cortex L5 through two different pathways - the blue direct pathway (*59*) and the red indirect pathway via Layer 4, recurrent Layer 2/3 (*60*). This cortical circuit resembles what we described in hippocampus: recurrent Layer 2/3 makes prediction; small stellate cells in Layer 4 may achieve the same function as DG granule cells for pre-processing; Layer 5 computes the prediction error and then propagates back to the upper structure.

Supplementary Materials

Methods

Tables 1 to 2

Figs. S1 to S4

A Methods

A.1 Neural evidence of transmission delay and predicting ahead

We used the publicly accessible visual encoding NeuroPixel dataset (*31*) from the Allen Brain Observatory. Neuropixels were used to simultaneously record the spiking activity of thousands of neurons in mice passively perceiving standard visual stimuli such as drifting gratings, natural scenes, natural movies and et al. We pre-selected recording sessions that involves recordings from DG, CA3 and CA1 in "Functional Connectivity" in WT mice. Very few neurons were recorded from EC. Number of of units being used was summarized in Table A.1. For mutual information calculation, only recordings from the natural movie viewing sessions (30 seconds $\times$ 80 repeats) were used while for cross correlation, recordings from all stimuli sessions were concatenated to increase signal-to-noise ratio.

The processed spike train was binned at 2 millisecond (ms). To compute cross-correlogram (CCG) between N neurons in region A and M neurons in region B, we first calculated jitter-corrected correlograms (*31*) between $N \times M$ neuron pairs using a jittering window of 20 millisecond. Jitter-correction was performed by randomly shuffling the spike train within the chosen time window, calculating the jitter-CCG repeatedly for 100 times, and then subtracting its average from the original CCG. Corrected-CCG was further normalized by the geometic firing rate of the neuron pair. In this way, slower time scale correlations, such as the strong theta

Session	CA1	CA3	DG
766640955	170	17	62
771160300	282	48	32
767871931	104	20	32
768515987	102	27	25
771990200	70	6	31
778240327	251	24	42
778998620	133	45	16
779839471	142	25	49
781842082	114	19	28
793224716	159	31	29
821695405	56	17	40
847657808	189	6	70

Table 1: Number of simultaneously recorded neurons from DG, CA1 and CA3 in each session

oscillation in hippocampus or nonstationary trend could be removed and then we could focus on fast time scale neural coupling. To increase signal-to-noise ratio for prediction of one neuron in region A, we used the CCG average of all M neurons in region B. Time shifting was performed in region B neurons. For a spike train denoted by $f(t)$, a positive τ shift would lead to a rightward shifted spike train of $f(t-\tau)$. The optimal time shift is defined as the time shift that maximizes the M-to-1 averaged cross correlation. We focused on time shifts from -20ms to 20ms as any shifted coupling above this range would be scrambled by jittering.

We compute the mutual information between the spike train of one neuron from region A and the shifted spike trains of 10 neurons from region B. The one-dimensional spike train from region A was treated as a random variable A. The latter high-dimensional spike train is treated as a random vector B which has 2^10 states being sampled at different time steps. The mutual information is then calculated as $I(A;B) = H(B) - H(B|A)$. For each neuron in A, we compute the information between that neuron and 10 neurons in B and repeat 100 times for different randomly sampled subsets of 10 recorded neurons from B. To show the results for one neuron in A, for each subset of 10 neurons from B, information over time shift is normalized

by its maximal value. Then the average of the 100 normalized information curves is taken to reveal the effect of time shift on the mutual information. The optimal time shift is defined as the time shift that maximizes mutual information.

A.2 Analysis of CA1 activity with respect to environment familiarity

Datasets were obtained from http://crcns.org/data-sets/hc/hc-3, contributed by the Buzsáki laboratory at New York University (*61*) (*62*). See http://crcns.org/files/data/hc3/crcns-hc3-processing-flowchart.pdf for more details about experiments, recording and data pre-processing. For rats exploring a 180 × 180-cm box, all sessions that have more than 50 simultaneously recorded CA1 neurons were included for analysis. We excluded neurons that are marked as inhibitory or not identified. For each session, we compute spike rate of the neurons during the last two-thirds of the sessions for stability of the responses.

A.3 PredRAE training

The network was implemented in PyTorch (v1.11.0) and training was performed through stochastic gradient descent of samples split into mini batches. Gradients were calculated as back-propagation through time (BPTT). The recurrent network has 200 units while the number input/output units depends on tasks. We used 'sigmoid' and 'tanh' nonlinearity for the activation of output and recurrent units, respectively. We stopped training the network when the process reached the maximum number (50,000) of epochs or the loss function didn't change more than 0.001% in 10 consecutive iterations. Network structure and related hyper-parameters were summarized in Table A.3.

A.4 Localization

The open arena was simulated as a 2m × 2m environment. Exploratory trajectory was generated as straight lines of 0.1m step size until hitting a border. Then a random turnaround angle will

Task	Sample size (S)	Input unit (N)	Sequence Length (T)	Hidden unit (H)	Loss
Fig. 3	1 or 2	200	100	200	MSE
Fig. 4	50	200	100	500	MSE + regularized h_t
Fig. 5	5	68	100	200	MSE
Fig. 6	9,000	62×62	7	500	MSE

Table 2: Details of network input and hyper-parameters used for simulation

be generated to continue exploration. Altogether 5,000 time steps split into 50 samples were used to train the network. Following (*39*) (Supplementary Eq. 1), the location information (path integrated distance, distance to the closest border, world direction and head direction) was randomly and nonlinearly expanded into higher dimensions ($N = 200$) as input and target signal. We switched to 'ReLU' nonlinearity for hidden unit activation as we would like to avoid negative responses in terms of place field calculation. To enable a sparse representation, a penalty of hidden unit firing was added to the loss function (Eq.2)

$$L = \sum_t L_t = \sum_t (||o_{t+1|t} - x_{t+1}||^2 + \lambda ||h_t||^2) \tag{2}$$

Mutual information of a hidden unit place field was calculated following (*63*) as the mutual information between firing rate and the arena location discretized into 25 $\times$ 25 grids. Specifically, It was calculated as MI $= \sum_i \lambda_i \log(\lambda_i) p_i$ in bits/second where i represents location grid, λ_i is the neuron's firing rate at location grid i and p_i is the occupancy probability in grid i.

A.5 Learning MNIST sequences

Input was constructed as the top 68 principle components (PC) of the entire MNIST dataset, which explain 87% variance. Input was organized as sequences consisting of 100 time steps, which repeats from digit 0 to digit 9 for 10 times. Five randomly sampled batches of digit images were used for training to predict the next time step PC vector. Independent component analysis (ICA) was performed to reduce the dimension of hidden unit activation from the number of hidden units to the number of chosen ICs (i.e. 10). We manually ordered the ICs by the

contribution (column L2 norm) of the converged demixing matrix. For the rotational dynamics, input was organized as 30 batches of sequences consisting of 12 time step rotating 30 degree at each time step. The 68 principle components were re-calculated using the rotated samples.

A.6 Learning Sprites action sequences

We obtained the Sprite dataset prepared by (*41*). This dataset has become a popular dataset in the field of representation learning. 1000 characters performing 9 actions (i.e. 9,000 sequences) consisting of 7 time steps were used for training. We only used the first color channel. To include more visual details, at each time step we added one time-invariant convolution layer before the input layer and another time-invariant de-convolution layer after the output layer. We used default settings in the pytorch built-in function Conv2d (kernel=3, pad=0, dilation=1, stride=1). Objective is to reconstruct the (next) input image. After training, ICA was performed to reduce the dimension of $H \times T$ to the number of chosen ICs. The ICs were ordered by the contribution (column L2 norm) of the converged demixing matrix.

B Supplementary figures

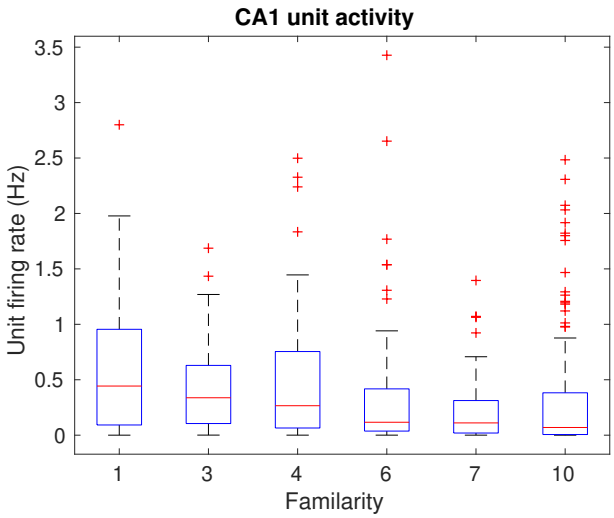

Figure S1: Activities of CA1 neurons decay as the increase of familiarity from CRCNS dataset (Methods).

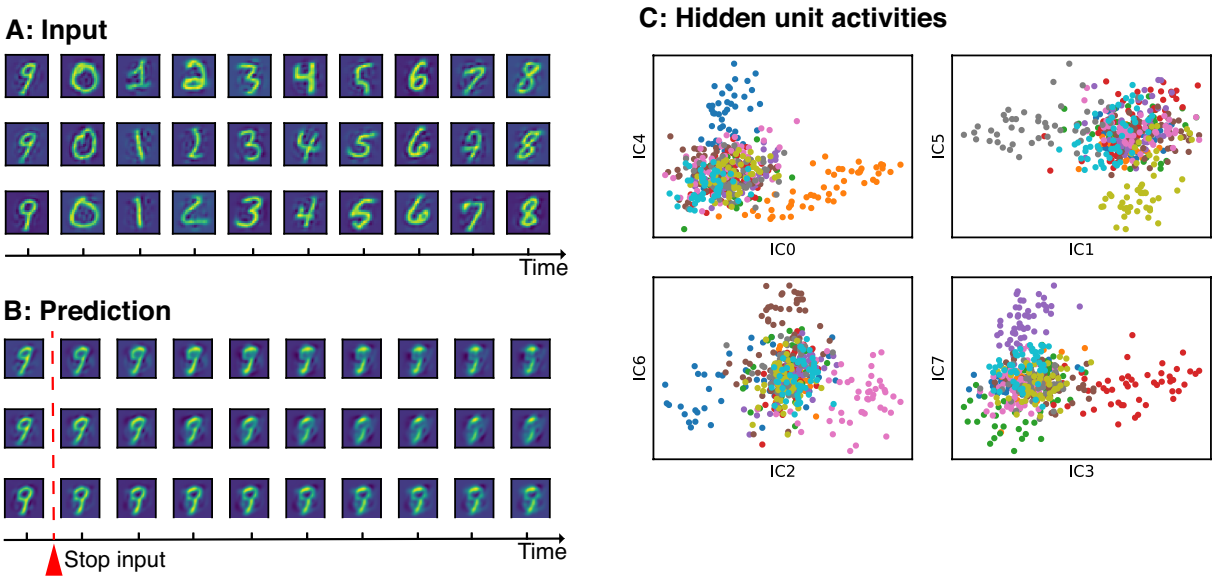

Figure S2: **Control using current loss function.** Similar to Fig. 5ABD except that the network was trained using current loss function as a control.

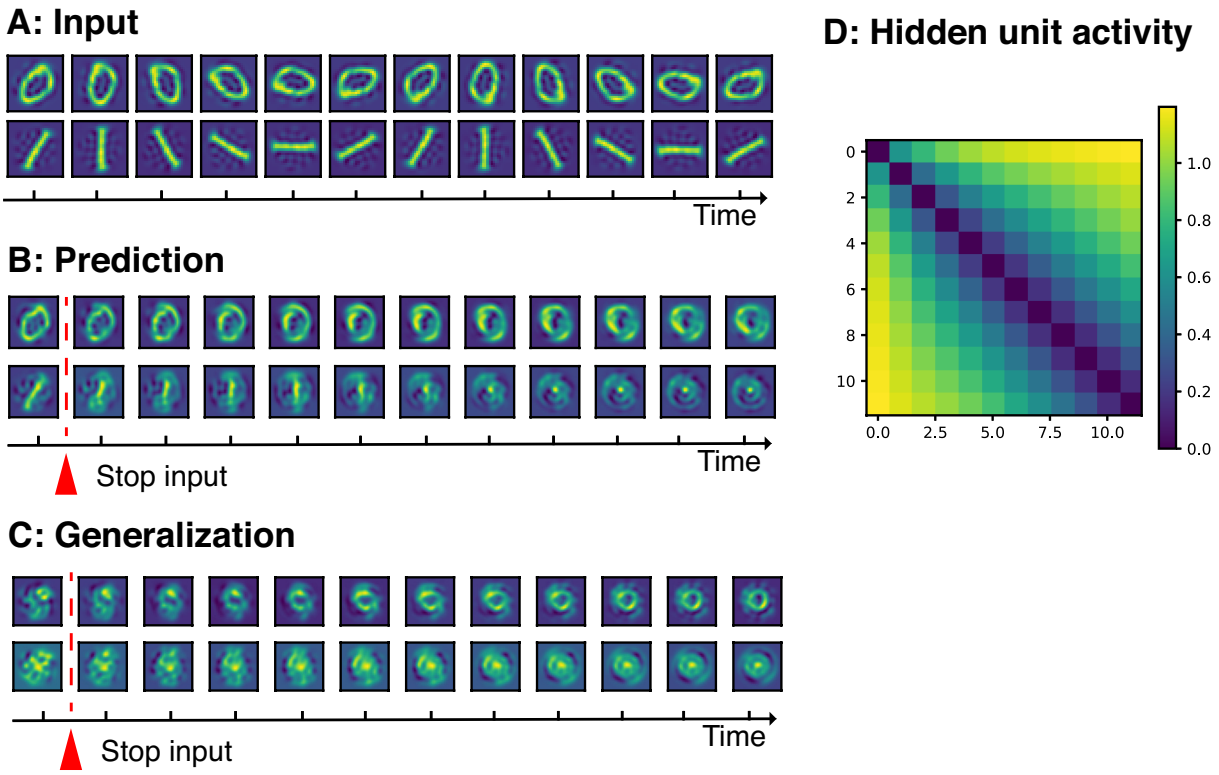

Figure S3: **Learning rotating sequences in RAE.** (A) Input as a sequence of MNIST digits rotating 30 degree counterclockwise. (B) RAE failed to continue the rotation operation when the input was stopped. (C) RAE failed to generalize the operation to unseen digits. (D) The Euclidean distance matrix of hidden unit activities between different time steps under the prediction scenario.

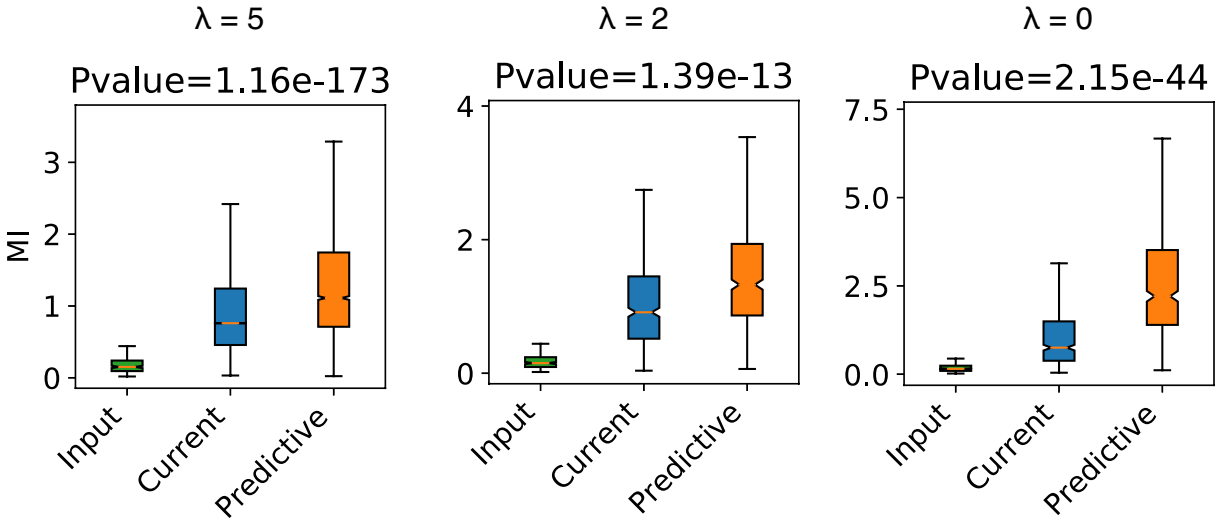

Figure S4: Mutual information of recurrent units trained using different regularization strength (λ). P value in the title refers to the comparison (t-test) between units from current network controls and predictive networks. For $\lambda = 5$, we trained 10 repetitive networks while for the other two, only one representative network was trained.