Sequential predictive learning is a unifying theory for hippocampal representation and replay

Daniel Levenstein[1,2], Aleksei Efremov[1,2], Roy Henha Eyono[2,3], Adrien Peyrache[1] *, Blake Richards[1,2,3,4] *

[1] Montreal Neurological Institute, Department of Neurology and Neurosurgery, McGill University

[2] Mila, Montreal, QC, Canada

[3] School of Computer Science, McGill University, Montreal, QC, Canada

[4] Learning in Machines and Brains Program, CIFAR, Toronto, ON, Canada

* These authors contributed equally to this work

Abstract

The mammalian hippocampus contains a cognitive map that represents an animal's position in the environment [1] and generates offline "replay" [2,3] for the purposes of recall [4], planning [5,6], and forming long term memories [7]. Recently, it's been found that artificial neural networks trained to predict sensory inputs develop spatially tuned cells [8], aligning with predictive theories of hippocampal function [9–11]. However, whether predictive learning can also account for the ability to produce offline replay is unknown. Here, we find that spatially-tuned cells, which robustly emerge from all forms of predictive learning, do not guarantee the presence of a cognitive map with the ability to generate replay. Offline simulations only emerged in networks that used recurrent connections and head-direction information to predict multi-step observation sequences, which promoted the formation of a continuous attractor reflecting the geometry of the environment. These offline trajectories were able to show wake-like statistics, autonomously replay recently experienced locations, and could be directed by a virtual head direction signal. Further, we found that networks trained to make cyclical predictions of future observation sequences were able to rapidly learn a cognitive map and produced sweeping representations of future positions reminiscent of hippocampal theta sweeps [12]. These results demonstrate how hippocampal-like representation and replay can emerge in neural networks engaged in predictive learning, and suggest that hippocampal theta sequences reflect a circuit that implements a data-efficient algorithm for sequential predictive learning. Together, this framework provides a unifying theory for hippocampal functions and hippocampal-inspired approaches to artificial intelligence.

Main

The mammalian hippocampus has been implicated in seemingly disparate cognitive processes including navigation [1,13–15], memory [16–19], planning [5,20,21], and imagination [22,23]. This functional diversity appears to rely on the interplay between two distinct modes of operation [24]. The first is an input-driven "online" mode during active behavior in the environment [25]. In the online mode, neural activity shows prominent theta rhythms [25], spatially tuned cells represent an animal's position [1], and population activity lies on a low-dimensional manifold reflecting environmental and task structure [26–29] (**Figure 1a**). The second is an "offline" mode during periods of behavioral quiescence and sleep [25]. In the offline mode, irregular neural activity shows prominent sharp-wave ripples [25,30] that "replay" realistic trajectories through the environment [31], including sequences of positions from previous experiences [3] and generative trajectories of untaken paths [6,32,33]. Importantly, replay is believed to be internally generated in the hippocampus [30,34–36], but may be influenced by its inputs [35,37], especially the head direction system which conveys a coherent but randomly drifting signal during sleep [38] (**Figure 1b**). However, a unified account of these modes of operation is still lacking. In particular, we lack a model for how a recurrent network of neurons like the hippocampus can learn a spatial representation from sensory information during the online mode and produce offline activity that meets three key desiderata: (1) An intrinsically-generated coherent representation of spatial position [39]; (2) Generation of sequences of positions that produce plausible trajectories through the environment [3,33]; and (3) Output of associated sensory inputs from learned associations at those positions [40] (**Figure 1c**).

A classic theory to explain hippocampal representation and replay is that the hippocampus implements a continuous attractor neural network (CANN, [41–43]). CANNs represent space using an attractive neural manifold [44] that maintains a localized bump of activity even in the absence of input. Indeed, CANN models can account for a number of experimental observations during both online and offline activity. These include: 1) the presence of spatially-tuned cells [1], 2) population activity that lies on a low-dimensional manifold [26–28], and 3) internally-generated offline trajectories that can range from diffusive motion [45] to memorized trajectories [46]. However, traditional CANN models rely on the pre-assignment of neurons to spatial locations or hand-tuned recurrent connectivity [41,42,47], and models that learn CANNs rely on signals with pre-existing tuning for allocentric space [48–50]. Further, CANNs cannot account for two other facets of hippocampal function. The first is the association of viewpoint-dependent sensory information with a viewpoint-independent representation of space. The second is the predominance of weakly tuned cells which are critical for the low dimensional structure of population activity [28,51] and the distributed hippocampal representation of space [52,53]. Thus, it remains unknown how a network can learn hippocampal-like representations and replay purely from egocentric sensory inputs such as images.

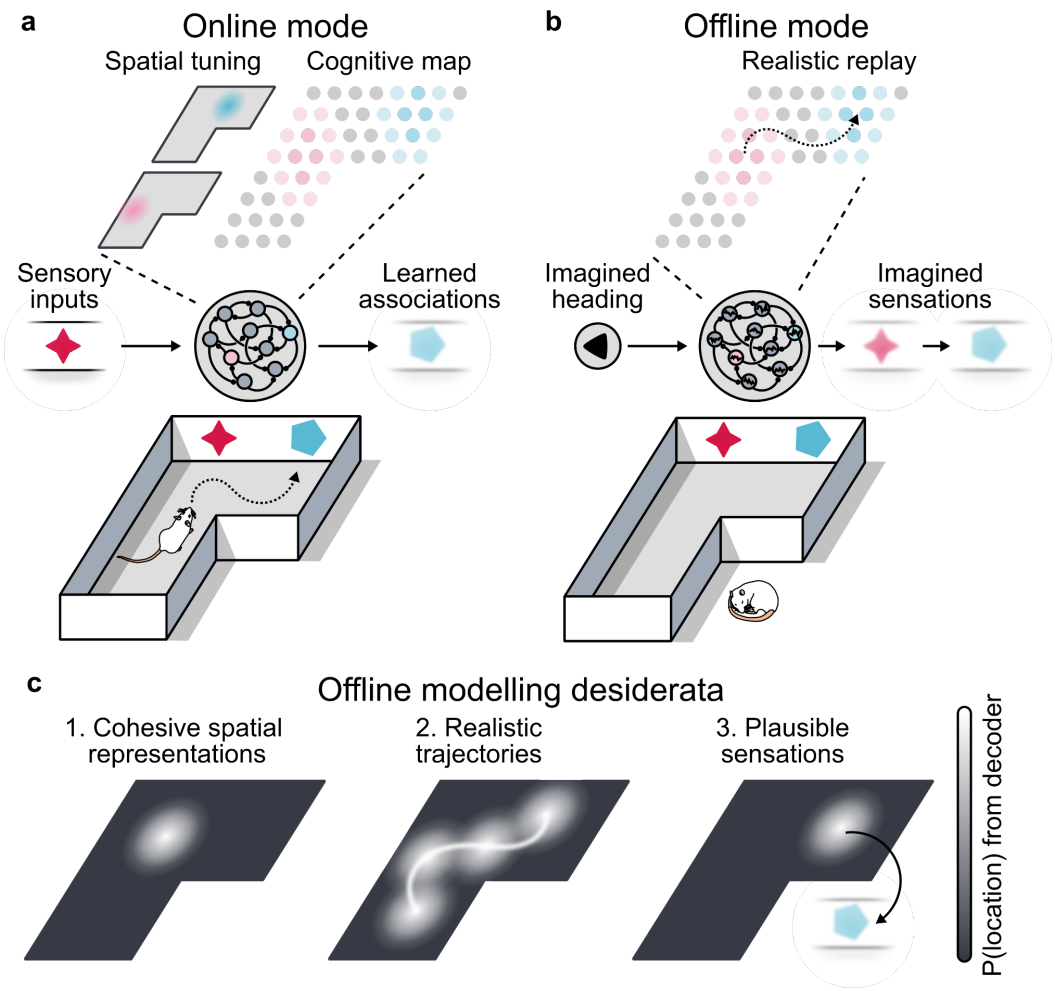

Figure 1: Hippocampal representation and replay.

a: During active behavior in the environment, the hippocampus is in an online mode of activity. Based on sensory inputs, it contains spatially tuned cells (e.g. two cells active in red and blue areas, respectively) and a cognitive map representation of the environment, which it uses to simulate the immediate consequences of future behavior and output learned associations to the rest of the brain. **b:** During consummatory and quiescent behavior, the hippocampus is in an offline mode of activity. It generates simulated "replay" trajectories, which project learned associations to the rest of the brain, possibly under the influence of an imagined head direction signal. **c:** Replay trajectories are sequences of internally-generated coherent representations of spatial locations, which follow realistic trajectories in the environment, and output learned associations of those locations to the rest of the brain.

One promising candidate is predictive learning. Computational models have demonstrated that learning to predict upcoming sensory inputs leads to the emergence of spatially-tuned units in artificial neural networks [8,54] and hippocampal place fields resemble predictive representations used in reinforcement learning [11,55]. In addition, the hippocampus encodes expectations about upcoming stimuli [56,57] and prediction errors [58,59]. Indeed, alternative theories of hippocampal function have proposed the hippocampus as a sequence prediction network, based on its anatomy [9,60] and observations that neuronal activity does not just encode an animals current location, but rather, "sweeps" in a manner that represents trajectories of possible future positions [12,61,62].

To determine the potential for predictive learning to unify the different aspects of hippocampal function, we trained recurrent neural networks (RNNs) to predict egocentric sensory inputs as an agent moved through a simulated environment (**Figure 2a**, Methods). This allowed us to compare the distributed representations of different predictive models (**Figure S1**) in a setting in which agents received spatially ambiguous sensory input. It also allowed us to study the models' offline generation capabilities in the absence of sensory input. We found that an attractor-based cognitive map capable of generating offline replay could emerge from predictive learning, but only when networks used recurrent connections to predict multi-step sequences of sensory input using an orienting (head direction) signal. In sum, sequential predictive learning can account for both online representations and offline replay in the hippocampus. As such, sequential predictive learning is a candidate theory to unify three views of the hippocampus: 1) the hippocampus is a predictive map, 2) the hippocampus is a CANN, and 3) the hippocampus is a sequence generator.

Next-step predictive learning produces spatially tuned cells but not replay

We simulated an agent in a "grid world" environment with visual cues consisting of different colors and patterns of floor tiles (**Figure 2a**). During a "wake" epoch, the agent took random actions that allowed for thorough exploration of the environment (Methods). At each location the agent received a 7×7 color image corresponding to the egocentric view of the floor and walls immediately in front of it (**Figure 2b**, **Supplemental Movie**). Critically, the inputs to this network were both spatially ambiguous and redundant (**Figure 2c, S2**) – individual sensory units had very little spatial tuning, multiple locations in the environment had the same sensory input, and the same position could have different sensory input depending on the agent's head direction. The sequence of actions and visual inputs from each epoch were used to train a RNN to predict the next visual input at each timestep using backpropagation through time (**Figure 2a, b, Figure S3a,** Methods).

After training, the network contained an allocentric representation of space (**Figure 2c,d**). The agent's location was accurately decodable from the predictive RNN units via a linear decoder (**Figure 2b, d)**, Methods), a large part of the variance of the activity of many RNN units' activity could be explained by the agent's position (141±15/600 units with >50% variance explained by spatial position), and they developed spatial tuning curves that carried significantly more spatial information than was observed in the visual inputs or untrained networks (**Figure 2c**, Methods). Like cells in the hippocampus [63], units in the network had a skewed distribution of spatial information (**Figure S3b**), with a large number of weakly tuned cells and a small number of cells with strong/reliable spatial tuning.

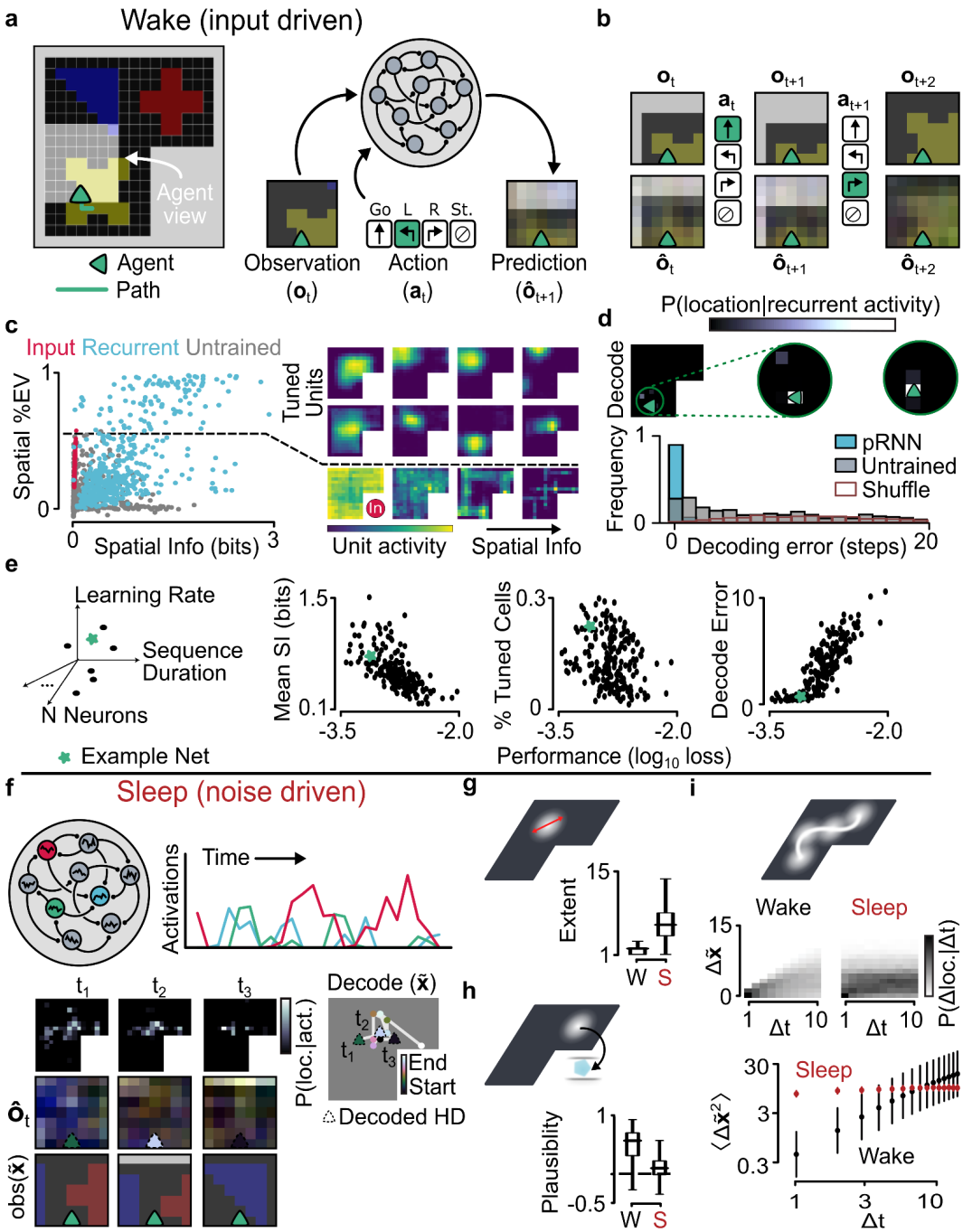

Figure 2: Next-step predictive learning produces spatially tuned cells but not replay.

a: (Left) The agent moves around the gridworld environment with a visual field corresponding to the 7x7 grid of tiles in front of it. Gray tiles represent impassable walls, while colored floor tiles provide visual cues. (Right) a predictive RNN (pRNN) is trained to use the current observation and action to predict the observation in the next timestep. **b:** Example sequence of observations and predicted observations from a trained network. **c:** (Left) Explained variance by spatial position and spatial information of all units in the example recurrent network, input units, and units from an untrained random network. (Right) spatial tuning curves from a selection of example tuned units (units with >50% variance explained by spatial position) with unimodal place fields, untuned units, and an example input unit. **d:** (Top) Decoded position from a linear decoder on hidden unit activity for the example three timesteps in B. (Bottom) distribution of decoding error over a 1000-timestep trajectory in the environment, for the example pRNN and untrained network, compared to a position-shuffled control trajectory. **e:** Spatial representation (mean spatial information, percentage of tuned cells, linear decoding

error) in a population of pRNNs with different hyperparameter values. Teal star reflects the example network in b-d. **f:** Sleep-like activity is generated in the network by removing the sensory input and driving each unit with noise. This activity is decoded using the linear decoder from d. Outputs from the network reflect the predicted "observations" associated with offline activity. **g**: The distribution of the decoded posterior's spatial extent (a measure of how coherently the network is representing a single position) during wake and sleep (combined over 15 sleep epochs). Box indicates Q1, median, Q3; whiskers indicate data range. **h**: The distribution of plausibility of the predicted visual output from the network during wake and sleep. Plausibility is measured as the correlation between the visual output and true visual field at the decoded position and head direction. Box indicates Q1, median, Q3. Whiskers indicate data range. **i**: Spatial transition statistics of the decoded trajectories during sleep and wake. While the variance of agent's position during wake increased with time (by construction), the decoded position during sleep incoherently shifted from location to location, irrespective of the position at the immediately preceding time step. Error bars indicate standard deviation over 15 sleep epochs.

The emergence of spatially tuned units was a robust property of predictive learning in a recurrent network. The trained RNNs had units with more spatial information than a random (untrained) RNN[64] (**Figure 2c,d**) or identically trained feedforward networks (**Figure S4**), and the emergence of spatially tuned cells did not rely on the geometry of the environment (**Figure S5**), or the use of a discrete environment and action space (**Figure S6**). Because the emergent properties of neural network models can be sensitive to specific hyperparameter choices [65], we trained a population of 250 predictive networks, each with a different hyperparameter setting (seed, learning rate, sequence duration, back-propagation time-window, number of neurons, and neural time-scale; **FIgure S7**). We found that the emergence of a spatial representation was strongly correlated with predictive performance across this population of RNNs (**Figure 2e**) and, critically, no network learned to predict well without developing spatially tuned cells. These results demonstrate that allocentric spatial tuning is a natural and robust consequence of learning to predict egocentric sensory information in recurrent neural networks.

We next tested if the predictive RNNs were able to generate plausible simulations during a sleep-like state. Specifically, we removed the sensory and action inputs and increased the amount of noise above that used during training (uncorrelated Gaussain noise, see Methods), such that network dynamics was predominantly governed by recurrent connectivity (**Figure S8**). To identify the locations represented by the network during sleep, we used the decoder trained to decode position from the RNNs' activity during wake (**Figure 2f**). We found that the network did not represent a spatially localized position (**Figure 2g**), the sensory "predictions " produced by offline activity did not correspond to the sensory input from the decoded viewpoint (i.e. position and head direction, **Figure 2h**), and that trajectories of represented locations did not conform to environmental statistics (**Figure 2I**). Instead, the trajectory frequently jumped around space and only visited a small number of disconnected locations in the environment, which were the same from trial to trial (**Figure S8**). This was in stark contrast to trajectories during wake, which were restricted to smooth transitions through the environment. The inability to produce replay was not a result of specific hyperparameter choices, as the networks were unable to produce reasonable simulations regardless of the network hyperparameters or level of noise (**Figure S8**). Thus, learning to predict the next image robustly produced spatial tuning but did not produce the ability to intrinsically generate replay.

A continuous attractor-based representation emerges from sequential predictive learning

In examining the population activity of the predictive RNNs, we observed that, despite the presence of spatially tuned units, activity in the RNN did not resemble a CANN (**Figure 3a,b**). That is, population activity during wake did not lie on a manifold that mirrored the spatial layout of the environment, and the activity during sleep converged to a single off-manifold fixed point. To quantify these observations, we developed two metrics: first, a spatial representational similarity analysis (sRSA) that measured the correlation between distance in neural space and distance in the environment (Methods); and second, a sleep-wake distance (S-W Dist) that measured the distance in neural space between activity during sleep and the wake manifold (Methods). We observed that the sRSA scores for the predictive RNNs were significantly lower than those for the CANNs (**Figure 3a,b**), while their S-W Dist was significantly higher, indicating that population activity lied outside the wake manifold.

We thus wondered whether other forms of predictive learning might produce an attractor-based cognitive map of the environment. Recent findings indicate that recurrent autoencoders can also contain spatially-tuned cells [66], and masked predictive learning, an approach in which an autoencoder is trained to predict inputs at held-out timesteps, has been found to produce good internal models in e.g. vision and language tasks [67]. Under a masked prediction paradigm, visual inputs can be masked for multiple time-steps (**Figure 3c, S1**), which we hypothesized would encourage the recurrent connectivity to maintain a coherent representation of the agent's current state and capture transitions through space. Indeed, we found that multi-step, or sequential, predictive learning led to the emergence of a neural manifold that reflected the layout of the environment and constrained neural activity during sleep (**Figure 3d**).

Interestingly, despite the ability of all masked networks to solve the prediction task and develop spatially tuned cells, only sequential networks trained with speed and head direction (HD) formed a cognitive map (**Figure 3e, f**). Specifically, an attractive manifold (low SW Dist) only emerged in networks trained to predict at least two time-steps of masked observations (**Figure 3e**) and the mapping to space (high sRSA) didn't emerge in one-step masked networks or in sequential networks receiving action identity or speed and angular velocity (**Figure 3e, Figure S9**). Furthermore, sequential predictive networks trained with speed and HD information were the only networks in which their spatial representation more closely reflected Euclidean distance than the number of steps between locations (**Figure 3f**). These results were robust across networks with different hyperparameters (**Figure 3g, S10**), consistent across environments with different geometries (**Figure 3h**), and in a continuous environment with realistic behavioral trajectories[68] (**Figure 3i, Figure S11**).

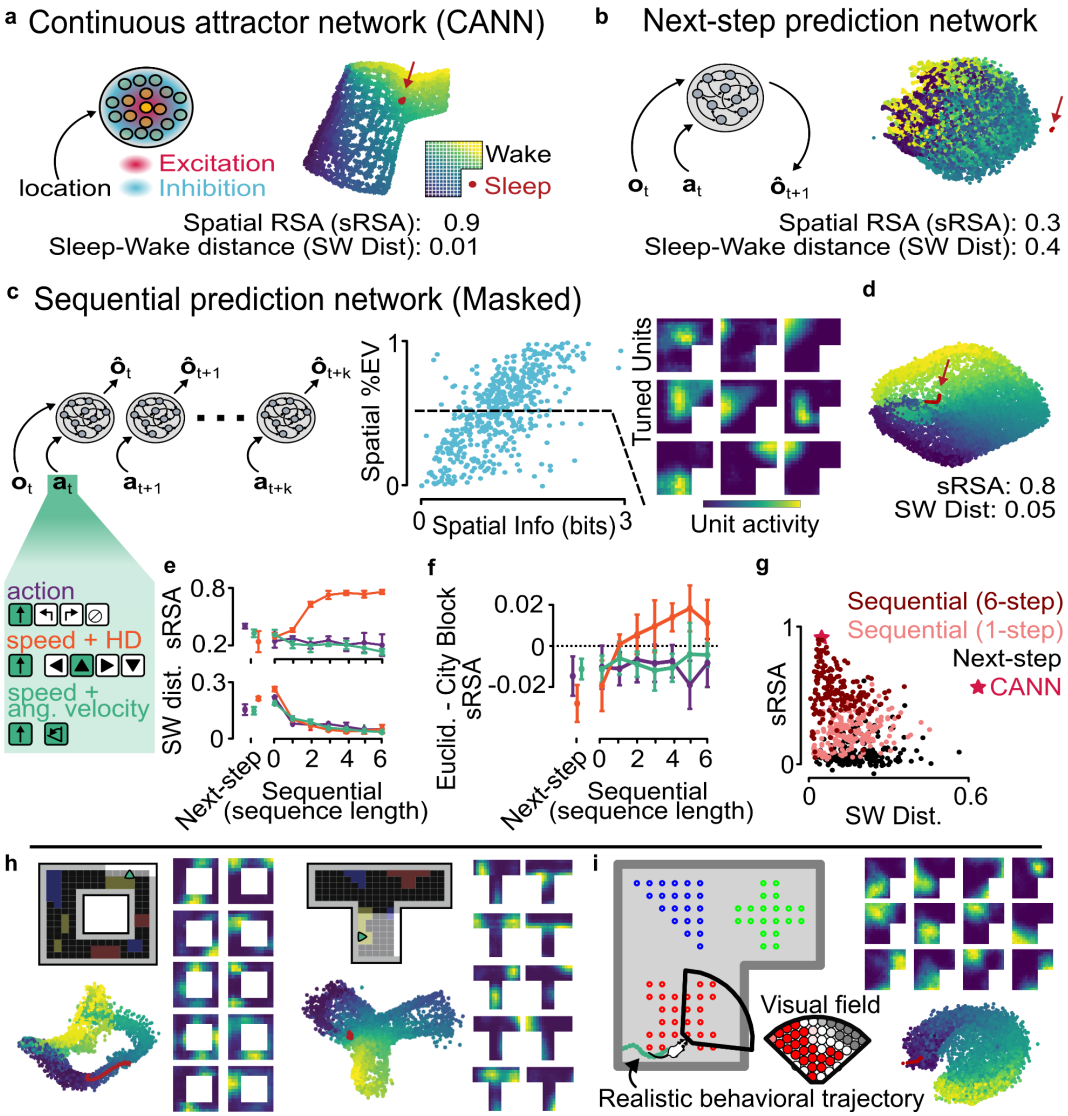

Figure 3: A continuous attractor-based representation emerges from sequential predictive learning.

a: Isomap visualization of population activity activity in a continuous attractor network (CANN), in which units assigned to represent nearby positions are connected with recurrent excitation and those representing distant positions are mutually inhibiting. Each point represents a population activity vector at a point in time, colored by the position of the agent (wake), with sleep timepoints in red. Spatial representational similarity analysis (sRSA) measures the correlation between spatial and neural activity distances, while Sleep-Wake distance (SW Dist) measures the neural distance between sleep and wake activity in the network. **b:** Similar to A, but for the Next-Step pRNN from Figure 1. **c:** In the masked sequential predictive network, predictions ($\hat{o}$) are made for a k-length sequence of held-out sensory observations at the current timestep (see also Methods and Figure S1). (Right) Spatial tuning properties of recurrent units from an example masked predictive learning with a sequence length of k=5, trained with actions encoded as speed and head direction. **d:** Similar to A, for the k=5 masked network from panel C. **e:** sRSA and SW Dist for next-step pRNNs and masked pRNNs with k=0 (autoencoder), k=1 (single-step), and k=2-6 (sequential), each trained with one hot (action), speed+HD, and speed+AV encoding of the action input, $\mathbf{a}_t$. Mean and standard deviation calculated over 9 seeds. **f:** Difference between sRSA calculated using Euclidean vs city-block distance, for all networks in **e**. Higher Euclidean sRSA indicates a euclidean, rather than behavioral-step-based map of space. **g:** sRSA and SW Dist for a hyperparameter-matched population of Next-step, single-step, and 5-step masked pRNNs. **h:** Neural manifolds and example spatial tuning curves for a 5-step masked pRNN trained in environments with different geometries. **i:** Neural manifolds and example spatial tuning curves for a 5-step masked pRNN trained in a continuous environment [68].

While the use of layer normalization, dropout, and noise injection (see Methods) improved the stability of training and offline activity, and spatial tuning in the next-step RNNs (**Figure S12**), it did not promote the formation of a continuous attractor in the next step networks, and their presence wasn't necessary for its emergence in sequential RNNs (**Figure S12**).

How can the next-step predictive RNN have spatially tuned representations but not an attractive neural manifold that maps the spatial layout of the environment? This apparent contradiction was a result of the network's representation of space being superseded by strong representations of action identity and visual inputs (**Figure S13**), which were only weakly represented in the sequential predictive network (**Figure S13**). Interestingly, increasing the degree of visual ambiguity in the environment reduced the spatial representation in next step, single step, and action identity encoded networks, but not in sequential net with speed+HD action encoding (**Figure S14**), further supporting the idea that sequential predictive learning with speed and head direction information encourages a cognitive map solution. The appearance of a cognitive map manifold could be attributed to the prevalence of cells that have more of their variance explained by spatial position. Despite having tuning curves with the same amount of spatial information, units in the sequential prediction networks had more of the variance in their activity explained by spatial position (**Figure S15**), and more of these spatially-tuned cells had unimodal tuning curves for single spatial locations (**Figure S15**). Across the hyperparameter population, sRSA was correlated with the fraction of tuned cells rather than mean SI (**Figure S15**).

Sequential predictive RNNs generate offline simulations that can autonomously replay recent locations or follow a head direction query

In contrast to the next-step predictive networks (**Figure 2**), offline activity in the sequential predictive RNNs maintained a coherent representation of position in the absence of sensory input. When the network was driven only by internal noise, the decoded position at each timestep was spatially concentrated (**Figure 4a, b**), and made local transitions around a single location which was different for each sleep epoch (**Figure 4c**, **Figure S16**).

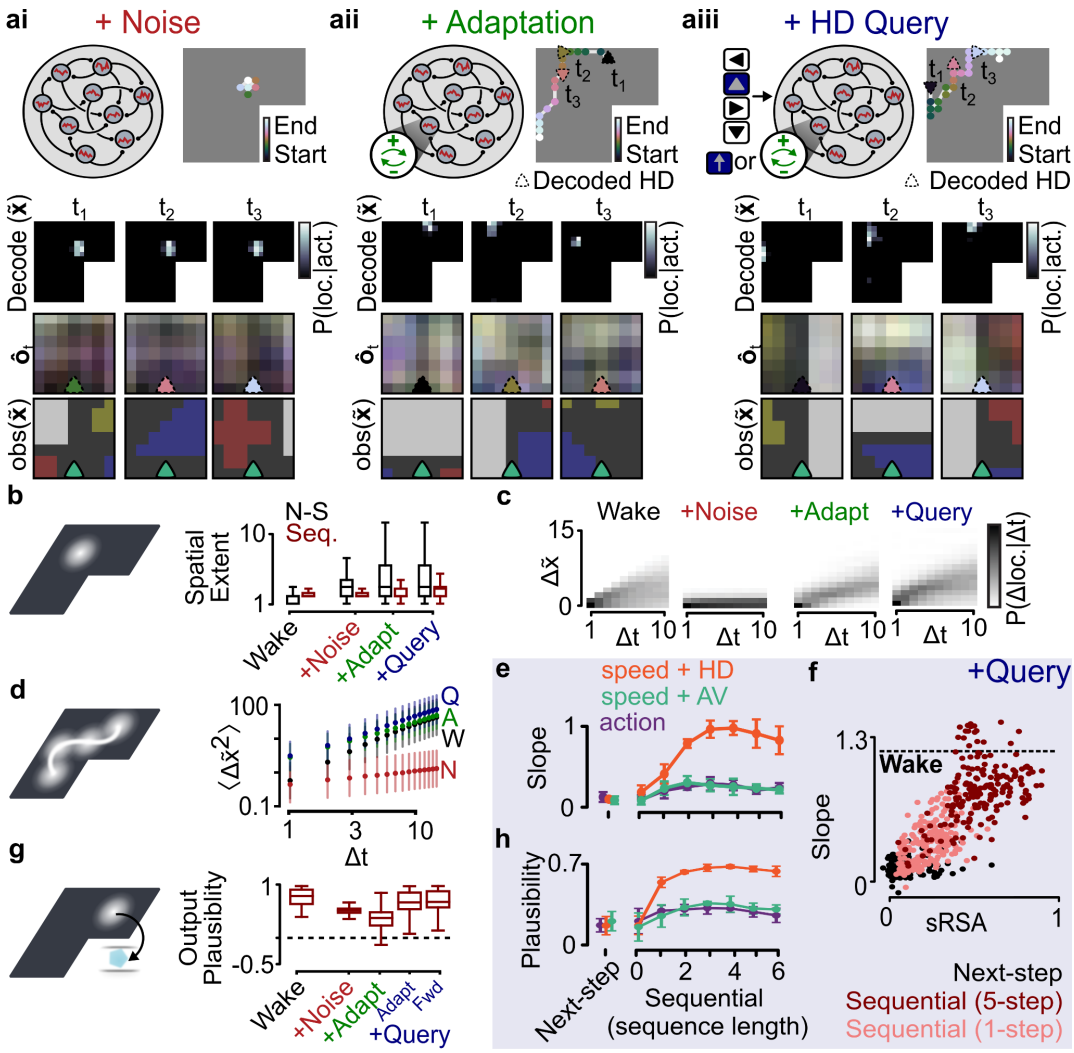

Figure 4: Sequential predictive RNNs generate offline trajectories that can follow a head direction query.

a: Example subset of a sleep trajectory from the example masked pRNN with a sequence length of k=5, including decoded posterior, observation output, and true observation from the decoded viewpoint. Similar to Figure 2f. Offline activity (sleep) is generated in the absence of sensory input using internal noise (**i**), the addition of a neuron-autonomous negative feedback (adaptation, **ii**), and a virtual head direction query with either adaptation or virtual speed query (**iii**). **b:** Distribution of spatial extent of the decoded posterior for offline activity in the next-step and sequential (k=5) masked pRNNs, with wake and each sleep mechanism. Compare to Figure 2g. **c,d:** Spatial diffusion statistics of the decoded trajectory during wake and each sleep mechanism. Compare to Figure 2i. **e:** log(<$\Delta\mathbf{x}^2$>)/log(Δt) slope for query-generated offline activity in next-step pRNNs and masked pRNNs with k=0 (autoencoder), k=1 (single-step), and k=2-6 (sequential), each trained with one hot (action), speed and head direction (speed+HD), or speed and angular velocity (speed+AV) encoding of the action input. Mean and standard deviation calculated over 9 seeds. **f:** log(<$\Delta\mathbf{x}^2$>)/log(Δt) slope as a function of sRSA for a hyperparameter-matched population of next-step, single-step, and 5-step masked pRNNs. **g:** Distribution of output plausibility of the decoded posterior for offline activity in the sequential (k=5) masked pRNNs, with wake and each sleep mechanism. Compare to Figure 2h. **h:** Output plausibility for query-generated offline activity in next-step pRNNs and masked pRNNs with k=0 (autoencoder), k=1 (single-step), and k=2-6 (sequential), each trained with one hot (action), speed+HD, or speed+AV encoding of the action input.

Offline activity in the hippocampus is not restricted to single locations, though, shows extended paths that travel through the environment and often recapitulate salient or rewarded positions from recent exploration [31]. A common way to produce trajectories in CANNs and other hippocampal models is the addition of adaptation [45,69–71] (**Figure 4a**), i.e. slow negative feedback on each neurons' activity which is thought to play a key role in producing the dynamics of offline activity in the hippocampus [72]. We found that adding an adaptive variable to our RNN units resulted in extended sleep trajectories (**Figure 4c, S16, S17**), with similar statistics to wake trajectories (**Figure 4d**). Remarkedly, in addition to transitions between adjacent spatial locations, offline trajectories often made diagonal and 2-step transitions (**Figure S16**), neither of which were possible in the training data. Thus, the network can autonomously produce generative simulations that reflect the structure of the environment, but are not strictly limited to the observed state transitions during wake. Increasing the learning rate during a waking trial caused the network to shift from generating plausible but random trajectories, to recapitulating locations visited within the trial during the subsequent sleep epoch (**Figure S18**).

In addition to replaying recent locations, offline hippocampal activity could be used for directed planning [5,6] and imagination [23]. Motivated by observations that the neural circuits that represent head direction are spontaneously active during offline activity in the brain [73], we hypothesized that the same input that indicated the agent's actions during wake could direct trajectories during sleep (**Figure 4a**). Indeed, like adaptation, the addition of an action "query" produced wake-like transition statistics (**Figure 4c,d, S17**), by influencing the direction of replay transitions (**Figure S16**) . This behavior was only seen in the sequential predictive networks trained with speed and head direction information (**Figure 4e**), and was related to their ability to form a cognitive map as measured by sRSA and S-W distance (**Figure 4f**). Interestingly, the ability to produce wake-like trajectories only relied on the head direction component of the action signal, while either fictitious speed or adaptation could modulate the velocity of this movement (**Figure S16**). In addition, the head direction query improved the plausibility of visual simulations, producing output that mimicked the associated sensory inputs at the replayed locations during sleep (**Figure 4g**,**h, S17**). Together, these results reveal that sequential predictive learning produces networks containing a cognitive map that achieves the three desiderata for offline activity (**Figure 1c**), namely it: (1) maintains a coherent representation in the absence of input, (2) produces extended trajectories that can replay recent experiences or directed via a head direction query, and (3) can recapitulate the learned sensory associations at the represented positions.

Rollout-based sequential predictive learning produces theta sweep representational dynamics and rapid learning

While the learning procedure above recapitulates several experimental observations, it requires a large amount of sensory "data" – more precisely, a map is only learned after about 2000 trials of 500 steps each (**Figure S20**). In contrast, a neural manifold emerges in the hippocampus after only a few days of exposure

to a novel environment [26,28] and the hippocampus is able to rapidly form associations between positions and novel sensory information [74]. Further, unlike the hippocampus [56,57] , masked networks predict present rather than future sensory observations. In self-supervised [75] and reinforcement learning [76], sequential prediction is often learned via "rollouts": prediction of multiple future timesteps, which results in higher data efficiency [77]. Therefore, we modified the learning procedure to use a rollout-based approach in which the RNN predicts multi-step sequences of future observations every timestep before it receives the next time-step's sensory inputs (**Figure 5a, S1**).

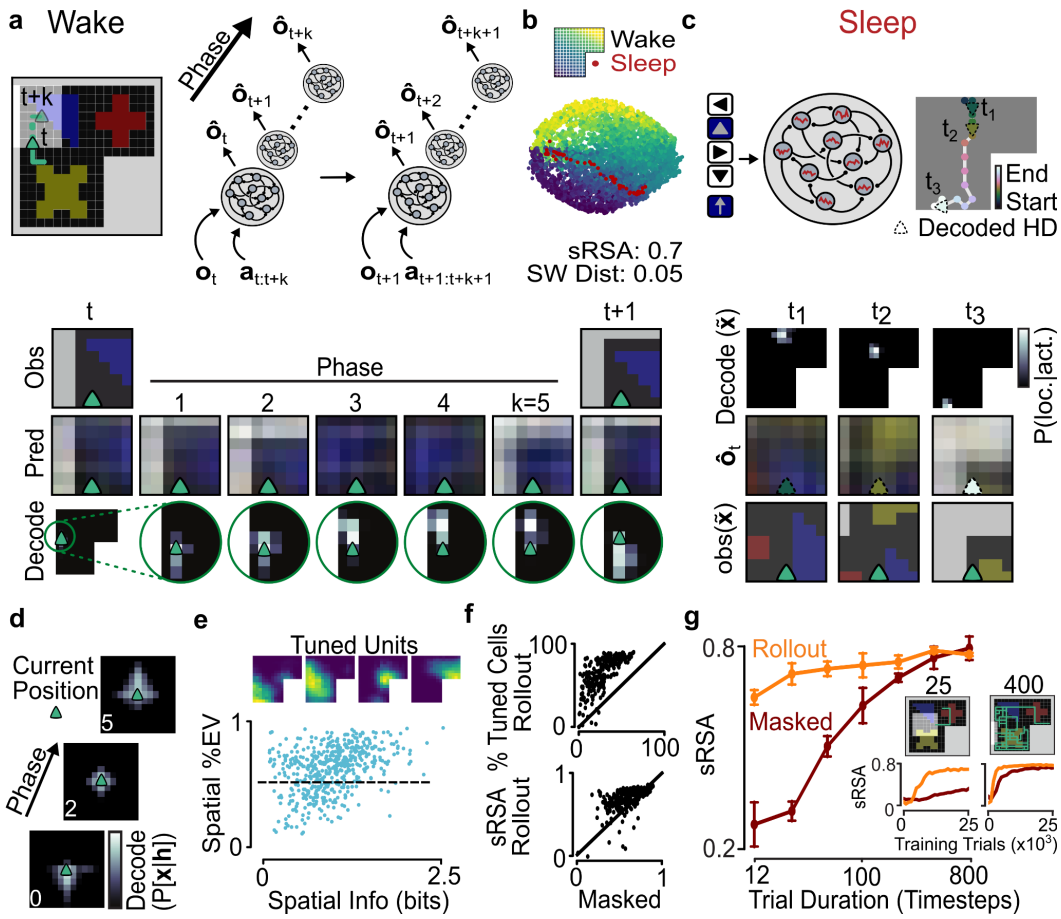

Figure 5: Rollout-based sequential predictive learning produces theta sweep representational dynamics and rapid learning
a: In the rollout-based predictive network, predictions ($\hat{o}$) are made for a k-length sequence of future sensory observations, based on the current sensory observation and the next k actions. **b:** Isomap visualization of the neural activity manifold for an example rollout-based network with a sequence duration of $k = 5$. Similar to Figure 3a. **c:** Example query-generated offline activity in the rollout-based sequential pRNN. Similar to Figure 4Aiii. **d:** Mean decoded posterior relative to the agent's current position for activity at early, middle, and late phases of the prediction cycle. **e:** Example units, and spatial tuning properties of all network cells in the rollout-based sequential pRNN. **f:** Percentage of tuned cells (top) and sRSA (bottom) for a population of hyperparameter-matched rollout-based and masked pRNNs with $k = 5$. **g:** sRSA for rollout-based and masked sequential networks ($k = 5$), trained using trials of different durations. Training for all networks was run for 5x10^4 trials. Mean and standard deviation calculated over 9 seeds. (Inset) sRSA over the first 2.5x10^4 trials for networks trained with 25 and 400-step trials.

Like sequential predictive networks using the masked approach, this rollout-based predictive RNN learned a continuous attractor map (**Figure 5b**), which maintained a coherent spatial representation during sleep and produced wake-like trajectories with plausible sensory associations (**Figure 5c**). Interestingly, when the decoder was applied to successive phases of the rollout, the position represented by the network would "sweep" from immediately behind the current agent's current position to a few steps ahead of it (**Figure 5a,d**). This cyclical representation was strikingly similar to hippocampal theta sweeps: spiking sequences representing locations that project ahead of the animal every ~120 ms, coordinated by an 8Hz theta oscillation[78]. To learn the attractor map, the network had to be provided with the prospective sequence of future speed and head direction (**Figure S19)**, akin to having planned its actions in advance. However, once the map was learned, the network was able to simulate possible but untaken future trajectories during rollouts, when provided with a hypothetical action sequence (**Figure S19**).

Rollout learning drastically improved learning performance and the emergence of a cognitive map. Rollout networks had more spatially-tuned cells (**Figure 5e-f**), developed a higher-sRSA representation of space than their masked counterparts (**Figure 5f, Figure S20**), and were better at sensory prediction (**Figure S20**). Further, this approach was able to rapidly form new sensory associations when a novel object was introduced in the environment, which persisted in the network's output even when the object was removed **(Figure S21**). The reason for this improvement is that the network was more data efficient - it was able to form a continuous attractor map with fewer trials of shorter sequence duration (**Figure 5g, Figure S20**).

Altogether, we found that rollout-based sequential predictive learning improved the speed and robustness of learning continuous attractor maps, and replicated the experimentally observed patterns of theta sweep "lookaheads". Hence, our findings suggest that theta sequences are a signature of sequential predictive learning in the hippocampus.

Discussion

In this study, we have used a RNN model to demonstrate that sequential predictive learning provides a unifying theory for hippocampal representation and replay. In line with previous results [8,54,66], we found that learning to predict sensory inputs robustly leads to the development of spatially tuned cells in recurrent neural networks. However, we found that the presence of spatially tuned cells is not *sufficient* to guarantee a cognitive map with the ability to generate offline replay. Nonetheless, a continuous attractor manifold consistently emerged in networks trained to predict multi-step sequences of sensory inputs using speed and head direction information, which allowed the networks to simulate plausible trajectories offline. This offline "replay" activity could recapitulate trial positions, generate novel trajectories, or be directed by a head direction query. We found that the efficiency of cognitive map formation was increased by making cyclical predictions of future sensory inputs, which produced representations that mimicked theta sweeps observed

in the hippocampus. These results show that multi-step sequence prediction is a promising unified account of hippocampal operations.

Our work builds upon previous studies that have explored the relationship between predictive models and spatial representations in the hippocampal-entorhinal system. This past work has shown that: (1) Aspects of spatial tuning in the hippocampus are well-explained by predictive representations [11,55], which can be learned from spatially tuned inputs with predictive Hebbian learning [79], TD learning [80], or spike-timing dependent plasticity [80,81]; (2) Learning to predict memory embeddings from actions can link allocentric and egocentric representations [82]; (3) Prediction can link spatially tuned inputs and relational structures [83]; (4) Cloned hidden-Markov models that use discrete states to predict observations from actions can recapitulate many of the features of the hippocampus and support offline evaluations [84,85]; (5) Engaging in path integration (i.e. predicting spatial locations from sequences of actions) leads to the emergence of grid cells and continuous attractor dynamics [48,50]; (6) Training recurrent networks to predict hippocampal spiking data or linear place cell sequences can reproduce various features of hippocampal activity, including spike cross-correlations and sequence replay [86]. Our work demonstrates that these findings can be brought together under the umbrella of sequential predictive learning of sensory data in RNNs, and applied to ego-centric, high-dimensional, continuous sensory inputs.

More broadly, sequential predictive learning unifies three distinct views of hippocampal operations. In addition to connecting views that the hippocampus is a predictive map [11] with those that the hippocampus implements a CANN [43], sequential predictive learning aligns with views that the hippocampus is a sequence generator [10,60,85,87]. These views are based on extensive physiological observations of theta sequences [78], which are tied to behavioral steps such as whisking [88], footsteps [89], sniffing [90], or saccades [91], and are essential for subsequent replay [92–94]. They are further supported by recent models demonstrating that associating observations with sequences of discrete states can reproduce many physiological phenomena associated with the hippocampus [85], and classic models demonstrating that sequence learning in neural networks can reproduce many of its psychological functions [60].

One of the most surprising results in our study was that spatially tuned cells are not a guarantee of a low-dimensional neural manifold with topological correspondence to the environment, or cognitive map. Supporting this distinction between spatial tuning and a cognitive map, experimental work has shown that, while place cells appear almost instantaneously in novel environments [95], a cognitive map only emerges after multiple days of exposure [26,28] and relies on weakly tuned cells [28,51]. Further, our observation that predicting sequences of sensory inputs was necessary for map formation and replay, but not spatial tuning, mirrors physiological results that place cells emerge in the hippocampus before sequential activity during development[96,97] and that perturbing its sequential structure disrupts its ability to produce replay, but not its associative code for space [94].

In addition to predicting sequences of sensory inputs, our results suggest that a head direction signal is critical to learn a cognitive map from viewpoint-dependent observations. While the inclusion of a head direction signal is, admittedly, allocentric, it has a striking resemblance to the inputs that convey an animal's head-direction and forward velocity to the hippocampal system [98]. Further, the head-direction system conveys a coherent signal during sleep which is disconnected from the animal's actual head direction [73] and shows increased activity immediately prior to hippocampal replay [99]. While adaptation alone was sufficient to produce replay trajectories in our model, a virtual head direction input was necessary to output associated sensory information at the replayed positions, and was able to influence the direction of replay trajectories. Together, these results predict that head direction is not necessary for forming place cells, but that is instrumental in the formation of a cognitive map, the production of replay that recapitulates learned associations for downstream processing [38], and can support the use of replay for directed planning and imagination. This critical role of the head direction system is an experimental prediction that could be tested in future work.

There are a number of limitations in this work that should be considered. First, we trained our networks from a random initial state, which is not an accurate reflection of learning in the hippocampus. There is substantial evidence showing that the hippocampus has pre-existing connectivity structures on which learning operates [100,101]. One prominent hypothesis is that these structures endow the hippocampus with a reservoir "library" of existing sequences [100,102,103], which can be associated with arbitrary sequences of sensory input. We hypothesize that such an initialization would facilitate rapid, and even single-shot, sequential predictive learning. Moreover, any adult animal will have ample experience with many environments, and this previous knowledge would be brought to bear on learning "novel" environments [104]. Future work could explore how sequential prediction behaves in networks with structured initializations and/or training on multiple continuously learned environments.

Second, while we were inspired by theta dynamics for the sequential learning algorithm, we did not concern ourselves here with the biological realism of the synaptic plasticity mechanism. However, the rollout-based predictive network can potentially be implemented by biologically-plausible plasticity mechanisms. In a biologically realistic network, behavioral time-scale plasticity driven by plateau potentials [105] could provide the eligibility traces required to maintain a record of past predictions and compare them with current observations in dendritic compartments with spatiotemporally-segregated inputs [106]. While these mechanisms have generally been studied in the context of feedforward inputs from CA3 to CA1 [107], they can also potentiate the recurrent synapses necessary for sequential prediction in CA3 [108].

Another limitation is that, unlike our model, the hippocampus does not receive direct visual inputs, but instead receives signals via the entorhinal cortex that are processed by multiple cortical structures.

Specifically, specialized neuronal populations such as grid and landmark cells in the medial entorhinal cortex provide highly organized and low-dimensional inputs to the hippocampus. It thus makes it possible to rapidly encode new environments and may be critical for learning more complex environments, such as the multi-sensory real world. Moreover, the neocortex itself may be engaged in predictive learning [109], and high-level predictive learning can result in the emergence of cells with cortical-like response properties in low-level circuits [110]. As hippocampal function can only be understood through its interactions with cortical and subcortical areas [111,112], it will be important to examine sequential predictive learning in deeper hierarchical circuits in follow-up research.

In conclusion, our study demonstrates the ability of sequential predictive learning to account for hippocampal representation during active exploration and replay during behavioral quiescence. This suggests that the hippocampus may be best understood as a future sequence prediction circuit.

Methods

Environment

Gridworld

The gridworld environment was developed using the minigrid package (https://minigrid.farama.org/, Chevalier-Boisvert et al 2023). An 18x18 L-shaped room (**Figure 1A**) was generated using walls and colored floor tiles, which each corresponded to a unique [R, G, B] color value. Walls ([0.6, 0.6, 0.6]) were impassible. Floor ([0.3, 0.3, 0.3]) and colored floor tiles ([0.45, 0.45, 0], [0, 0, 0.45], and [0.45, 0, 0]) were passable, and arranged in different shapes on the floor to serve as visual cues.

For each "wake" epoch, the agent was initialized at time $t = 0$ at a random position, $\boldsymbol{x}_0$ and head direction, $\boldsymbol{HD}_0$, in the environment. It then took a series of T actions, which were randomly selected from (move forward, rotate left, rotate right, stop) with probability 0.6, 0.15, 0.15, and 0.1, respectively. As the agent moved, the observation at time t corresponded to the 7x7x3 image of tiles immediately in front of it, with the agent's current position at the bottom center of the image with color [0.5, 0.3, 0.3] (**Figure 1B**). This image was then flattened, resulting in $T + 1$ observation vectors, $\boldsymbol{o}_t \in \mathbb{R}^{N_{obs}} = obs(\boldsymbol{x}_t, \boldsymbol{HD}_t)$, $N_{obs} = 147$, which started with the initial observation at $t = 0$ and ended with the observation following the final action, $\boldsymbol{a}_T$.

The actions, $\boldsymbol{a}_t$, were encoded in one of three different forms: 1) a *one-hot encoding* vector of length $N_{act} = 4$, in which the action at time t was represented by a 1 and the other actions were represented by a 0; 2) a *speed and angular velocity encoding* vector of length $N_{act} = 2$, in which the first element was 1 if the agent took the move forward action and 0 otherwise, and the second element was 1 if the agent took the rotate right action and -1 if the agent took the rotate left action; 3) a *speed and head direction encoding* vector of length $N_{act} = 5$, in which the first element was 1 if the agent took the move forward action and 0 otherwise, and the other 4 elements were a one-hot encoding of the agent's current head direction (North, South, East, West).

Rat in a Box

The continuous environment was developed with the RatInABox toolkit (https://github.com/RatInABox-Lab/RatInABox, George et al 2024). For comparability, the room was designed as close as possible to the one described above (**Figure 3i, S6a**): a 1x1m room of the same L-shaped form with three distinct classes of objects positioned to fit the locations of colored floor tiles in the Gridworld environment.

Agent's movements in the RatInABox are generated using two independent Ornstein-Uhlenbeck processes for rotational velocity and linear speed as described in the original paper. The agent was

initialized with the following parameters: time discretization (the amount of time a single step takes, *dt*): 0.1s; speed_coherence_time (time over which linear speed decoheres): 0.7s; speed_mean: 0.2m/s; rotational_velocity_coherence_time (time over which rotational speed decoheres): 0.08s; rotational_velocity_std: $\frac{2\pi}{3}$; thigmotaxis (tendency for agents to linger near walls [0 = not at all, 1 = max]): 0.2; wall_repel_distance: 0.1m.

The observations along the movement trajectories were collected using the Field of View (FoV) neurons feature. Each of these neurons has preferred distance from the agent and angle relative to agent's heading. If the object of interest is within proximity of this preferred location, the neuron outputs a value according to a double-gaussian model, with a maximum of 1. The agent was initialized with separate sets of FoV neurons for each of the three classes of objects and for the walls: $\{F_1, F_2, F_3, F_w\}$. Each set contained 48 cells that tiled a sector in front of the agent with a radius of 0.33m and the angle boundaries (-45°,45°) relative to the agent's head direction. Thus, each neuron for one class of object had corresponding neurons with the same preferred distance and angle for each other object class and for the walls. Note that having orthogonal representations for each type of object and for the walls is different from what was done in the Gridworld environment where the floor marks and the walls shared the RGB channels (for example, both red and yellow mark were present in the red channel, and the walls were represented in all three channels). To match the Gridworld RGB pattern, we transformed the outputs from FoV neurons into a 48x3 matrix where 48 is the number of neurons and 3 is for red, green, and blue channels. F_1 output was added to channel 1 (red), F_2 output was added to channel 3 (blue), F_3 output was added to channels 1 and 2 (green), and F_w output was multiplied by *0.39* and added to all three channels. This matrix was then flattened, resulting in observation vector o_t of size $N_{obs} = 144$. The actions a_t were encoded by a vector of length $N_{act} = 13$ with first element corresponding to the agent's linear speed, and the rest one-hot encoding its head direction that was binned into 12 sectors.

Predictive RNNs

The core of each predictive RNNs followed the same general architecture.

$$\boldsymbol{h}_t = \mathrm{NormReLU}(\boldsymbol{W}_{\boldsymbol{rec}}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{\boldsymbol{in}}\boldsymbol{o}_t + \boldsymbol{W}_{\boldsymbol{act}}\boldsymbol{a}_t) \quad (1)$$

$$\boldsymbol{y}_t = \mathrm{Sigmoid}(\boldsymbol{W}_{\boldsymbol{out}}\boldsymbol{h}_t) \quad (2)$$

With activation functions,

$$\mathrm{NormReLU}(\boldsymbol{x}) = \max\left(0, \quad \frac{\boldsymbol{x} - \mathrm{E}[\boldsymbol{x}]}{\mathrm{Std}[\boldsymbol{x}] + \epsilon} + \boldsymbol{b} + \boldsymbol{\mathcal{N}}(0, \eta)\right) \quad (3)$$

$$\mathrm{Sigmoid}(\boldsymbol{x}) = \frac{1}{(1 + e^{-\boldsymbol{x}})}$$

In which $\boldsymbol{h}_t \in \mathbb{R}^N$ is the state of the network at time t, $\boldsymbol{b} \in \mathbb{R}^N$ is a neuron-specific bias, $\boldsymbol{W_{rec}}$ is the (NxN) recurrent weight matrix, $\boldsymbol{W_{in}} \in \mathbb{R}^{N \times N_{obs}}$ and $\boldsymbol{W_{act}} \in \mathbb{R}^{N \times N_{act}}$ are the input weight matrices for observation and action, respectively, $\boldsymbol{W_{out}} \in \mathbb{R}^{N_{obs} \times N}$ is output weight matrix, $\epsilon = 10^{-4}$ is a small constant to prevent division by 0, and η is the standard deviation of uncorrelated noise injection to the network (Lim et al 2021, Krishna et al 2024). The first term in the second argument of equation 3 corresponds to layer normalization (Ba et al 2016), which was replaced with $\boldsymbol{x}$ for networks trained without layer normalization (**Figure S12**). For networks trained without noise injection (**Figure S12**), η was set to 0.

The three different pRNN architectures (Next Step, Masked, Rollout) were distinguished by the timesteps in which $\boldsymbol{o}_t$ was available to the network and the identity of the observation predicted by the output, $\boldsymbol{y}_t$, per the loss function, $\mathcal{L}$.

In the *next-step* architecture, the network received observation input at each timestep, and was trained to match the output to the subsequent observation,

$$\mathcal{L} = \mathrm{MSE}(\boldsymbol{y}_t, \boldsymbol{o}_{t+1}) = \mathrm{E}[(\boldsymbol{y}_t - \boldsymbol{o}_{t+1})^2]$$

i.e., $\boldsymbol{y}_t = \widehat{\boldsymbol{o}}_{t+1}$, where $\widehat{\boldsymbol{o}}$ indicates a prediction of variable $\boldsymbol{o}$.

In the *masked* architecture, the network was trained to match the output to the current observation,

$$\mathcal{L} = \mathrm{MSE}(\boldsymbol{y}_t, \boldsymbol{o}_t)$$

i.e., $\boldsymbol{y}_t = \widehat{\boldsymbol{o}}_t$. However, the observation was only provided as input to the network every $k+1$ timesteps. That is,

$$\boldsymbol{W_{in}}\boldsymbol{o}_t \to 0, \ \mathrm{mod}(t, k+1) \neq 0.$$

Like the masked architecture, the *rollout* architecture was trained to match the output to the current observation. However, at every observation timestep, t, the RNN would rollout k timesteps,

$$\boldsymbol{h}_{t,\theta} = \mathrm{NormReLU}\left(\boldsymbol{W_{rec}}\boldsymbol{h}_{t,\theta-1} + \boldsymbol{W_{act}}\boldsymbol{a}_{t+\theta}\right)$$
$$\boldsymbol{y}_{t,\theta} = \mathrm{Sigmoid}\left(\boldsymbol{W_{out}}\boldsymbol{h}_{t,\theta}\right)$$

Where $\boldsymbol{h}_{t,0} = \boldsymbol{h}_t$, $\theta \in 1{:}k$ indexes the phase of the rollout, and the network was trained to match the output to the corresponding present ($\theta = 0$) or future ($\theta > 0$) observation.

$$\mathcal{L} = \mathrm{MSE}\left(\boldsymbol{y}_{t,\theta}, \boldsymbol{o}_{t+\theta}\right)$$

i.e., $\boldsymbol{y}_{t,\theta} = \hat{\boldsymbol{o}}_{t+\theta}$. For rollout-based networks trained with "looped" time, $\boldsymbol{h}_{t-1} = \boldsymbol{h}_{t-1,0}$ in eqn. 1, for "continuous" time, $\boldsymbol{h}_{t-1} = \boldsymbol{h}_{t-1,k}$.

Training

The trainable network parameters $\Omega = (\boldsymbol{W_{rec}}, \boldsymbol{W_{in}}, \boldsymbol{W_{act}}, \boldsymbol{W_{out}}, \boldsymbol{b})$ were initialized as follows:

$$\boldsymbol{W_{in}}, \boldsymbol{W_{act}} \rightarrow \mathcal{U}\left(-\sqrt{k_{in}}, \quad \sqrt{k_{in}}\right); \; k_{in} = \frac{1}{N_{obs} + N_{act}}$$

$$\boldsymbol{W_{out}} \rightarrow \mathcal{U}\left(-\sqrt{k_h}, \quad \sqrt{k_h}\right); \; k_h = \frac{1}{\mathrm{N}}$$

$$\boldsymbol{W_{rec}} \rightarrow \mathcal{U}\left(-\sqrt{k_h}, \quad \sqrt{k_h}\right) + \mathbb{I}\left(1 - \frac{1}{\tau}\right)$$

$$\boldsymbol{b} \rightarrow \boldsymbol{0}$$

Where τ is a hyperparameter that sets the neural timescale. $\boldsymbol{o}_t$ was passed through a dropout layer. The networks were trained using backpropagation through time (BPTT) with the RMSProp algorithm as implemented in *pytorch* (Paszke et al 2019), with $\alpha = 0.95$ and $\epsilon = 10^{-7}$, for 8x10^4 single-batch trials in the environment, unless otherwise stated. Network state was randomly initialized from $\mathcal{N}(0, \eta)$ on each trial. The learning rate was set by a global learning rate hyperparameter, λ, and then further scaled for each trainable parameter by its initialization value or an additional hyperparameter, as follows:

$$\mathrm{LearningRate}(\boldsymbol{W_{rec}}, \boldsymbol{W_{out}}) = \lambda\sqrt{k_h}$$

$$\mathrm{LearningRate}(\boldsymbol{W_{in}}, \boldsymbol{W_{act}}) = \lambda\sqrt{k_{in}}$$

$$\mathrm{LearningRate}(\boldsymbol{b}) = \lambda_b \lambda$$

$$\mathrm{WeightDecay}(\Omega) = \lambda_{wd} * \mathrm{LearningRate}(\Omega)$$

The training hyperparameters can be found in Table 1 for the example networks used and the range for the random hyperparameter sweep.

Hyperparameter	**Standard Value**	**Sweep range**	**Sweep distribution**
Global Learning rate, λ	2x10^{-3}	10$^{-3.5}$ - 10$^{-1.5}$	Log-uniform
Bias Learning rate, λ_b	0.1	10$^{-2.5}$ – 10^{0}	Log-uniform
Weight decay, λ_{wd}	3x10^{-3}	10^{-4} – 10^{-1}	Log-uniform

Sequence duration, T	600	30 - 1000	Log-uniform
BPTT truncation window	600	3 - 1000	Log-uniform
# Hidden Units, N	500	100 - 1000	Log-uniform
Neural timescale, τ	2	1 - 4	Uniform
Dropout probability	0.15	-	-
Noise std., η	0.03	-	-

Table 1. Training hyperparameters. Standard value was used in all experiments, unless stated otherwise. Values for the random hyperparameter population (**Figures S7, S10**) were sampled from the sweep range using the sweep distribution.

Sleep activity generation

To generate offline activity (sleep) in each RNN, the network was initialized from $\mathcal{N}(\mathbf{0}, \boldsymbol{\eta})$, and allowed to run for $\boldsymbol{T_{sleep}}$ timesteps (as specified for each analysis) without the observation input,

$$\boldsymbol{h}_{t+1} = \mathrm{NormReLU}(\boldsymbol{W_{rec}}\boldsymbol{h}_t + \boldsymbol{W_{act}}\boldsymbol{a}_t)$$

In the case of noise-generated offline activity, the action was also removed ($\boldsymbol{a}_t = 0$), and the standard deviation of noise injection (Eqn. 3), η, was increased to 0.2 unless otherwise indicated (**Figures S8, S16**).

In the case of adaptation-generated offline activity, an additional cell-autonomous slow variable, $\boldsymbol{c}_t \in \mathbb{R}^N$, was added to the RNN equations (compare to eqns 1, 3),

$$\boldsymbol{h}_t = \mathrm{AdaptNormReLU}(\boldsymbol{W_{rec}}\boldsymbol{h}_{t-1} + \boldsymbol{W_{act}}\boldsymbol{a}_t,\ \boldsymbol{c}_t)$$
$$\boldsymbol{c}_t = \left(1 - \frac{1}{\tau_a}\right)\boldsymbol{c}_{t-1} + \frac{b_a}{\tau_a}\boldsymbol{h}_{t-1}$$
$$\mathrm{AdaptNormReLU}(\boldsymbol{x}, \boldsymbol{y}) = \max\left(0, \quad \frac{\boldsymbol{x} - \mathrm{E}[\boldsymbol{x}]}{\mathrm{Std}[\boldsymbol{x}] + \epsilon} + \boldsymbol{b} + \mathcal{N}(0, \eta) - \boldsymbol{y}\right),$$

where $\tau_a = 100$ and $b_a = 0.5$ are the timescale and strength of adaptation. The effect of modifying their values is shown in **Figure S17**.

In the case of query-generated offline activity, $\boldsymbol{a}_t$ was randomly generated following the same action probabilities as wakefulness. In the case of the speed and head direction encoding, the dependence

on the speed signal was tested by setting the speed component to 0 and using the network with adaptation (**Figure S17**).

Analysis

Spatial representation

To decode the position represented by the network, the agent was run for a trial of duration $T = 1.5 \times 10^4$ timesteps, and the sequence of hidden unit activity, $\boldsymbol{h}_t$, was used to train a linear decoder with the following architecture.

$$\boldsymbol{z} = \boldsymbol{W}_{\text{decode}}\boldsymbol{h}$$
$$\text{P}[\boldsymbol{x}|\boldsymbol{h}] = \text{Softmax}(\boldsymbol{z})$$

in which $\text{P}[\boldsymbol{x}|\boldsymbol{h}]$ is the posterior probability of the agent being in position $\boldsymbol{x}$ given a hidden unit activity vector, and $\boldsymbol{W}_{\text{decode}} \in \mathbb{R}^{N_{pos} \times N}$ is the decoding matrix with N_{pos} being the number of possible positions in the environment and N being the number of hidden units in the network.

$\boldsymbol{W}_{\text{decode}}$ was trained to minimize the following loss:

$$\mathcal{L}_{decode} = \sum_t \text{CrossEntropy}(\boldsymbol{z}_t, \boldsymbol{x}_t)$$

in which $\boldsymbol{x}_t \in \mathbb{R}^{N_{pos}}$, is a one-hot encoding of the agent's position at time t. The network was trained using the AdamW optimizer with learning rate = 10^{-3} and weight decay = $3\text{x}10^{-1}$, for $5\text{x}10^3$ weight updates, with a batch of $5\text{x}10^3$ random timepoints used in each update to prevent overfitting. To decode the head direction to calculate the expected viewpoint during sleep, a second decoder was trained with $\boldsymbol{x}_t = \boldsymbol{HD}_t$ as a one-hot encoding of the agent's head direction at time t.

The decoding error was calculated by running an additional test trial of duration $T = 2 \times 10^3$ timesteps, and calculating the cityblock distance between the agents actual and decoded position,

$$\tilde{\boldsymbol{x}}_{\boldsymbol{t}} = \text{argmax}(\text{P}[\boldsymbol{x}|\boldsymbol{h}_t])$$
$$Error_t = \sum(\tilde{\boldsymbol{x}}_{\boldsymbol{t}} - \boldsymbol{x}_t)$$

To calculate spatial tuning curves and spatial information, the agent was run for a trial of duration $T = 1.5 \times 10^4$ timesteps. The tuning curve, or mean activation as a function of position, $h_i(x) = E[h_i|x]$ was

calculated for each unit using the *pynapple* package for neural data analysis (Viejo et al., 2023). The spatial information was then calculated for each unit's tuning curve (Skaggs et al., 1992),

$$I_i = \sum_x \frac{h_i(x)}{h} \log_2 \frac{h_i(x)}{h} p(x)$$

To calculate the spatial explained variance (%EVs) for each unit, the agent was run for another trial of duration $T = 5 \times 10^3$ timesteps. %EVs was taken to be the decrease in the variance of the unit's activity when the expected rate given the spatial tuning curve was removed. That is,

$$\%\mathrm{EV}_{\mathrm{s,i}} = 1 - \frac{\mathrm{Var}[h_i(x_t)]}{\mathrm{Var}[h_i]}$$

Replay analysis

To analyze the spatial extent, output plausibility, and trajectory statistics of the represented activity during sleep, the networks was run for 15 sleep trials of $T_{sleep} = 300$ timesteps each, and the linear decoder was used to calculate the decoded posterior, $\mathrm{P}[\boldsymbol{x}|\boldsymbol{h}_t]$, and decoded position $\widetilde{\boldsymbol{x}}_t$ and head direction $\widetilde{\boldsymbol{HD}}_t$ at each timestep.

Spatial extent of each timepoint was taken to be the furthest distance at which the 2D autocorrelation dropped below 0.01.

Plausibility of each timepoint was taken to be the spearman correlation between the network output $\boldsymbol{y}_t$ and the environmental observation from the decoded viewpoint, $obs(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{HD}}_t)$.

Trajectory statistics were calculated as in (Stella et al., 2019). The average squared Euclidian distance, $\langle \Delta\widetilde{\boldsymbol{x}}^2 \rangle$, was calculated between the decoded positions of all pairs of timepoints separated by $\Delta t = [0{:}\,15]$ timesteps in the same trial. A linear fit was then calculated for the relationship between $\log\langle \Delta\widetilde{\boldsymbol{x}}^2 \rangle$ and $\log\langle \Delta t \rangle$, producing a slope, α, that quantified the degree to which the trajectories reflect sub or supra-diffusive motion, and an intercept, G, that reflected the average speed of trajectory propagation.

Neural manifold analysis

To calculate the manifold properties of each network, we simulated a $T = 1.5 \times 10^4$ timestep wake trial and a $T_{sleep} = 1000$ timestep sleep trial.

To visualize the neural manifolds during wake and sleep, we used Isomap (Tenenbaum et al., 2000) with the hidden unit activation from 4x10^3 randomly sampled timepoints from the wake trial and all

timepoints from the sleep trial. The *scipy* implementation of isomap was used with parameters n_neighbors=150, n_components=2, and metric='cosine'.

Spatial RSA (sRSA) was taken to be $\rho\big(D_E(\boldsymbol{x}_t, \boldsymbol{x}_{t'}), D_C(\boldsymbol{h}_t, \boldsymbol{h}_{t'})\big)$, where $\rho(X, Y)$ is the spearman rank correlation between X and Y, $D_E(\boldsymbol{x}_t, \boldsymbol{x}_{t'}) = \sqrt{(\boldsymbol{x}_t - \boldsymbol{x}_{t'})^2}$ is the Euclidian distance between the agent's position at all pairs of timepoints t and t' during the wake trial, and $D_C(\boldsymbol{x}, \boldsymbol{y}) = 1 - \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{\|\boldsymbol{x}\|\|\boldsymbol{y}\|}$ is the cosine distance between the hidden unit activations at all corresponding pairs of timepoints $\boldsymbol{h}_t$ and $\boldsymbol{h}_{t'}$.

Sleep-Wake distance (SW dist) was taken to be the average distance between the neural activations during the sleep trial and the closest point in neural space during the wake trial. That is,

$$\mathrm{SWDist} = \langle \min_{t' \in wake} D_C\big(\boldsymbol{h}_{t \in sleep}, \boldsymbol{h}_{t' \in wake}\big)\rangle_{t \in sleep}$$

Experiments

The following experiments were run on networks after training.

Single-trial experience replay

To calculate the sleep reactivation of positions visited during wake experiences, 150 unique wake trajectories ($\boldsymbol{x}_0$, $\boldsymbol{HD}_0$, $\boldsymbol{a}_t$) of $T = 100$ timesteps were simulated. The resulting observation sequences were each used for a subsequent single weight update of a trained network with a range of learning rates (learning rate was varied for recurrent weights and biases, only). The resulting networks were then each run for 20 adaptation-generated sleep trials of $T_{sleep} = 100$ timesteps. The spearman rank correlation between the occupancy of the decoded position, $P[\tilde{\boldsymbol{x}}_t]$, and the position from corresponding wake trial $P[\boldsymbol{x}_t]$.

Object association learning

To test the network's ability to learn new associations between sensory inputs and positions, the network was placed into a new environment, which was identical to the original with the exception of a single-square bright green [0.3, 0.5, 0.3] "object". The network was then trained for 200 trials of $T = 1000$ timesteps with learning rate for the output weights doubled. Its ability to remember the new sensory association was assessed by running a $T = 1000$ timestep trial in the original (object-free) environment, and measuring the change of the value of the corresponding green pixel in the network output, $\boldsymbol{y}_t$, when the network was viewing the position where the object had been located.

Acknowledgements

The authors would like to thank Antonio Fernandez-Ruiz and Dileep George for helpful comments on the manuscript. This work was supported by the following sources DL: FRQNT Strategic Clusters Program (2020-RS4-265502 – Centre UNIQUE – Unifying Neuroscience and Artificial Intelligence – Québec) and the Richard and Edith Strauss Postdoctoral Fellowship in Medicine. RHE: Deepmind fellowship. AP&BR: New Frontiers in Research Fund (Exploration grant NFRFE-2021-00926 and HBHL - Healthy Brains for Healthy Lives (Innovative Ideas grant 1c-II-15). BR: NSERC (Discovery Grant: RGPIN-2020-05105; Discovery Accelerator Supplement: RGPAS-2020-00031; Arthur B. McDonald Fellowship: 566355-2022) and CIFAR (Canada AI Chair; Learning in Machine and Brains Fellowship). This research was enabled in part by support provided by (Calcul Québec) (https://www.calculquebec.ca/en/) and the Digital Research Alliance of Canada (https://alliancecan.ca/en). The authors acknowledge the material support of NVIDIA in the form of computational resources. AP: NSERC (Discovery Grant: RGPIN-2018-04600), CIHR (Project grants 155957 and 180330)

Author Contributions

DL, AP, BR designed the research. DL, AP, BR wrote the manuscript. DL, AE, RHE carried out analysis.

Competing Interests

The authors declare no competing interests.

Additional Information

Supplementary Information is available for this paper.

Correspondence and requests for materials should be addressed to Blake Richards (blake.richards@mila.quebec), Adrien Peyrache (adrien.peyrache@mcgill.ca), and Dan Levenstein (daniel.levenstein@mila.quebec).

Supplemental Figures

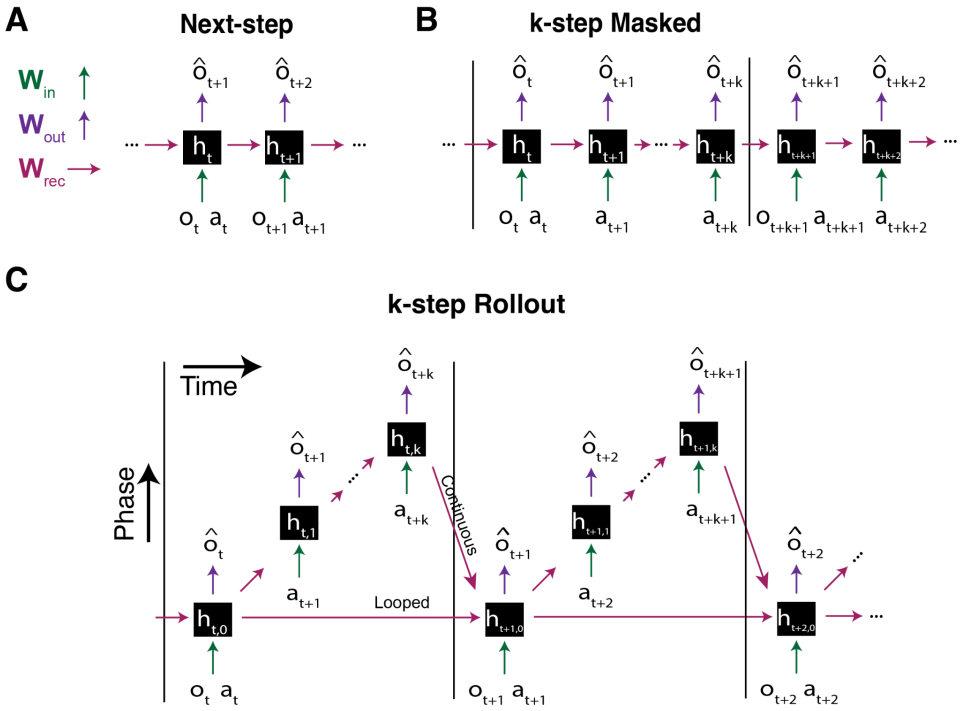

Figure S1: Block diagrams for predictive RNN architectures used in this study

All networks are trained using T-length sequences of observations and actions (default $T = 500$), in which o_t is the sensory observation at time t, a_t is the subsequent action taken, and $\hat{o_t}$ is the network's prediction of o_t. All weights, as well as the bias of the recurrent hidden units, h, are trained using back propagation through time. Further details can be found in the methods. **A:** The next step pRNN is trained to predict the observation at the next time step. **B:** The masked pRNN is trained to predict the observation at the current time step, which is not included in the input except every $k + 1$ time steps. The network is engaged in sequential predictive learning when $k > 1$, as the network must predict k time steps of masked inputs in a row. **C:** The rollout pRNN is trained to predict a k-length sequence of future observations. Time in the rollout pRNN can be "continuous" ($h_{t+1,0} = f(h_{t,k}, o_{t+1}, a_{t+t})$) or "looped" ($h_{t+1,0} = f(h_{t,0}, o_{t+1}, a_{t+t})$), depending on if the recurrent input in the beginning of the rollout comes from the hidden state at the end or beginning of the previous rollout.

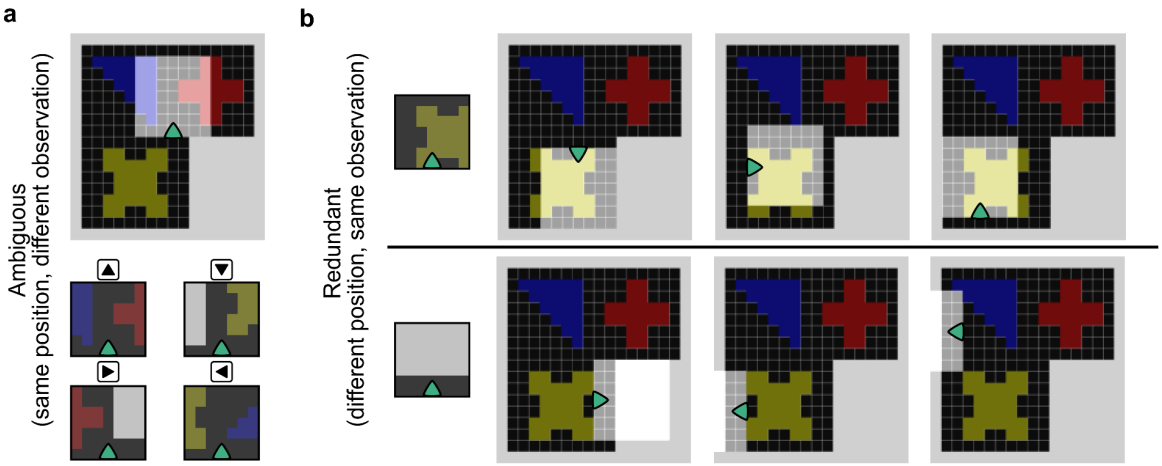

Figure S2: Spatial ambiguity and redundancy of sensory input in the gridworld environment.

A: Sensory input in the environment is ambiguous – the same position can have different sensory inputs, depending on the orientation ("head direction") of the agent. **B**: Sensory input in the environment is redundant – different positions in the environment can have identical sensory inputs.

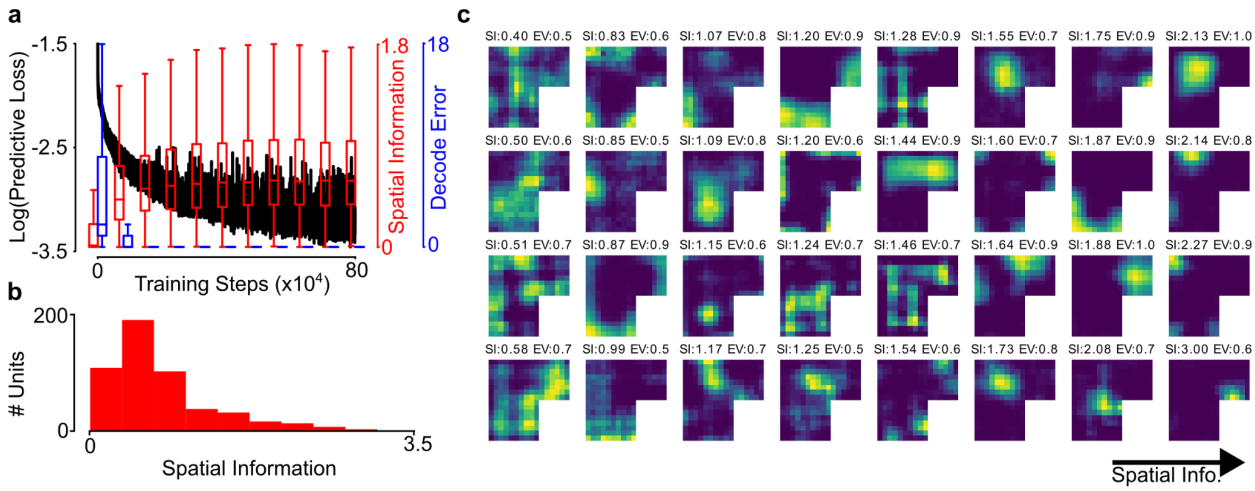

Figure S3: Training and Spatial Tuning in the Next-Step pRNN

A: The evolution of predictive loss, distribution of spatial information, and decode error over training. Each training step is a single wake epoch. **B:** Distribution of spatial information over all cells in the trained network. **C:** Tuning curves from a random selection of tuned cells (>50% variance explained by space).

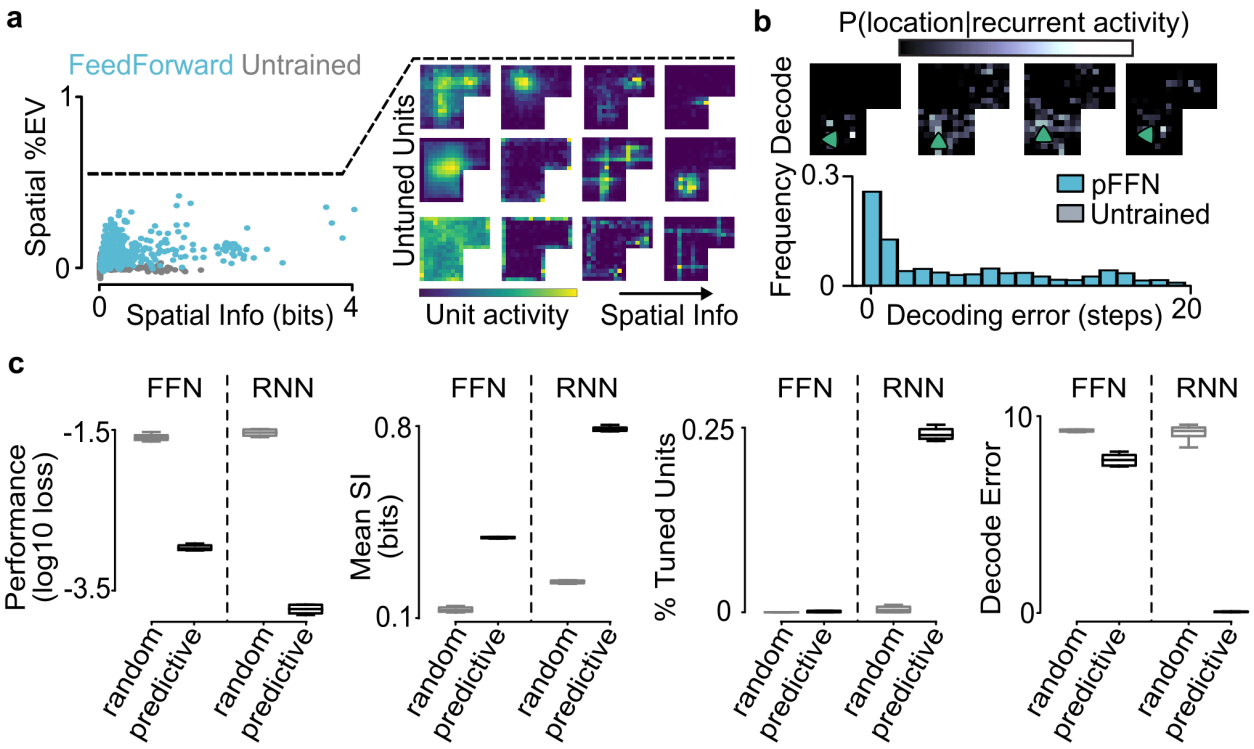

Figure S4: Spatial tuning in feedforward and untrained networks

A: (Left) Explained variance by spatial position and spatial information of all units in the example predictive feed-forward network and units from an random feedforward network. (Right) spatial tuning curves from a selection of example units from the network, sorted by the spatial information of their tuning curves. Note that no units have more than 50% of their activity variance explained by the agent's position. Compare to Figure 2c. **B:** (Top) Decoded position from a linear decoder on hidden unit activity in the feedforward network for four example timesteps. (Bottom) distribution of decoding error over a 1000-timestep trajectory in the environment, for the example feedforward and untrained network. (Compare to Figure 2d) **C:** Predictive performance, mean spatial information, % of tuned cells, and mean decoding error for random and predictive feedforward networks (FFN) and recurrent networks (RNN). Box and whiskers correspond to distribution over 9 seeds.

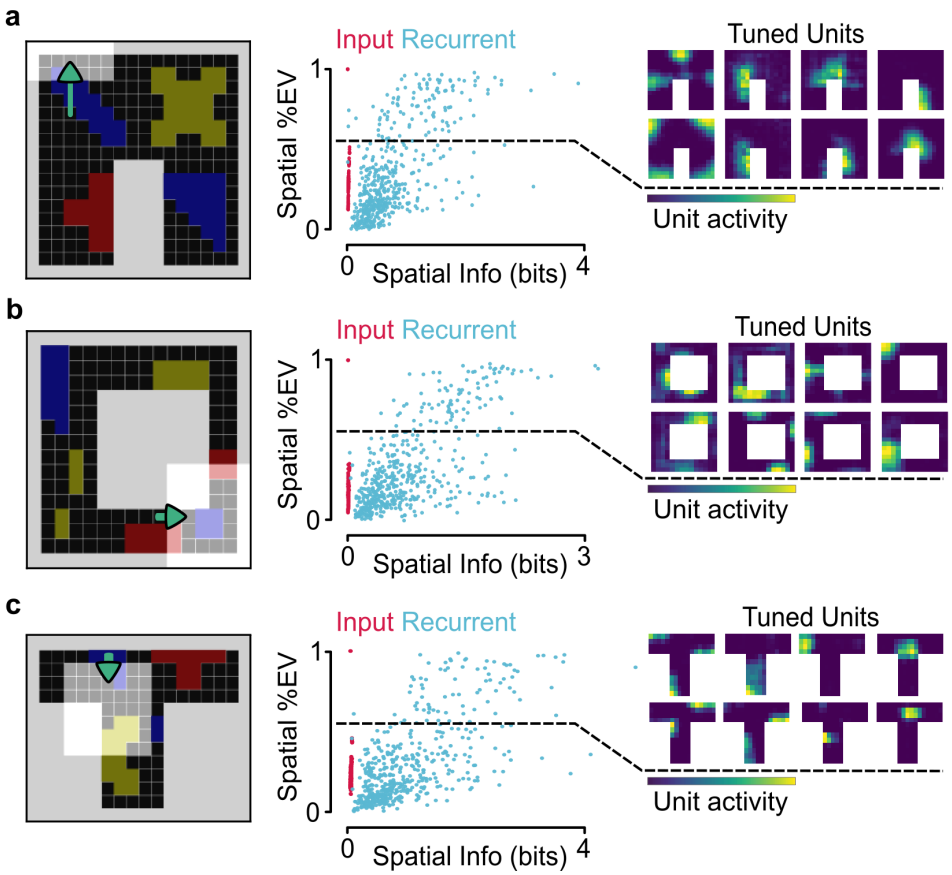

Figure S5: Spatial tuning of recurrent units in gridworld environments with different geometries

A: A U-shaped environment. (Left) Top-down view of the environment. (Middle) Variance explained by the agent's position and spatial information for all recurrent and input units in an example next-step pRNN. (Right) Spatial tuning curves for example tuned units (units with more than 50% variance explained by spatial position). Compare to Figure 2c. **B:** Same as panel a, for a circular environment. **C:** Same as panel a, for a T-shaped environment.

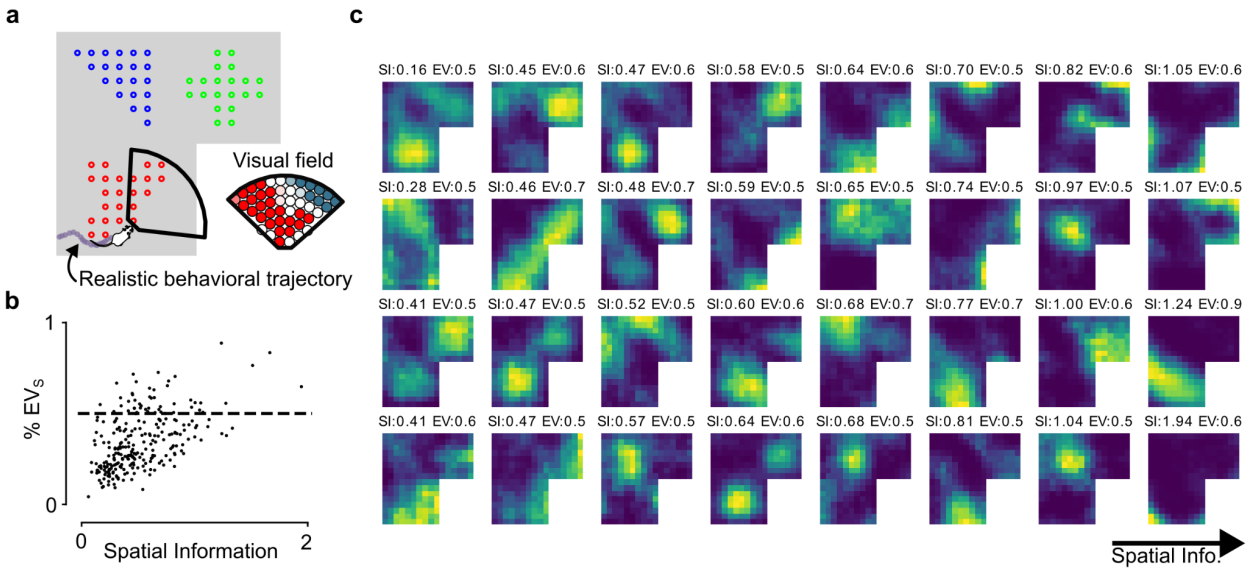

Figure S6: Spatially-tuned cells in a next-step pRNN in the RatInABox environment.

A: The RatInABoxEnvironment (George et al 2024). An agent takes rodent-like continuous trajectories in the environment, and collects egocentric visual inputs. **B**: %EVs and Spatial information for all units in the network. **C**: Tuning curves from a random selection of tuned cells (>50% variance explained by space).

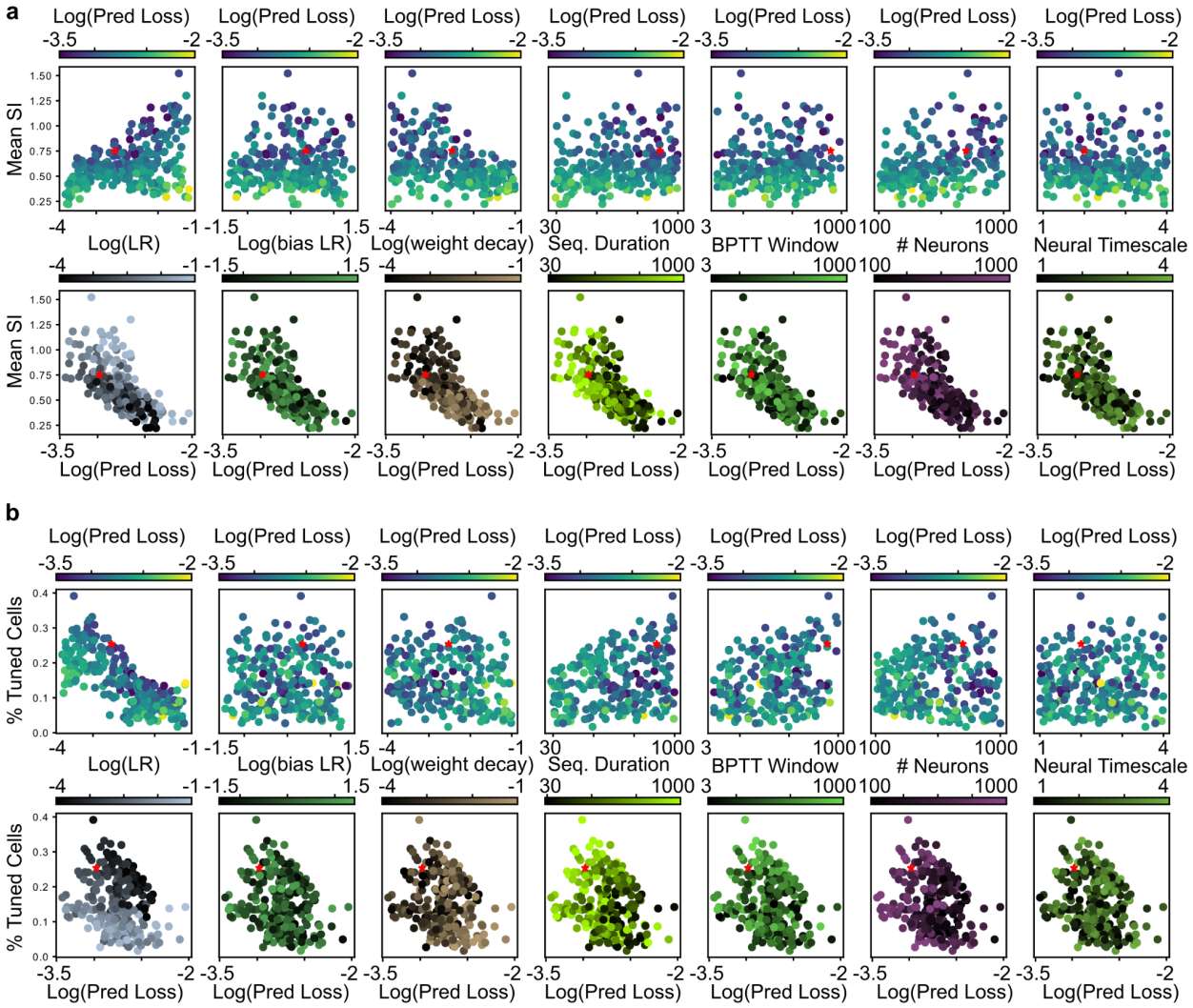

Figure S7: Random hyperparameter population

A: (Top) Spatial information and predictive loss (color) for all networks in the random hyperparameter sweep, as a function of each hyperparameter. (Bottom) Spatial information (color) and predictive loss for all networks in the random hyperparameter sweep, as a function of each hyperparameter. **B:** Similar to A, for % Tuned Cells.

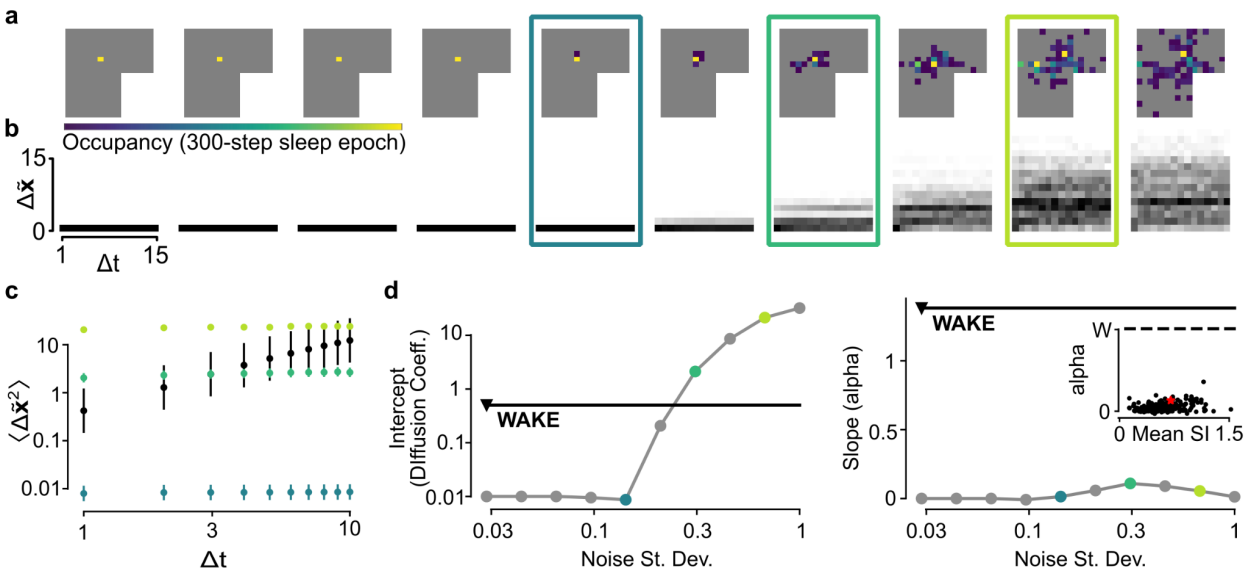

Figure S8: Sleep activity in the Next-step pRNN as a function of the variance of internal noise.

A: Spatial occupancy histograms for sleep epochs with different levels of noise. Note: each epoch has a different seed for initial hidden unit state and noise generation. **B:** Spatial transition statistics for different levels of noise. **C:** Diffusion plot for three example levels of noise. **D:** Diffusion constant (intercept, right) and alpha (slope, left) of offline activity as a function of noise. Note: due to layer normalization, the standard deviation of recurrent input is 1, black triangle indicates noise level during training, and noise was set to 0.5 for Figure 2 f-i. Inset: alpha for all networks in the random hyperparameter population.

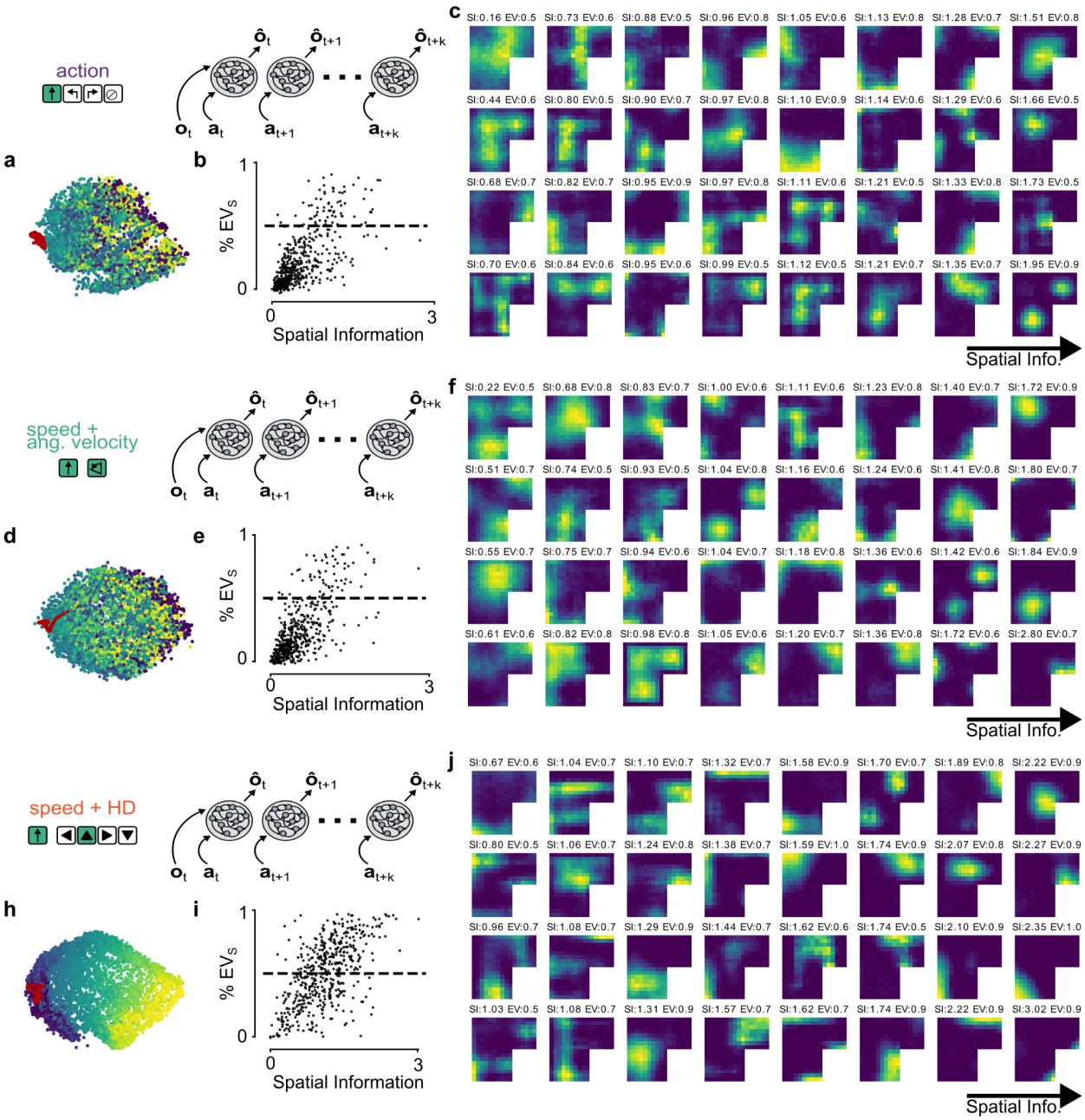

Figure S9: Action-encoding-dependence of cognitive map emergence

A: Isomap projection for a 5-step masked network trained with action identity (one-hot encoding). **B:** Spatial information and %EVs for all cells in the network. **C:** Tuning curves from a random selection of tuned cells (>50% variance explained by space). **D-F**: Same as A-C, for a 5-step masked network trained with speed and angular velocity. **H-J:** Same as A-C, for a 5-step masked network trained with speed and head direction (different seed from figure 3C,D).

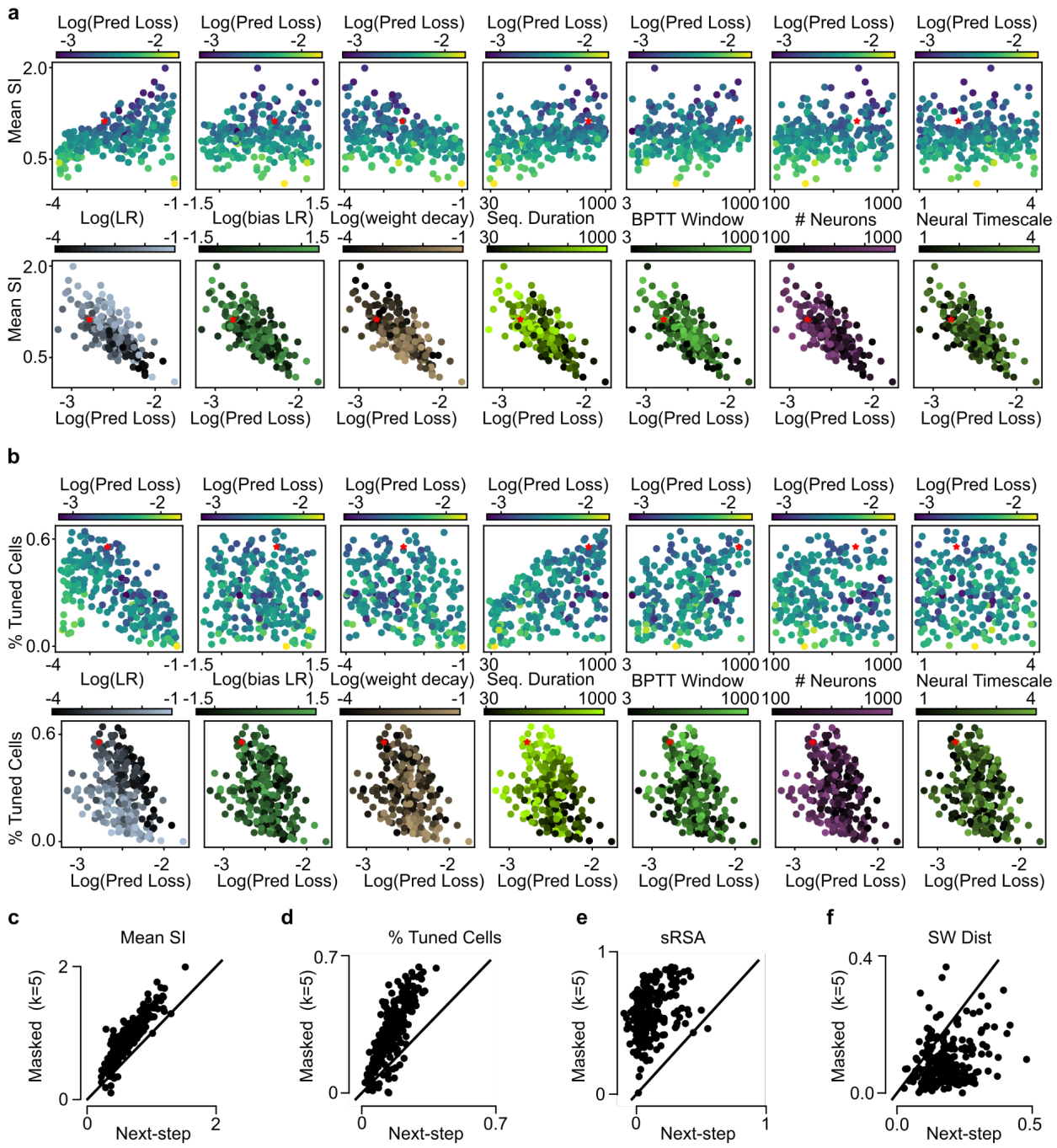

Figure S10: Random hyperparameter population - 5-step masked networks

A: (Top) Spatial information and predictive loss (color) for all networks in the random hyperparameter sweep, as a function of each hyperparameter. (Bottom) Spatial information (color) and predictive loss for all networks in the random hyperparameter sweep, as a function of each hyperparameter. Compare to S7a **B:** Similar to A, for % Tuned Cells. Compare to S7b. **C:** Comparison of mean spatial information between 5-step masked and next-step networks with otherwise identical hyperparameters. **D:** Same as c, for % tuned cells. **E:** Same as c, for sRSA. **DF:** Same as c, for SW Dist.

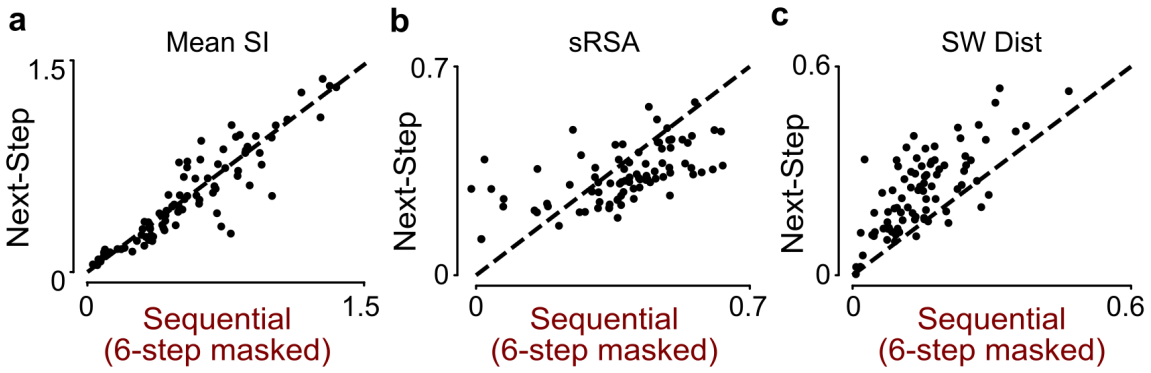

Figure S11: Rat in a box - Sequential vs Next step

A: Comparison of mean spatial information between 5-step masked and next-step networks trained in the RatInABox environment, with otherwise identical hyperparameters. **B:** Same as a, for % tuned cells. **C:** Same as a, for sRSA.

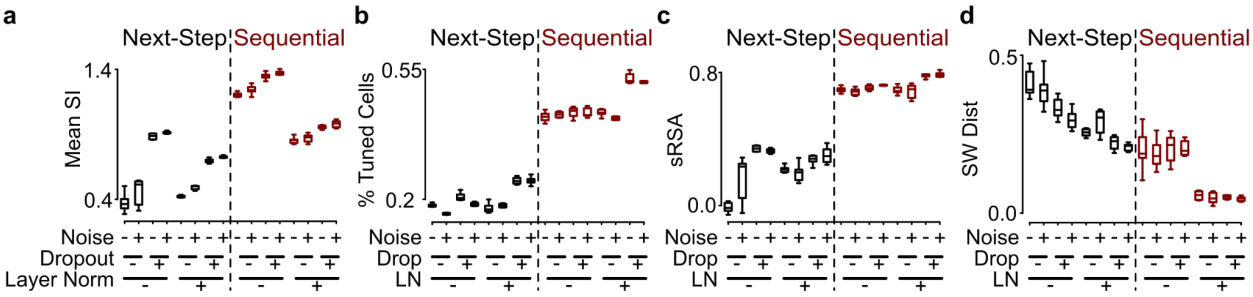

Figure S12: The effect of layer normalization, noise injection, and dropout

A-D: A comparison of mean spatial information (**a**), % of tuned cells (cells with >50% variance explained by spatial position, **b**), sRSA (**c**), and SW dist (**d**), across 9 different seeds of next-step and 5-step masked pRNNs trained with/without noise injection, dropout, and layer normalization (see Methods).

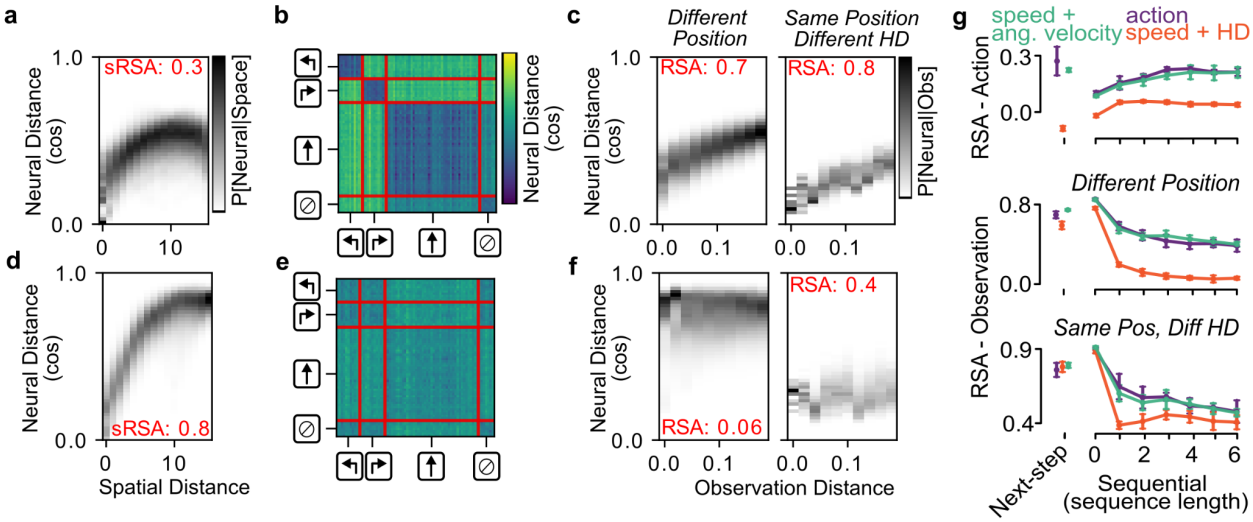

Figure S13: Neural representation of action, observation, and position in next-step and 5-step masked pRNN

A: Neural distance between hidden unit activations of a next-step pRNN at different timepoints, sorted by action identity. **B:** Distribution of neural distance between hidden unit activations at different timepoints, conditioned on the (pixel) distance between the observations at those timepoints, for distant positions (left), and for the same position but different head direction (right). **C:** Distribution of neural distance between hidden unit activations at different timepoints, conditioned on the spatial distance between the agent's position at those timepoints. **D-F:** Same as A-C, for a 5-step masked pRNN. **G:** Representational similarity analysis of the relationship between neural activity in pRNNs and action identity (top), observation in different positions (>4 steps apart, middle), and observation in the same position but different head direction (bottom). pRNNs include next-step pRNNs and masked pRNNs with k=0 (autoencoder), k=1 (single-step), and k=2-6 (sequential), each trained with one hot (action), speed+HD, and speed+AV encoding of the action input.

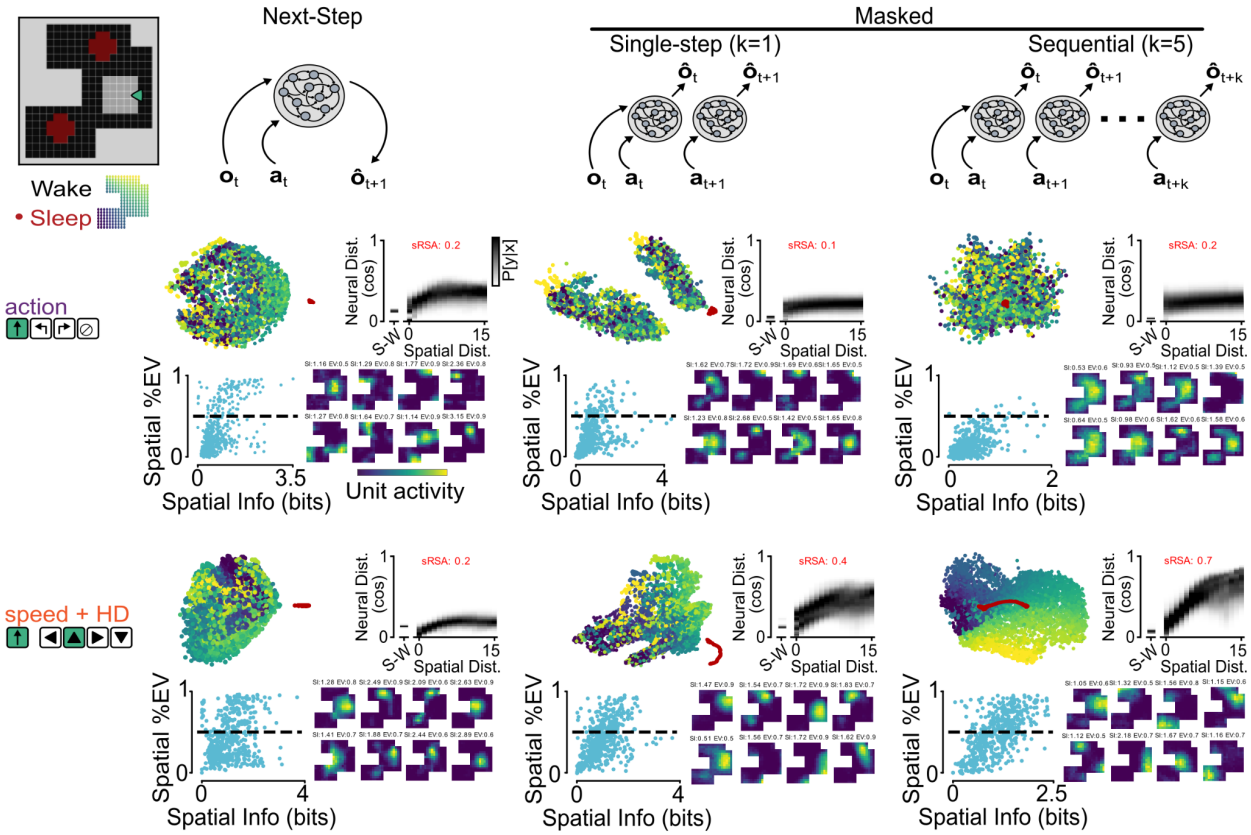

Figure S14: Spatial tuning and cognitive map in an environment with more sensory redundancy

Spatial representation and neural manifold in a gridworld environment with a high degree of redundancy (aliasing). In this environment, the visual field is reduced to a 5x5 grid in front of the agent, and many more positions have identical observation due to the identical floor cues and unmarked floor locations. Comparison is made between next-step (left), single-step masked (k=1, middle), and sequential masked (k=5, right) pRNNs, each trained with action input encoded as action identity (top) or speed+HD (bottom).

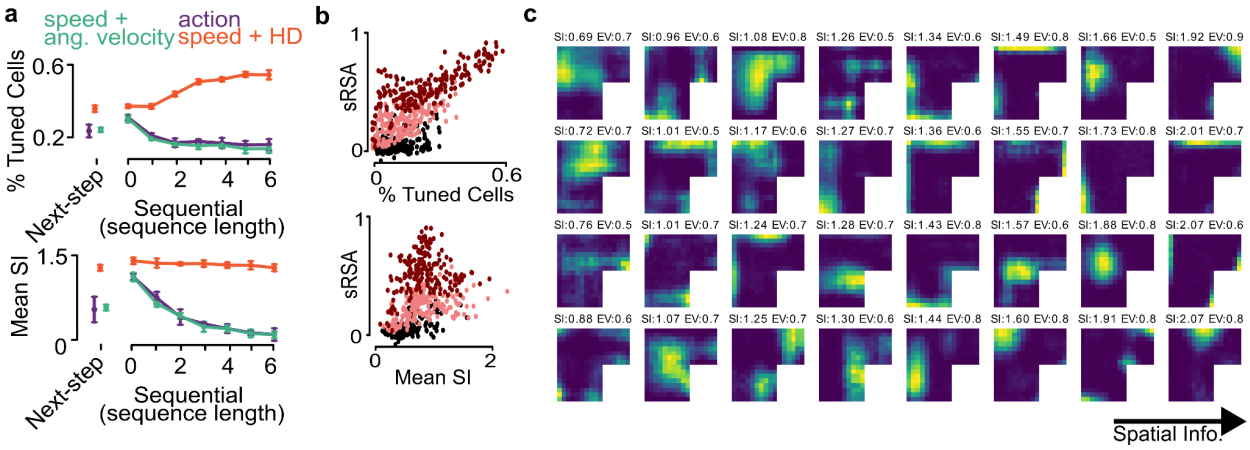

Figure S15: Percentage of spatially tuned cells relates to cognitive map formation

A: % Tuned cells (top) and mean SI (bottom) for next-step pRNNs and masked pRNNs with k=0 (autoencoder), k=1 (single-step), and k=2-6 (sequential), each trained with one hot (action), speed+HD, and speed+AV encoding of the action input, a_t. Mean and standard deviation calculated over 9 seeds. **B:** sRSA as a function of % Tuned cells (top) and mean spatial information (bottom) for the random hyperparameter population of Next-step, single-step, and 5-step masked pRNNs. **C**: Tuning curves from a random selection of tuned cells (cells >50% variance explained by space).

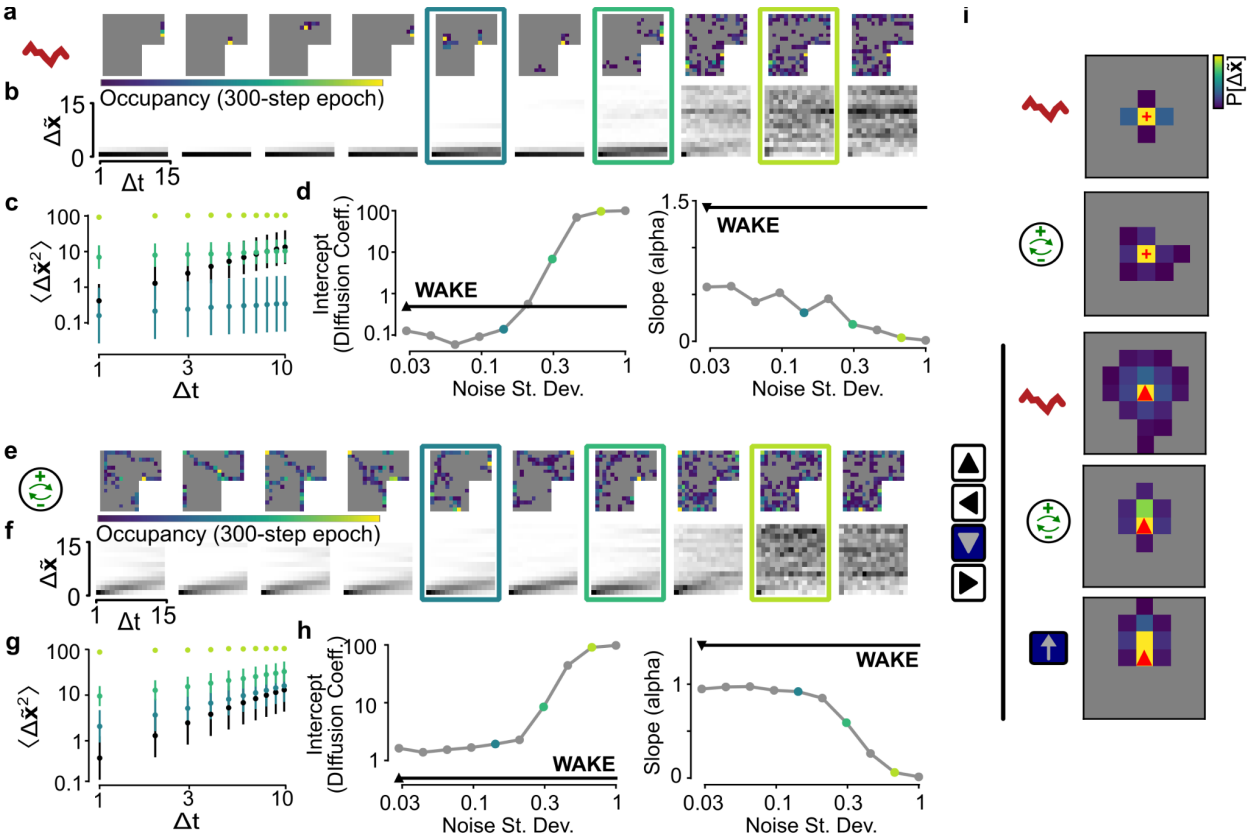

Figure S16: Sleep activity in the 5-step masked pRNN as a function of the variance of internal noise.

A-D: Similar to Figure S8 A-D, for a 5-step masked pRNN. **A:** Spatial occupancy histograms for sleep epochs with different levels of noise. Note: each epoch has a different seed for initial hidden unit state and noise generation. **B:** Spatial transition statistics for different levels of noise. **C:** Diffusion plot for three example levels of noise. **D:** Diffusion constant (intercept, right) and alpha (slope, left) of offline activity as a function of noise. Note: due to layer normalization, the standard deviation of recurrent input is 1, black triangle indicates noise level during training, and noise was set to 0.2 for all +noise conditions in Figure 4. **E-H:** Similar to A-D, but with adaptation-generated sleep activity. **I:** 1-step transition probability for decoded position during noise-generated, adaptation-generated, and query-directed sleep activity.

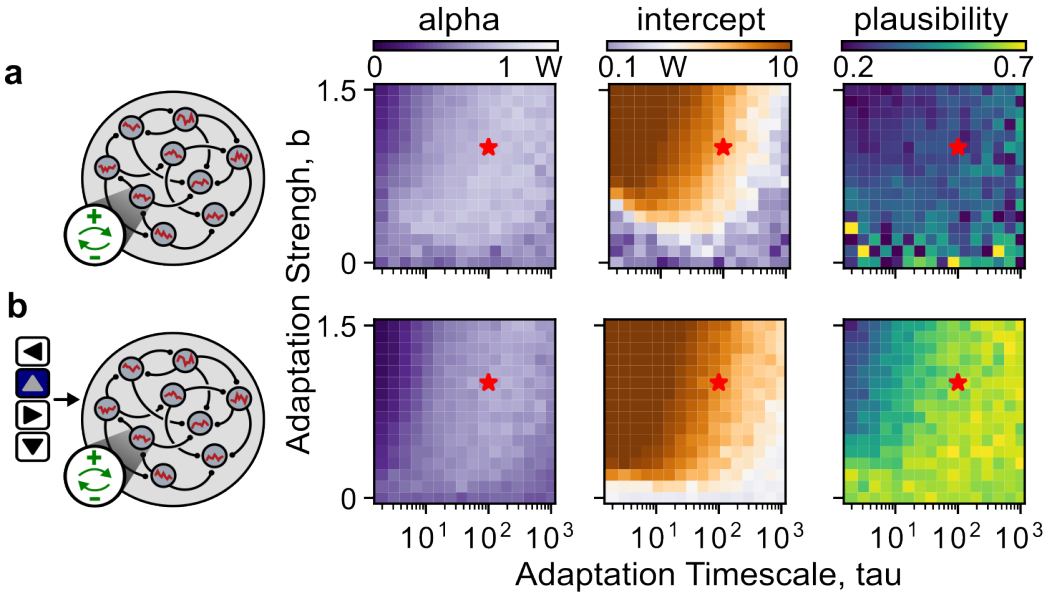

Figure S17: The effect of adaptation parameters on offline trajectory statistics and plausibility

A: Alpha (left), intercept (middle), and output plausibility (right) of offline activity as a function of the adaptation strength and timescale during adaptation-driven offline activity in the 5-step masked pRNN. **B:** Same as A, with the head direction query.

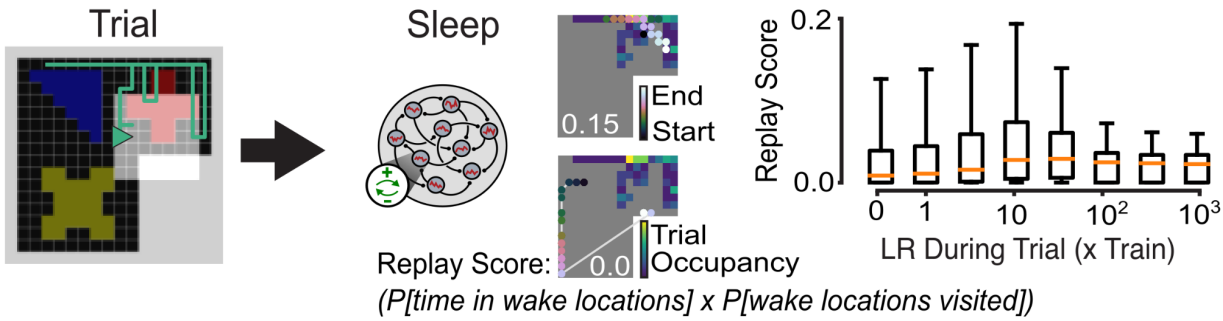

Figure S18: Reactivation of positions from a single-trial.

A: The trained agent is trained for a single additional wake trial of 100 timesteps. It is then run for a single 100-timestep trial of adaptation-driven sleep activity. Replay score is calculated as (the fraction of time in the sleep trial spent in locations from the wake trail) x (the fraction of wake locations visited). Therefore, a sleep trial in which the sleep visits exactly the same positions as the agent was in during wake will have a replay score of 1. Two example sleep trials are shown for the example wake trial, one with a high replay score and one with a replay score of 0. **B:** Distribution of replay scores over 3000 sleep trials as a function of the learning rate (recurrent weights and biases only) during the wake trial.

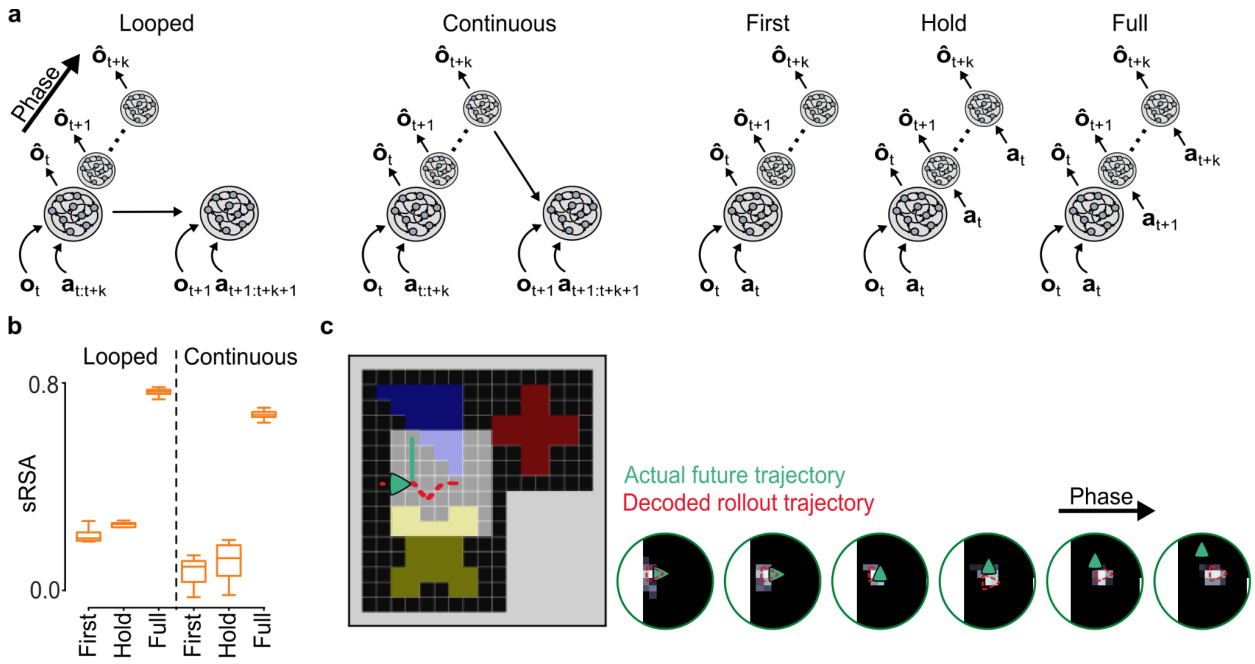

Figure S19: Rollout-based pRNN needs to be trained with true future actions, but once trained can use untaken action to simulate possible trajectories

A: Rollout-based pRNN can be trained either with looped or continuous time, and with actions only presented at the first timestep of the rollout (first), with the first action throughout the rollout (hold), or with the actual future action sequence (full). **B:** Comparison of sRSA in networks trained with looped/continuous time and first/hold/full actions provided during the rollout. Only networks trained with the full action sequence develop a cognitive map, per high sRSA, but cognitive map emerges whether time is looped or continuous. Box/whisker indicates distribution over 6 seeds. **C:** An example 5-step rollout in a trained pRNN. Action sequence provided to the network indicated forward movement for the entire rollout, while the actual future actions involved a left turn (teal). Decoded network activity during the rollout continues ahead of the agent (red), diverging from the actual future trajectory.

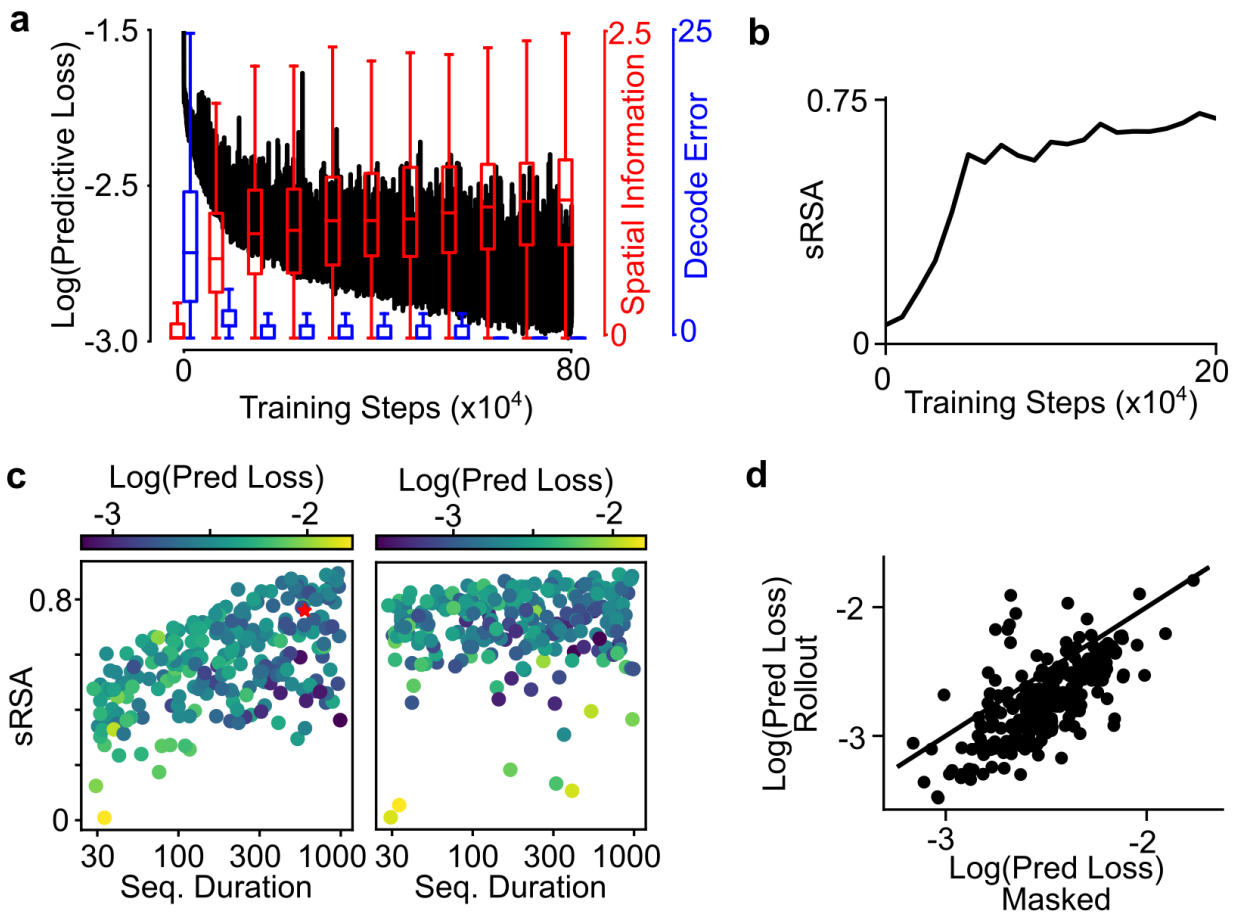

FIgure S20: Rollout-based sequential pRNN learns higher-sRSA representation with shorter sequences, and better sensory prediction.

A: Same as S1A, for a 5-step masked network. **B:** Evolution of sRSA over the first 20,000 timesteps of training for the network in A. Note that sRSA reaches a gradual plateau at ~5,000 timesteps. **C:** sRSA for all networks in the random hyperparameter population of 5-step masked (left) and 5-step rollout (right) networks, as a function of the duration of sequences used for training. **D:** Predictive loss for hyperparmeter-matched 5-step masked and rollout-based sequential pRNNs.

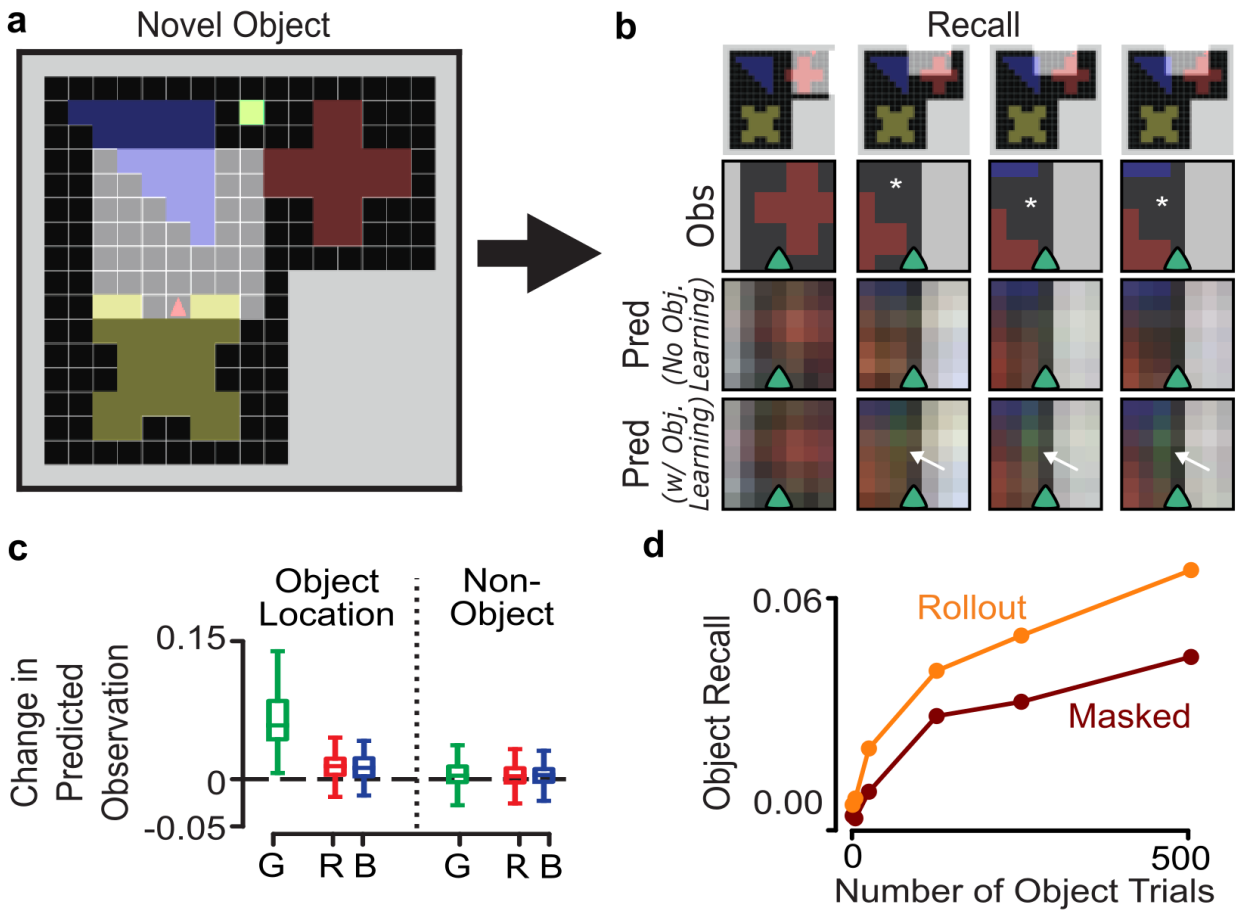

FIgure S21: Rollout-based sequential pRNN can quickly learn and recall novel stimuli.

A: The novel object environment. After training, the network is exposed to an environment with a novel object (green square), and trained for a small number of wake epochs. B: After novel object training, the agent is placed back in the original environment (no object), and the activation in the green pixel at the object's location (white star) is compared to the network before object exposure. C: Difference between output of object-trained and original network, for each pixel color at the object location and non-object locations. D: Comparison of object recall (change in green pixel value at object location) between rollout-based and masked sequential (k=5) predictive networks, after a given number of trials in the novel object environment.

References

1. O'Keefe, J. & Nadel, L. *The Hippocampus as a Cognitive Map*. (1**9**78).
2. Pavlides, C. & Winson, J. Influences of hippocampal place cell firing in the awake state on the activity of these cells during subsequent sleep episodes. *J. Neurosci.* **9**, **2**907–2918 (1989).
3. Skaggs, W. E. & McNaughton, B. L. Replay of *Neurona*l Firing Sequences in Rat Hippocampus During Sleep Following Spatial Experience. *Science* **271**, 1870–1873 (19**96**).
4. Carr, M. F., Jadhav, S. P. & Frank, L. M. Hippocampal replay in the awake state: a potential substrate for memory consolidation and retrieval. *Nat. Neurosci.* **14**, 147–153 (**20**11).
5. Pfeiffer, B. E. & Foster, D. J. Hippocampal place-cell sequences depict future paths to remembered goals. *Nature* **497**, 74 81 (2013).
6. Igata, H., Ikegaya, Y. & Sasaki, T. Prioritized experience replays on a hippocampal predictive map for learning. *Proc. Natl. Acad. Sci. United States Am.* **118**, e201126**611**8 (2021).
7. Girardeau, G. J., Benchenane, K., Wiener, S. I., Buzsáki, G. & Zugaro, M. Selective suppression of hippocampal ripples impairs spatial memory. *Nat Neurosci* **12**, 1**22**2 1223 (**20**09).
8. Recanatesi, S. *et al.* Predictive learning as a network mechanism for extracting low-dimensional latent space representations. *Nat Commun* **12**, 1417 (2021).
9. Levy, W. B. A Computational Approach to Hippocampal Function. *Psychol Learn Motiv* **23**, 243–**30**5 (1989).
10. Lisman, J. & Redish, A. D. Prediction, sequences and the hippocampus. *Philos. Trans. R. Soc. B: Biol. Sci.* **364**, 1193–1201 (2009).
11. Stachenfeld, K. L., Botvinick, M. M. & Gershman, S. J. The hippocampus as a predictive map. Nat Neurosci 20, 1951 22 (2017).
12. Kay, K. et al. Constant Sub-second Cycling between Representations of Possible Futures in the Hippocampus. *Cell* **180**, 552-567.e**25** (2020).
13. Burgess, N., Recce, M. & O'Keefe, J. A model of hippocampal function. *Neural Netw.* **7**, 10**65**–1081 (1994).
14. McNaughton, B. L. et al. Deciphering The Hippocampal Polyglot: the Hippocampus as a Path Integration System. *J. Exp. Biol.* **199**, 173–185 (1996).
15. Buzsáki, G. & Moser, E. I. Memory, navigation and theta rhythm in the hippocampal-entorhinal system. Nat Neurosci **16**, 130–1**38** (2013).
16. Scoville, W. B. & Milner, B. LOSS OF RECENT MEMORY AFTER BILATERAL HIPPOCAMPAL LESIONS. *J Neurology Neurosurg Psychiatry* **20**, 11 (1957).
17. Eichenbaum, H. B. Prefrontal-hippocampal interactions in episodic memory. *Nat Rev Neurosci* **18**, 547 558 (2017).
18. Morris, R. G. M., Garrud, P., Rawlins, J. N. P. & O'Keefe, J. Place navigation impaired in rats with hippocampal lesions. *Nature* **297**, 681–683 (1982).
19. Liu, X. et al. Optogenetic stimulation of a hippocampal engram activates fear memory recall. Nature **484**, 381–385 (2012).
20. Miller, K. J., Botvinick, M. M. & Brody, C. D. Dorsal hippocampus contributes to model-based planning. Nat. Neurosci. 20, 1269–1276 (2017).
21. Wikenheiser, A. M. & Redish, A. D. Hippocampal theta sequences reflect current goals. Nat. Neurosci. 18, 289–294 (2015).
22. Buckner, R. L. The Role of the Hippocampus in Prediction and Imagination. *Annu. Rev. Psychol.* **61**, 27–48 (2010).
23. Hassabis, D., Kumaran, D., Vann, S. D. & Maguire, E. A. Patients with hippocampal amnesia cannot imagine new experiences. Proc. Natl. Acad. Sci. **104**, 1726–17**31** (2007).
24. Buzsáki, G. Two-stage model of memory trace formation: A role for "noisy" brain states. *Neuroscience* **31**, 551–570 (1989).
25. Vanderwolf, C. H. Hippocampal electrical activity and voluntary movement in the rat. *Electroen Clin Neuro* **26**, 407 418 (1969).
26. Sun, W. et al. Learning produces a hippocampal cognitive map in the form of an orthogonalized state machine. (2023) doi:10.1101/2023.08.03.551900.
27. Tang, W., Shin, J. D. & Jadhav, S. P. Geometric transformation of cognitive maps for generalization across hippocampal-prefrontal circuits. *Cell Rep.* **42**, 112246 (2023).

28. Guo, W., Zhang, J. J., Newman, J. P. & Wilson, M. A. Latent learning drives sleep-dependent plasticity in distinct CA1 subpopulations. *bioRxiv* 2020.02.27.967794 (2020) doi:10.1101/2020.02.27.967794.
29. Esparza, J., Sebastián, E. R. & Prida, L. M. de la. From cell types to population dynamics: Making hippocampal manifolds physiologically interpretable. *Curr. Opin. Neurobiol.* **83**, **102**800 (2023).
30. Buzsáki, G. Hippocampal sharp wave-ripple: A cognitive biomarker for episodic memory and planning. Hippocampus 25, 1073–188 (2015).
31. Pfeiffer, B. E. The content of hippocampal "replay." Hippocampus 30, 6–18 (2020).
32. Gupta, A. S., Meer, M. A. A. van der, Touretzky, D. S. & Redish, A. D. Hippocampal replay is not a simple function of experience. *Neuron* 65, 695 705 (2010).
33. Stella, F., Baracskay, P., O'Neill, J. & Csicsvari, J. Hippocampal Reactivation of Random Trajectories Resembling Brownian Diffusion. Neuron 102, 4**50** 461.e7 (2019).
34. Buzsáki, G. Hippocampal sharp waves: Their origin and significance. *Brain Res.* **398**, 242–2**52** (1986).
35. Isomura, Y. et al. Integration and segregation of activity in entorhinal-hippocampal subregions by neocortical slow oscillations. Neuron 52, 871 882 (2006).
36. O'Neill, J., Boccara, C. N., Stella, F., Schoenenberger, P. & Csicsvari, J. Superficial layers of the medial entorhinal cortex replay independently of the hippocampus. *Science* **355**, 184–188 (2017).
37. Zutshi, I. & Buzsáki, G. Hippocampal sharp-wave ripples and their spike assembly content are regulated by the medial entorhinal cortex. *Curr. Biol.* (2023) doi:10.1016/j.cub.2023.07.039.
38. Peyrache, A. Querying hippocampal replay with subcortical inputs. Curr. Opin. Neurobiol. **77**, 102645 (2022).
39. Wilson, M. & McNaughton, B. Reactivation of hippocampal ensemble memories during sleep. Science **265**, 676 679 (1994).
40. Ji, D. & Wilson, M. A. Coordinated memory replay in the visual cortex and hippocampus during sleep. Nat Neurosci 10, **100 107** (2007).
41. Samsonovich, A. & McNaughton, B. L. Path Integration and Cognitive Mapping in a Continuous Attractor Neural Network Model. *J Neurosci* **17**, 5900–5920 (1997).
42. Battaglia, F. P. & Treves, A. Attractor neural networks storing multiple space representations: A model for hippocampal place fields. *Phys Rev E* **58**, 7738–7753 (1998).
43. Tsodyks, M. Attractor neural network models of spatial maps in hippocampus. Hippocampus 9, 481–489 (1999).
44. Khona, M. & Fiete, I. R. Attractor and integrator networks in the brain. *Nat. Rev. Neurosci.* **23**, 744–766 (2022).
45. Dong, X., Chu, T., Huang, T., Ji, Z. & Wu, S. Noisy Adaptation Generates Lévy Flights in Attractor Neural Networks. *35th Conference on Neural Information Processing Systems (NeurIPS 2021)* (2021).
46. Spalla, D., Cornacchia, I. M. & Treves, A. Continuous attractors for dynamic memories. *Elife* **10**, e69499 (2021).
4**7**. Burak, Y. & Fiete, I. R. Accurate Path Integration in Continuous Attractor Network Models of Grid Cells. *PLoS Comput. Biol.* **5**, e1000**29**1 (2009).
48. Sorscher, B., Mel, G. C., Ocko, S. A., Giocomo, L. M. & Ganguli, S. A unified theory for the computational and mechanistic origins of grid cells. Neuron **111**, 121-137.e13 (2023).
49. Schaeffer, R. et al. Self-Supervised Learning of Representations for Space Generates Multi-Modular Grid Cells. *arXiv* (2023) doi:10.48550/arxiv.2**31**1.02316.
50. Krishna, N. H., Bredenberg, C., Levenstein, D., Richards, B. A. & Lajoie, G. Sufficient conditions for offline reactivation in recurrent neural networks. *ICLR 2024* (2024).
51. Meshulam, L., Gauthier, J. L., Brody, C. D., Tank, D. W. & Bialek, W. Collective Behavior of Place and Non-place Neurons in the Hippocampal Network. Neuron 96, 1 **27** (2017).
52. Levy, E. R. J. et al. A manifold neural population code for space in hippocampal coactivity dynamics independent of place fields. Cell Rep. 42, 113142 (2023).
53. Stefanini, F. et al. A Distributed Neural Code in the Dentate Gyrus and in CA1. Neuron 107, 703-716.e4 (2020).
54. Gornet, J. & Thomson, M. Automated mapping of virtual environments with visual predictive coding. arXiv (2023) doi:10.48550/arxiv.2308.10913.
55. Russek, E. M., Momennejad, I., Botvinick, M. M., Gershman, S. J. & Daw, N. D. Predictive representations can link model-based reinforcement learning to model-free mechanisms. *PLOS Computational Biology* **13**, e1005768 (2017).
56. Davachi, L. & DuBrow, S. How the hippocampus preserves order: the role of prediction and context. *Trends Cogn. Sci.* **19**, 92–99 (2015).
57. Kok, P. & Turk-Browne, N. B. Associative Prediction of Visual Shape in the Hippocampus. J. Neurosci. 38, 6888–6899 (2018).

58. Aitken, F. & Kok, P. Hippocampal representations switch from errors to predictions during acquisition of predictive associations. *Nat. Commun.* **13**, 3294 (2022).
59. Miller, A. M. P. et al. Emergence of a predictive model in the hippocampus. Neuron **11**1, 1952-1965.e5 (2023).
60. Levy, W. B. A sequence predicting CA3 is a flexible associator that learns and uses context to solve hippocampal-like tasks. Hippocampus 6, 579–590 (1996).
61. Johnson, A. & Redish, A. D. Neural Ensembles in CA3 Transiently Encode Paths Forward of the Animal at a Decision Point. J Neurosci 27, 12176–12189 (2007).
62. Ujfalussy, B. B. & Orbán, G. Sampling motion trajectories during hippocampal theta sequences. *eLife* **11**, e74058 (2022).
63. Mizuseki, K., Royer, S., Diba, K. & Buzsáki, G. Activity dynamics and behavioral correlates of CA3 and CA1 hippocampal pyramidal neurons. Hippocampus 22, 1659–1680 (20**12**).
64. Luo, X., Mok, R. M. & Love, B. C. The inevitability and superfluousness of cell types in spatial cognition. bioRxiv 2024.01.10.575026 (2024) doi:10.1101/2024.01.10.575026.
65. Schaeffer, R., Khona, M. & Fiete, I. R. No Free Lunch from Deep Learning in Neuroscience: A Case Study through Models of the Entorhinal-Hippocampal Circuit. bioRxiv 2022.08.07.503109 (2023) doi:10.1101/2022.08.07.503109.
66. Benna, M. K. & Fusi, S. Place cells may simply be memory cells: Memory compression leads to spatial tuning and history dependence. Proc. Natl. Acad. Sci. 118, e2018422118 (2021).
67. Zhang, C. et al. A Survey on Masked Autoencoder for Self-supervised Learning in Vision and Beyond. arXiv (2022) doi:10.48550/arxiv.2208.00173.
68. George, T. M. et al. RatInABox, a toolkit for modelling locomotion and neuronal activity in continuous environments. eLife **13**, e85274 (2024).
69. Itskov, V., Curto, C., Pastalkova, E. & Buzsáki, G. Cell assembly sequences arising from spike threshold adaptation keep track of time in the hippocampus. J Neurosci 31, 2828 2834 (2011).
70. Romani, S. & Tsodyks, M. Short-term plasticity based network model of place cells dynamics. Hippocampus **25**, 94–105 (2015).
71. Milstein, A. D., Tran, S., Ng, G. & Soltesz, I. Offline memory replay in recurrent neuronal networks emerges from constraints on online dynamics. *J. Physiol.* **601**, 3241–3264 (2023).
72. Levenstein, D., Buzsáki, G. & Rinzel, J. NREM sleep in the rodent neocortex and hippocampus reflects excitable dynamics. Nat Commun 10, 1 12 (2019).
73. Peyrache, A., Lacroix, M. M., Petersen, P. C. & Buzsáki, G. Internally organized mechanisms of the head direction sense. Nat Neurosci 18, 569 575 (2015).
74. Barker, G. R. I. & Warburton, E. C. When Is the Hippocampus Involved in Recognition Memory? J. Neurosci. 31, 10721–10731 (2011).
75. Han, T., Xie, W. & Zisserman, A. Video Representation Learning by Dense Predictive Coding. *Arxiv* (2019).
76. Silver, D. et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. Science **362**, 1140–1144 (2018).
77. Gloeckle, F., Idrissi, B. Y., Rozière, B., Lopez-Paz, D. & Synnaeve, G. Better & Faster Large Language Models via Multi-token Prediction. arXiv (2024).
78. Dragoi, G. & Buzsáki, G. Temporal Encoding of Place Sequences by Hippocampal Cell Assemblies. Neuron 50, 145 157 (2006).
79. Fang, C., Aronov, D., Abbott, L. & Mackevicius, E. L. Neural learning rules for generating flexible predictions and computing the successor representation. eLife 12, e80680 (2023).
80. Bono, J., Zannone, S., Pedrosa, V. & Clopath, C. Learning predictive cognitive maps with spiking neurons during behavior and replays. eLife 12, e80671 (2023).
81. George, T. M., Cothi, W. de, Stachenfeld, K. L. & Barry, C. Rapid learning of predictive maps with STDP and theta phase precession. eLife 12, e80663 (2023).
82. Uria, B. et al. The Spatial Memory Pipeline: a model of egocentric to allocentric understanding in mammalian brains. *Biorxiv* 2020.11.11.378141 (2020) doi:10.1101/2020.11.11.378141.
83. Whittington, J. C. R. et al. The Tolman-Eichenbaum Machine: Unifying Space and Relational Memory through Generalization in the Hippocampal Formation. Cell **183**, 1249-1263.e23 (2020).
84. George, D. et al. Clone-structured graph representations enable flexible learning and vicarious evaluation of cognitive maps. Nat Commun 12, 2392 (2021).
85. Raju, R. V., Guntupalli, J. S., Zhou, G., Lázaro-Gredilla, M. & George, D. Space is a latent sequence: Structured

sequence learning as a unified theory of representation in the hippocampus. arXiv (2022) doi:10.48550/arxiv.2212.01508.
86. Chen, Y., Zhang, H., Cameron, M. & Sejnowski, T. Predictive Sequence Learning in the Hippocampal Formation. bioRxiv 2022.05.19.492731 (2024) doi:10.1101/2022.05.19.492731.
87. Buzsáki, G. & Tingley, D. Space and Time: The Hippocampus as a Sequence Generator. *Trends Cogn Sci* **22**, 853 869 (2018).
88. Grion, N., Akrami, A., Zuo, Y., Stella, F. & Diamond, M. E. Coherence between Rat Sensorimotor System and Hippocampus Is Enhanced during Tactile Discrimination. *PLoS Biol.* **14**, e1002384 (2016).
89. Joshi, A. et al. Dynamic synchronization between hippocampal representations and stepping. Nature **617**, 125–131 (2023).
90. Macrides, F., Eichenbaum, H. & Forbes, W. Temporal relationship between sniffing and the limbic theta rhythm during odor discrimination reversal learning. J. Neurosci. 2, 1705–1717 (1982).
91. Jutras, M. J., Fries, P. & Buffalo, E. A. Oscillatory activity in the monkey hippocampus during visual exploration and memory formation. *Proc National Acad Sci* **110**, 13144 13149 (2013).
92. Drieu, C., Todorova, R. & Zugaro, M. Nested sequences of hippocampal assemblies during behavior support subsequent sleep replay. Science 362, 675 679 (2018).
93. Theodoni, P., Rovira, B., Wang, Y. & Roxin, A. Theta-modulation drives the emergence of connectivity patterns underlying replay in a network model of place cells. Elife 7, (2018).
94. Liu, C., Todorova, R., Tang, W., Oliva, A. & Fernandez-Ruiz, A. Associative and predictive hippocampal codes support memory-guided behaviors. Science **382**, eadi8237 (2023).
95. Hill, A. J. First occurrence of hippocampal spatial firing in a new environment. *Exp. Neurol.* **62**, 282–297 (1978).
96. Farooq, U. & Dragoi, G. Emergence of preconfigured and plastic time-compressed sequences in early postnatal development. Science **363**, 168–173 (2019).
97. Muessig, L., Lasek, M., Varsavsky, I., Cacucci, F. & Wills, T. J. Coordinated Emergence of Hippocampal Replay and Theta Sequences during Post-natal Development. Curr. Biol. 29, 834-840.e4 (2019).
98. Sargolini, F. et al. Conjunctive Representation of Position, Direction, and Velocity in Entorhinal Cortex. Science **312**, 758–762 (2006).
99. Viejo, G. & Peyrache, A. Precise coupling of the thalamic head-direction system to hippocampal ripples. Nat. Commun. 11, 2524 (2020).
100. Ikegaya, Y. et al. Interpyramid spike transmission stabilizes the sparseness of recurrent network activity. *Cereb Cortex* **23**, 293 304 (2013).
101. Dragoi, G. & Tonegawa, S. Preplay of future place cell sequences by hippocampal cellular assemblies. Nature **469**, 397–401 (2011).
102. Dragoi, G. The generative grammar of the brain: a critique of internally generated representations. Nat. Rev. Neurosci. 25, 60–75 (2024).
103. Teramae, J. & Fukai, T. Computational implications of lognormally distributed synaptic weights. *P Ieee* **102**, (2014).
104. Farzanfar, D., Spiers, H. J., Moscovitch, M. & Rosenbaum, R. S. From cognitive maps to spatial schemas. Nat Rev Neurosci 1–17 (2022) doi:10.1038/s41583-022-00655-9.
105. Bittner, K. C., Milstein, A. D., Grienberger, C., Romani, S. & Magee, J. C. Behavioral time scale synaptic plasticity underlies CA1 place fields. Science **357**, 1033 1036 (2017).
106. Fernández-Ruiz, A. et al. Entorhinal-CA3 Dual-Input Control of Spike Timing in the Hippocampus by Theta-Gamma Coupling. Neuron 93, 1213 1226.e5 (2017).
107. Grienberger, C. & Magee, J. C. Entorhinal cortex directs learning-related changes in CA1 representations. Nature 611, 554–562 (2022).
108. Li, Y., Briguglio, J. J., Romani, S. & Magee, J. C. Mechanisms of memory storage and retrieval in hippocampal area CA3. bioRxiv 2023.05.30.542781 (2023) doi:10.1101/2023.05.30.542781.
109. Keller, G. B. & Mrsic-Flogel, T. D. Predictive Processing: A Canonical Cortical Computation. Neuron 100, 424 435 (2018).
110. Bakhtiari, S., Mineault, P., Lillicrap, T., Pack, C. C. & Richards, B. A. The functional specialization of visual cortex emerges from training parallel pathways with self-supervised predictive learning. Biorxiv 2021.06.18.448989 (2021) doi:10.1101/2021.06.18.448989.
111. McClelland, J. L., McNaughton, B. L. & O'Reilly, R. C. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and

memory. *Psychol Rev* **102**, 419 (1995).
112. Teyler, T. J. & DiScenna, P. The hippocampal memory indexing theory. *Behav Neurosci* **100**, 147 (1986).