Spatially embedded recurrent neural networks reveal widespread links between structural and functional neuroscience findings

Jascha Achterberg [1,5] ✉, Danyal Akarca [1,5] ✉, D. J. Strouse[2], John Duncan[1,3,6] & Duncan E. Astle [1,4,6]

Brain networks exist within the confines of resource limitations. As a result, a brain network must overcome the metabolic costs of growing and sustaining the network within its physical space, while simultaneously implementing its required information processing. Here, to observe the effect of these processes, we introduce the spatially embedded recurrent neural network (seRNN). seRNNs learn basic task-related inferences while existing within a three-dimensional Euclidean space, where the communication of constituent neurons is constrained by a sparse connectome. We find that seRNNs converge on structural and functional features that are also commonly found in primate cerebral cortices. Specifically, they converge on solving inferences using modular small-world networks, in which functionally similar units spatially configure themselves to utilize an energetically efficient mixed-selective code. Because these features emerge in unison, seRNNs reveal how many common structural and functional brain motifs are strongly intertwined and can be attributed to basic biological optimization processes. seRNNs incorporate biophysical constraints within a fully artificial system and can serve as a bridge between structural and functional research communities to move neuroscientific understanding forwards.

As they develop, brain networks learn to achieve objectives, from simple functions such as autonomic regulation, to higher-order processes such as solving problems. Many stereotypical features of networks are downstream consequences of resolving challenges and trade-offs they face, across their lifetime[1,2] and evolution[3–5]. One example is the optimization of functionality within resource constraints; all brain networks must overcome metabolic costs to grow and sustain the network in physical space, while simultaneously optimizing that network for information processing. This trade-off shapes all brains within and across species, meaning it could be why many brains converge on similar organizational solutions[4]. As such, the most basic features of both brain organization and network function—such as its sparse and small-world structure, functional modularity, and characteristic neuronal tuning curves—might arise because of this basic optimization problem.

[1]MRC Cognition and Brain Sciences Unit, University of Cambridge, Cambridge, UK. [2]Google DeepMind, London, UK. [3]Department of Experimental Psychology, University of Oxford, Oxford, UK. [4]Department of Psychiatry, University of Cambridge, Cambridge, UK. [5]These authors contributed equally: Jascha Achterberg, Danyal Akarca. [6]These authors jointly supervised this work: John Duncan, Duncan E. Astle. ✉e-mail: jascha.achterberg@mrc-cbu.cam.ac.uk; danyal.akarca@mrc-cbu.cam.ac.uk
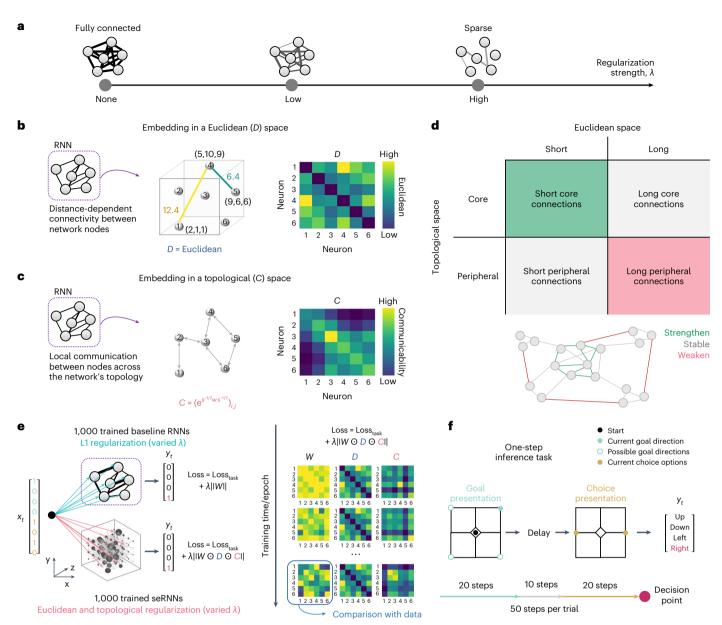
Fig. 1 | Task structure and seRNNs. a, We use regularization to influence network structure during training to promote smaller network weights and hence a sparser connectome. **b**, Through regularization, we embed RNNs in Euclidean space by assigning units a location on an even $5 \times 5 \times 4$ grid. We show a schematic of a six-node network in its space. **c**, We similarly embed RNNs in a topological space, guiding the pruning process towards efficient intra-network communication operationalized by a weighted communicability measure (see main text). The weighted communicability term is shown for the same network. **d**, When these constraints are placed within a joint regularization term, networks are incentivized to strengthen short connections, which are core to the networks topological structure, and weaken long connections, which are peripheral. Networks are generally incentivized to weaken connections while optimizing task performance. **e**, In the main study, we trained 1,000 L1-regularized RNNs as a baseline. L1 networks optimize task performance while minimizing the strength of their absolute weights (W). The network receives task inputs from an eight-unit-wide fully connected feed-forward layer and represents its choice as one of four choice units in the output layer. We compare these with 1,000 seRNNs, which include both Euclidean and topological constraints in their regularization term, by multiplying the weight matrix (W) by its Euclidean distance (D) and weighted communicability (C). Elements of the resulting matrix are summed, forming the structural loss. We minimize the sum of the task loss and the structural loss. To the right, we show the evolution of W, D and C matrices over training. **f**, Networks solve a one-step inference task starting with a period of twenty steps where the goal is presented in one of four locations on a grid: top/bottom, left/right (depicted in light blue). Subsequently, there is a ten-step delay where the goal location must be memorized. Then two choice options are provided for twenty steps. Using prior goal information, agents must choose the option closer to the goal. In this example, given left and right options, the correct decision is to select right.

Our understanding of how the brain's structure and function interact largely comes from observing differences in brain structure, such as across individuals[6] or following brain injury[7], and then systematically linking these differences to brain function or behavioural outcomes. But how do these relationships between structure, function and behaviour emerge in the first place? To address this question, we need to be able to manipulate experimentally how neural networks form, as they learn to achieve behavioural objectives, to establish the causality of these relationships. Computational models allow us to do this[8]. They have shown that network modularity can arise through the spatial cost
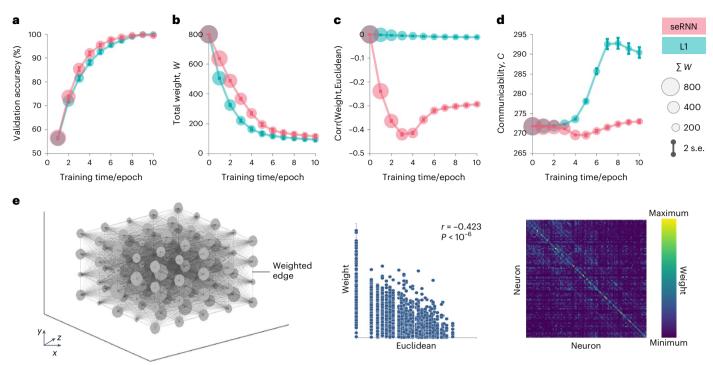
Fig. 2 | Validating the training of seRNNs. a, The validation accuracy of all converging neural networks is shown across L1 RNNs ($n = 479$, blue, for all plots) and seRNNs ($n = 390$, pink, for all plots), showing that equivalent performance is achieved on the one-step inference task. For all plots, error bars correspond to two standard errors. **b**, At the same time, both groups of networks show a general trend of weakening the weights in their recurrent layer, showing that the overall regularization is working in both groups of networks. **c**, As a result of their unique regularization function, seRNNs have a negative correlation between weight and Euclidean distance over the course of epochs/training, but in L1 networks there is no relationship between weights and Euclidean distance. **d**, The regularization function of seRNNs also successfully influences the topology of networks to prefer topologically central weights over topologically peripheral weights, as shown by lower weighted communicability values. **e**, Left: an example of a representative seRNN network in the 3D space in which it was trained. The size of the nodes reflects their node strength. This network was taken from epoch 9 at a regularization of 0.08 and is the network used for visualizations for the rest of this paper. Middle: we show the negative relationship between the connection weights of seRNN versus the Euclidean distances of the connections. Pearson's correlation coefficient is provided, with the corresponding P value ($P = 7.03 \times 10^{-7}$). No adjustments were required for multiple comparisons. Right: we show the weight matrix of this seRNN, showing how weights are patterned throughout the network.

of growing a network[9], how orthogonal population dynamics can arise purely through optimizing task performance[10] and how predictive coding can arise through limiting a brain's energy usage[11]. But we have yet to incorporate both the brain's anatomy and the brain's function into a single coherent model, allowing a network to dynamically trade-off its different structural, functional and behavioural objectives in real time.

To achieve this, we introduce spatially embedded recurrent neural networks (seRNNs). An seRNN is optimized to solve a task, making decisions to achieve functional goals. However, as it learns to achieve these goals and to optimize its behavioural performance, its constituent neurons face the kind of resource constraints experienced within biological networks. Neurons must balance their finite resources to grow or prune connections, while the cost of a connection is proportional to its length in three-dimensional (3D) Euclidean space[12–16]. At the same time, the network attempts to optimize its intra-network communication to allow for efficient propagation of signals[17–21]. By allowing seRNNs to dynamically manage both their structural and functional objectives simultaneously, while they learn to behave, multiple simple and complex hallmarks of biological brains naturally emerge.

Results

Spatially embedded recurrent neural networks

Our first goal was to create a supervised optimization process that subjects recurrent neural networks (RNNs; 'RNN modelling' in Methods) to the constraints of biophysical space while they are optimized for task performance. An established way of influencing a network's weight matrix while it is optimized for task performance is regularization (Fig. 1a).

In regularization, instead of merely optimizing a network's weights to maximize task performance, one adds an additional regularization term to the optimizer to minimize the strength of a network's weights. This is related to regularized regression, such as L1 (LASSO) regression, where the sum of the absolute beta weights is minimized to improve a model's out-of-sample prediction performance. We use the same idea to spatially embed an RNN. We start with fully connected RNNs and while they are trained to maximize task performance, we nudge them to minimize weights that are long in 3D space. To achieve this, we assign every unit in the RNN's recurrent layer a location in 3D space (Fig. 1b) and regularize a weight more strongly if it belongs to two units that are far apart in Euclidean space. In this pruning process, we also want the network to optimize within-network communication, meaning a weight should be more readily pruned if it does not contribute strongly to the propagation of signals within the network. A standard measure of signal propagation in a (binary) network is communicability, reflecting the shortest routes between all pairs of nodes[22] (Fig. 1c; see details in 'Communicability' in Supplementary Information). When adapted for a weighted network (weighted communicability[19]), the communicability value of a network is low when there are strong global core connections supporting short paths across the network while avoiding redundant peripheral connections to achieve sparsity (Fig. 1d). In Supplementary Information ('Minimizing redundant connectivity by minimizing weighted communicability'), we provide information on how weighted communicability differentially optimizes peripheral and core connection strengths. By combining the spatial distance and weighted communicability terms in an RNN's regularization while it
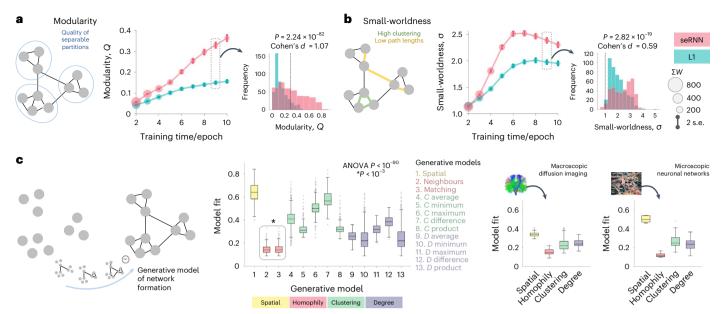
Fig. 3 | seRNNs show a brain-like structural topology. a, Left: a schematic illustration of the concept of modularity in networks. While both L1 ($n = 479$) and seRNN ($n = 390$) networks show increasing modularity over epochs/training, there is a consistently greater modularity in seRNNs compared with L1 networks. Error bars correspond to two standard errors. Right: we show very large (Cohen's $d = 1.07$) statistical differences in modularity distributions for functioning (validation accuracy ≥90%) epoch 9 networks in L1 and seRNN networks. A two-sample t-test was taken to provide the P value. No adjustments were required for multiple comparisons. **b**, Left: a schematic illustration of the concept of small-worldness in networks. While both L1 ($n = 479$) and seRNN ($n = 390$) networks show a similar trajectory shape of small-worldness over epochs/training, there is a consistently greater small-worldness in seRNNs compared with L1 networks. Error bars correspond to two standard errors. Right: we show moderate-to-large (Cohen's $d = 0.59$) statistical differences in small-worldness distributions for functioning epoch 9 networks in L1 and seRNN networks.

A two-sample t-test was taken to provide the P value. No adjustments were required for multiple comparisons. **c**, For a range of generative network models ('Generative network modelling' in Methods), we present the model fit of the top performing simulations fit to seRNNs ($n = 390$). Note that the lower the model fit, the better the performance, as the model fit function is a measure of dissimilarity between the RNN and the generative simulation. The results show that homophily models achieve the best model fits. These findings are congruent with published data from adolescent whole-brain diffusion-MRI structural connectomes[35] (middle right) and high-density functional neuronal networks at single-cell resolution[15] (right). The boxplots present the minimum value (bottom), maximum value (top), median value (centre) and the interquartile range (bounded 25th and 75th percentile). A one-way ANOVA was taken to provide the first P value ($P = 1.04 \times 10^{-91}$), followed by a Tukey's test for pairwise comparisons in which homophily models had a pairwise P value $<10^{-3}$ for all comparisons.

learns to solve a task, we arrive at seRNNs (Fig. 1e). We provide a detailed walkthrough of the regularization function in 'seRNN regularization function' in Methods. While learning to solve a task, seRNNs are nudged to prefer short core weights over long peripheral weights.

To understand how this spatial embedding impacts a network's structure and function, we set up 2,000 RNNs. Half of the networks were seRNNs trained with the new optimization process described above. The other half were regular RNNs regularized with a standard L1 regularizer minimizing the sum of the absolute weights, to arrive at a population of baseline networks that match seRNNs in overall connectivity strength. In both cases, the regularizer was applied to the hidden recurrent layer of the network and the regularization strength was systematically varied within each subgroup of networks to cover a wide spectrum of regularization strength that is matched across subgroups (Fig. 1e and 'Regularization strength set-up and network selection' in Methods). All networks had 100 units in their hidden layer and were trained for 10 epochs. All networks started strongly connected and learned through pruning weights in accordance with their regularization. We trained networks on a one-choice inference task that required networks to develop two fundamental cognitive functions of recurrent networks: remembering task information ('goal') and integrating it with new incoming information ('choices') (Fig. 1f and 'Task paradigm' in Methods).

When training the networks, we found that both types of network manage to learn the task with high accuracy (Fig. 2a). Focusing on networks that successfully solve the task (>90% task accuracy; $n = 390$

for seRNNs, $n = 479$ for L1s; see 'Regularization strength set-up and network selection' in Methods for discussion of network numbers), we first validated that our optimization process is working. By using L1 networks as a baseline, we observed that both groups decrease in their average connectivity strength (Fig. 2b) but that only seRNNs did so by pruning long-distance connections (Fig. 2c). This is commonly found in empirical brain networks across species and scales[23]. In addition, we validate that seRNNs successfully focus their pruning process on weights that are less important for the network's communicative structure, as represented by lower weighted communicability (Fig. 2d). Figure 2e shows an example visualization of one seRNN.

Having shown that the new regularization function in seRNNs has the expected effects on the weight matrix of networks, we next tested which features result from the spatial embedding. Specifically, we tested whether seRNNs show features commonly observed in primate cerebral cortices, including structural motifs such as modularity[24–26] and small-worldness[27,28], before testing for functional clustering of units in space[27,28]. We then go beyond structural and functional organization and test whether spatial embedding forces networks to implement an energy-efficient mixed-selective code[29,30]. In short, we wanted to test whether established organization properties of complex brain networks arise when we impose local biophysical constraints.

Modular small-world networks emerge from constraints

We first investigated two key topological characteristics that are commonly found in empirical brain networks across spatial scales

and proposed to facilitate brain function: modularity[24-26] and small-worldness[27,28]. Modularity denotes dense intrinsic connectivity within a module but sparse weak extrinsic connections between modules and small-worldness indicates a short average path length between all node pairs, with high local clustering.

Computing modularity Q statistics and small-worldness ('Topological analysis' in Methods) shows that seRNNs consistently show both increased modularity (Fig. 3a) and small-worldness (Fig. 3b) relative to L1 networks over the course of training. Differences are smaller initially, but later in training, the effect size for differences in modularity are large (at epoch 9, modularity $P = 2.24 \times 10^{-82}$, Cohen's $d = 1.07$; Fig. 3a, right) and for small-worldness moderate to large ($P = 2.82 \times 10^{-19}$, Cohen's $d = 0.59$; Fig. 3b, right). seRNNs achieve modularity Q statistics within ranges commonly found in empirical human cortical networks[31]. Both L1 and seRNNs achieve the technical definition of small-worldness of >1 (ref. 32), but seRNNs show a higher value more consistent with empirical networks[33]. 'Replication across architectures' in Supplementary Information shows how the subparts of the regularization interact with the task optimization to shape these structural effects. It is important to note that within the population of seRNNs, we find varying degrees of modularity and small-worldness (Fig. 3a, right, and Fig. 3b, right). We will return to this variability in a later section.

To further validate the structural likeness of seRNNs to empirical neural connectivity, we used generative network models[9,34-36]. These models elucidate which topological wiring rules can accurately approximate observed neural graphs. Corroborating empirical macro- and microscopic data[15,35], we find that homophily wiring rules—where neurons preferentially form connections to other neurons that are self-similar in their connectivity profiles—perform best in approximating the topology of seRNNs relative to all other wiring rules (Fig. 3c and additional detail in 'Generative network modelling of RNNs' in Supplementary Information).

Functionally related units spatially organize in seRNNs

So far, we have explored how imposing biophysical constraints within seRNNs produces structures that mimic observed networks. However, this ignores the functional roles of neurons or their patterning within the network. We next examined this by exploring the configuration of functionally related neurons in 3D space (Fig. 4a). In brain networks, neurons sharing a tuning profile to a stimulus tend to spatially group[37,38]. This can be observed on fine-grained cortical surfaces with preferences for stimuli features[39] (Fig. 4b) and in whole-brain functional connectivity forming modular network patterns[40] (Fig. 4c). In addition, high-resolution recordings in rodents show how the brain keeps many codes localized but also distributes some across the network[41]. To test whether seRNNs recapitulate functional co-localization, we decoded how much variance of unit activity can be explained by the goal location or choice options, over the course of each trial ('Decoding' in Methods). In Fig. 4d, we show a visualization in a representative network and unit-specific preferences over the course of a single trial.
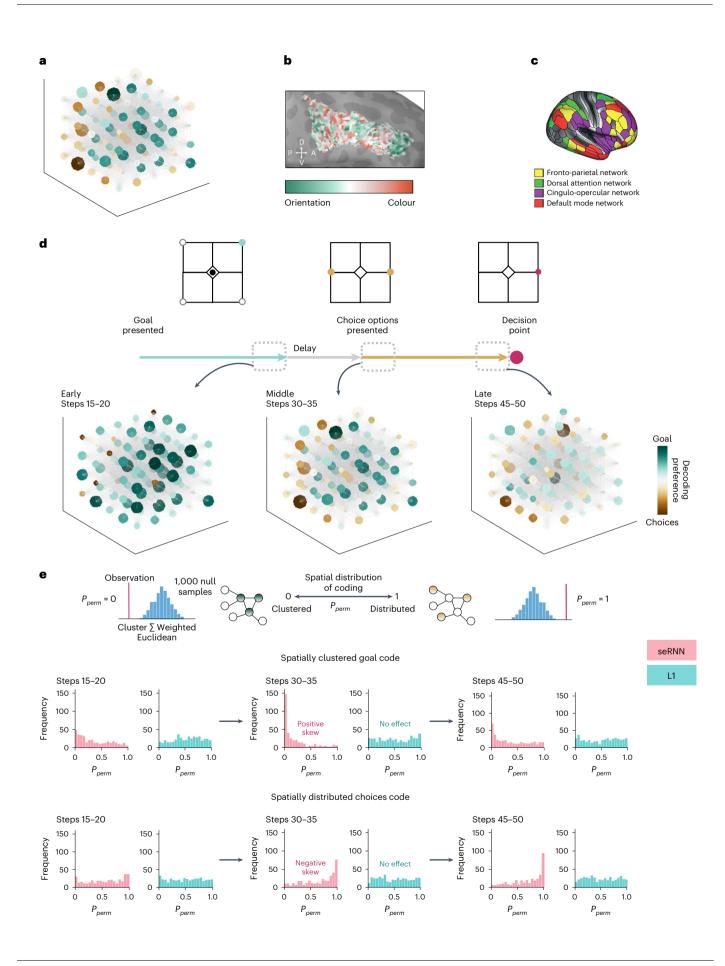
By taking the relative preference for goal versus choice for each unit, we tested whether the relative sensitivity to stimuli was concentrated in parts of the network. We used a spatial permutation test ('Spatial permutation test' in Methods) to test whether the Euclidean distance between highly 'goal' or 'choice' selective neurons was significantly less or more than would be expected by chance. A small P_{perm} value highlights that functionally similar neurons tend to be significantly clustered in space whereas a large P_{perm} corresponds to functionally similar neurons being distributed in space (Fig. 4e, top).
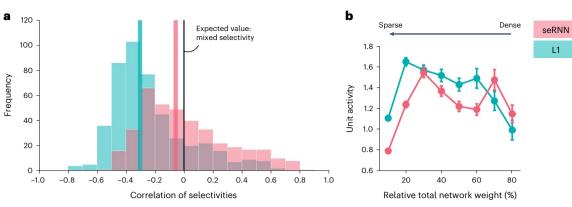
We tested for functional co-localization across three time windows of the trial (the total duration of a trial was 50 steps; Fig. 1e): (1) early stage (goal presented, steps 15–20); (2) middle stage (choice options presented, steps 30–35) and (3) late stage (decision point, steps 45–50). At the early stage, when only goal information is presented, neurons code for only the goal information (widespread dark green nodes in Fig. 4d, left). In seRNNs, there is a slight positive skew in P_{perm} values, suggesting clustering of highly goal-coding neurons (Fig. 4e, middle left). Subsequently, in the middle stage, when choice options are first shown, goal information clusters within a concentrated area of space, leaving the choice information distributed (seen by clustering of green nodes and distribution of brown nodes in Fig. 4d, middle). This is highlighted by a large positive skew in P_{perm} values for the goal in seRNN networks (Fig. 4e, middle top) and correspondingly the opposite for choice information (Fig. 4e, middle bottom). In the late stage, the clustering of goal information in space dissipates such that by the time a decision must be made, the goal information has now spread out more but still retains some clustering (Fig. 4e, middle right). The choice code remains distributed (Fig. 4e, bottom right). This suggests that seRNNs use their highly modular structure to keep a connected core with goal information, which needs to be retained across the trial. It uses spatially proximal units to form this core. The presented choices information is then represented by units outside this core and dynamically integrates with the information in the core as the decision point approaches. These findings are unique to seRNNs, as L1 P_{perm} values remain uniform, indicative of no functional organization. The control analysis in Supplementary Fig. 12 shows these findings hold true when variables are treated independently instead of relatively.
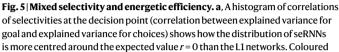
Mixed selectivity and energy-efficient coding

So far, we have shown that adding spatial constraints to a network gives rise to patterns of network connectivity that are highly reminiscent of observed biological networks. Nodes functionally co-localize and the spatial embedding causes differences in how they code task-relevant information. This selectivity profile has been widely studied. Studies show that neurons in task-positive brain regions tend to show a mixed selectivity profile, meaning that neurons do not only code for a single

Fig. 4 | Functional clustering and distribution of coding in space. a, An example of a representative seRNN network. The colour of the nodes relates to the decoding preference of that neuron, where a preference for goal information is represented by green and choices information by brown. **b**, The spatial clustering of neuronal ensembles that are preferentially tuned for orientation versus colour in human prefrontal cortex. The Dorsal-Ventral (D-V) and Anterior-Posterior (A-P) axes are shown[39]. **c**, The macroscopic spatial organization of functional networks[40]. **d**, We show decoding of units for goal (green) versus choice (brown) information at different points in the trial, within the representative seRNN network. **e**, A schematic illustration of the spatial permutation test for determining whether the neurons are functionally clustered (top left) or distributed (top right) in space. For this permutation test, we compute the summed Euclidean distance between units with an observed preference for goal or choice information, respectively, weighted by the magnitude of their preference (termed cluster ∑ weighted Euclidean). This gives a statistic, for every network, corresponding to the weighted distance between units (that is, goal or choice units) in space. To determine whether this statistic

was equivalent to chance, for each statistic we computed a null distribution of expected distances between goal and choice units, respectively, under the assumption that they are randomly located in space. This was calculated by taking 1,000 random samples of the same size as the number of empirical neurons with a preference for goal or choice information. The P_{perm} relates to where the statistic sits within this null distribution, where each network gets a P_{perm} for goal and choice information. The skew of the P_{perm} towards zero shows that the code of networks is more clustered than the null distribution whereas a skew towards one highlights a more distributed code. The P_{perm} values across RNNs are given for goal information (middle) and choice information (bottom) for seRNNs (pink) and L1 networks (blue). Goal information is shown to be clustered, as given by the positively skewed P_{perm} distributions. In contrast, choice information is shown to be distributed. No adjustments were required for multiple comparisons. Panel **b** reproduced with permission from ref. 39, under a Creative Commons licence CC BY 4.0. Panel **c** adapted with permission from ref. 40, Elsevier.

Fig. 5 | Mixed selectivity and energetic efficiency. a, A histogram of correlations of selectivities at the decision point (correlation between explained variance for goal and explained variance for choices) shows how the distribution of seRNNs is more centred around the expected value $r = 0$ than the L1 networks. Coloured lines mark the median of the distribution. The expected value corresponds to a fully mixed-selective code. **b,** seRNN ($n = 390$) networks spend less energy on unit activations than L1 ($n = 479$) networks, which are matched for mean weight strength in the recurrent layer. Error bars correspond to two standard errors.

task variable but instead a mixture of them[30,42–44]. A mixed-selective code is assumed to allow networks to solve complex tasks by increasing the decodability of information from the network's neurons[29,45]. There are many ways to quantify selectivity profiles[46]. One simple method is to calculate the correlation of explained variances of task variables across the population of neurons. These are expected to be uncorrelated, implying a neutrally mixed code where a neuron's coding preference for one variable does not predict its code for another variable. In single-unit recordings, correlations can be close to zero or sometimes slightly positive[47].

We looked at the correlation of selectivities of trained networks (epoch 9) for the goal and choices variables. At the time in the trial when networks make a choice, the median correlation is $r = -0.057$ for seRNN but $r = -0.303$ for L1, showing that L1 networks produce an anticorrelated code while seRNNs have a more mixed-selective code (Fig. 5a). It is possible that this effect is driven by the differential connectome structure of the two groups of networks. While a modular and separated network would not automatically mix codes across variables evenly, we find a well-mixed code in seRNNs. The additional highly communicative connections between modules of the small-worldness characteristic might help seRNNs to organize units in space while retaining a mixed code across the population. 'Mixed selectivity' in Supplementary Information shows how networks specifically show a mixed-selective code at the time when the decision is made. Like our structural results, we saw that there is variation across the population of networks (Fig. 5a), where some networks fall neatly on $r = 0$ and others might show correlated codes. The following section provides an analysis of this variance.

The choice of a neuronal code in populations of neurons is strongly linked to the question of energy demand. As the firing of action potentials uses a substantial amount of energy[48], a population of neurons should choose a code with a good trade-off of metabolic cost and information capacity[29]. To test our networks' energy consumption, we calculated the mean activation of each unit in a network's recurrent layer (epoch 9) during the period of information integration (after onset of choices). Then we tested for the difference between seRNNs and L1 networks, controlling for the effect of the average weight strength in the recurrent layer (Fig. 5b). Across most weight strengths, seRNNs showed significantly lower unit activations compared with L1 networks ($P < 0.001$, $t(86,497) = 21.4$, 95% confidence interval = $[-0.271, -0.226]$). Sustaining a mixed-selective code at the time of choice might help downstream integration units to decode information more easily, with fewer unit activations needed to communicate the correct choice. This effect disappears for networks with higher average weights, with weak regularization and hence weaker spatial embedding.

Constraints cause linked brain-like structure and function

So far, we have seen that seRNNs show a collection of features that are commonly observed in brains but have not previously been related. The caveat not addressed so far is that for any feature we observed in seRNNs, we also see strong variation across the population of networks (for example, Fig. 3b for modularity or Fig. 5a for mixed selectivity). This opens the possibility that these features do not arise in parallel in seRNNs but instead each feature could emerge in its unique subgroup of networks. This would be unlike biological brains, which exist in a critical sweet-spot area[49] where all the features described in this paper are observed. In this section, we tested whether all seRNN features co-appear in a similar subset of trained networks, defined by a unique combination of training parameters.

To study the co-occurrence of brain features in seRNNs, we looked at the distribution of feature magnitude across the space of training parameters (regularization strength, number of training epochs passed). Figure 6a shows matrix plots for accuracy (left), total sum of weights (middle left), modularity (middle right) and small-worldness (right) across the entire spectrum of training epochs (x axis) and regularization strengths (y axis). As before, there is variation in the magnitude of features across the population of networks, but now we also see that this variation is structured. Brain-like topology emerges in a sweet-spot of low to medium strength regularization and during the later training epochs (pink box). The schematic in Fig. 6b highlights this space of sparse, highly accurate, modular small-world networks with an example network showing all properties (Fig. 6b, middle right). Above this space (that is, networks with less regularization, highlighted in orange) networks can solve the task and show small-worldness, but remain very dense and lack the modular organization found in empirical brain networks. Below this space (that is, networks with more regularization, highlighted in light blue) networks show extreme sparsity and modularity, but fail to functionally converge on the task and they lose their small-world topology.

Next, we wanted to look at the same 'sweet spot' in terms of the network's functional properties. As the decoding required us to focus this analysis on networks with high task performance ('Decoding' in Methods), we use networks with an accuracy >90% at epoch 9. Figure 6c shows the functional results across regularization strengths, highlighting the sweet spot of regularization from Fig. 6a with the pink box. In the first two plots from the top, we show two structural metrics (sparsity and short connection preference). We observed the same distribution when looking at the homophily generative wiring rule (Supplementary Fig. 11b). Looking at mixed selectivity (Fig. 6c,
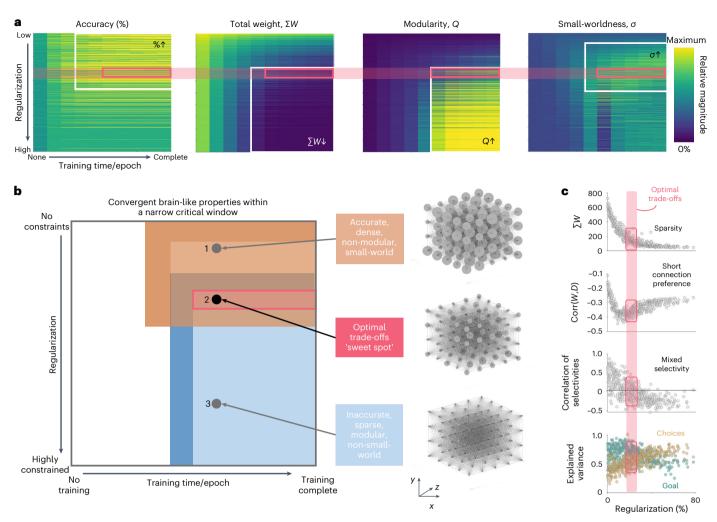
Fig. 6 | The seRNN parameter space converges on brain-like topology and function. a, The white borders within the regularization-training parameter space delineate the conditions where seRNNs achieve robust accuracy (left), sparse connectivity (middle left), modular networks (middle right) and small-worldness (right). The pink box shows where all these findings can be found simultaneously. The colour of the matrix corresponds to the relative magnitude of the statistic compared with the maximum. **b**, This is further highlighted by a schematic representation, which shows the space of possible seRNNs. The pink box shows the overlap of all findings, where accurate, sparse, modular, small-world networks are generated, which we term as being at the optimal trade-off. Networks 1, 2 and 3 each represent example networks across the space. The nodes of the representative graph reflect the node's strength, defined as the total sum of the node's in- and out-connection weights. **c**, In this pink window, networks are sparse (top), prefer short connections (middle top), have a correlations of variable selectivities centring around zero, consistent with mixed selectivity (middle bottom) and have equivalent explained variance for both the goal and the choice (bottom).

third from top), our analyses revealed that networks show a mixed-selective code at the decision point in the sweet-spot window identified before. Units here show a balanced code with information for both goal and choices (Fig. 6c, bottom), whereas very dense or sparse networks show a preference for either goal or choices information. As such, the density and related modular small-world structure influences the time horizon of information flowing through the network. Dense networks show greater focus on past information, which resonates with how functional networks reconfigure to support memory[50]. Supplementary Fig. 14 shows a correlation matrix showing pairwise relationships between features studied here.

Our findings show that there is a critical parameter window in which both structural and functional brain features jointly emerge in seRNNs. Brains are often said to live in a unique but critical niche where all characteristics needed to support their function can exist in parallel[51]. seRNNs show the same preference for a critical parameter window but also give us the ability to study networks on their way to converging on brain-like characteristics in this critical window.

Discussion

Functioning brains have key organizational features endowing them with computational capacities to perform a broad range of cognitive operations efficiently and flexibly. These include sparse connectivity with a modular small-world structure[25,27,52], generatable via homophilic wiring rules[34–36], with spatially configured functional units that implement a mixed-selective code[30,45] and minimize energy expenditure[29,48]. We argue that these complex hallmarks can be, at least in part, attributed to three forces impacting virtually any brain network: optimization of functional performance in a (task) environment, metabolic/structural costs of the network and signal communication within the network. In this work we have shown that seRNNs allow us to manipulate these optimization goals, demonstrating that seemingly unrelated neuroscientific findings can emerge in unison and appear to have a strong co-dependence. We believe that these findings also have an impact on how we think about the interlinked structural and functional optimization processes in the brain under economic constrains ('Network economics in structural and functional neuroscience models' in Supplementary Discussion).

Our model provides an important tool to continue the work on jointly studying structure and function in neuroscience models[53–57]. In addition, our results are relevant for developments on the intersection of neuroscience and artificial intelligence (NeuroAI[58]) ('Implications of seRNN findings on artificial intelligence') in Supplementary Discussion.

There are many areas that we wish to improve on with future research. Principally, our models did not include a substantial amount of biological detail that, while inevitably critical for neuronal functioning, does not speak to the observations we aimed to recapitulate in the present study. Implementing such details including molecular mechanisms guiding circuit development[59] or heterogeneous spiking of neurons[60] will probably provide insights into the trade-offs specific to biological brains. The addition of such details will help us expand the applicability of our models to explore the effect of developmental time courses[61,62], functional brain specialization[63] and how network variability may underpin individual differences[64]. Beyond these biological details, it will be important to see how different functional goals would have differential effects on structural optimization processes. The simple working memory task used here provides a first realistic cognitive challenge, but it will be interesting to consider seRNNs in continuous choice multi-task environments. Finally, it is unknown to what extent the inclusion of biophysical constraints has on the randomness of network structure, although we speculate it would generate less-random network structures, compared with regular task-optimized networks.

The development of seRNNs allowed us to observe the impact of optimizing task control, structural cost and network communication in a model system that can dynamically trade off its structural and functional objectives. This suggests that providing artificial neural networks with a topophysical structure[65,66] can enhance our ability to directly link computational models of neural structure and function. We believe that the modelling approach shown to work in seRNNs will speed up innovations in neuroscience by allowing us to systematically study the relationships between features that all have been individually discussed to be of high importance to the brain.

Methods

seRNN regularization function

In a canonical supervised RNN, all the network's trainable parameters are optimized to minimize the difference between the predicted value and correct value. To achieve this, we define a task loss function (L), which defines the prediction error to be minimized to optimize task performance. To produce a network that generalizes well to unseen data, we can add a regularization term. Regularization incentivizes networks to converge on sparse solutions and is commonly applied to neural networks in general[67] and neuroscientific network models[68,69]. For a regularized network, the loss function becomes a combination of both the task loss and the regularization loss. One example of a commonly applied regularization is the L1 regularization, which is also used in LASSO regression[70] and incentivizes the network to maximize task performance while concurrently minimizing the sum of all absolute weights in the neural network. If we want to regularize the recurrent weight matrix (W) with the dimensions $m \times m$, where m is number of units in the recurrent layer, the loss function would be:

$$L = L_{\text{Task}} + \text{L1} = L_{\text{Task}} + \gamma \, ||W|| \tag{1}$$

$$||W|| = \sum_{i=1}^{m} \sum_{j=1}^{m} |w_{i,j}| \tag{2}$$

An RNN with this loss function would learn to solve the task with a sparse weight matrix ($w_{i,j}$), where γ would determine the extent to which the network is forced to converge on a sparse solution. This parameter is called the regularization strength.

Unlike regular RNNs, real brain networks are embedded in a physical space[12–14]. To simulate the pressures caused by existing in a biophysical space, we manipulated the regularization term. We hypothesized that by incorporating constraints that appear common to any biological neural system, we could test whether these local constraints are sufficient to drive a network architecture that more closely resembles observed brain networks. Specifically, we included spatial constraints in two forms—Euclidean and network communication—that we argue are integral to any realistic neural network. To implement this, we first embed units within a 3D space, such that each unit has a corresponding x, y and z coordinate. Using these coordinates, we can generate a Euclidean distance matrix that describes the physical distance between each pair of nodes (Fig. 1b). This allows to minimize weights multiplied by their Euclidean distance ($d_{i,j}$), thereby incentivizing the network to minimize (costly) long-distance connections. The element-wise matrix multiplication is denoted with the Hadamard product $\odot$. Adding this to our optimization term gives us:

$$L = L_{\text{Task}} + \gamma \, ||W \odot D|| \tag{3}$$

$$||W \odot D|| = \sum_{i=1}^{m} \sum_{j=1}^{m} |w_{i,j}| \times |d_{i,j}| \tag{4}$$

The above formalization provides a spatial context for RNN training. In a next step, we want to follow the same approach to incentivize networks to preferably prune weights that are not strongly contributing to the within-network communication structure. We can impose this influence of communication via a weighted communicability term[19,22], which computes the extent to which, under a particular network topology, any two nodes are likely to communicate both directly and indirectly over time (Fig. 1c). Now taking this topological communication into account, we get the following loss function:

$$L = L_{\text{Task}} + \gamma \, ||W \odot D \odot C|| \tag{5}$$

$$C = e^{S^{-\frac{1}{2}} |W| S^{-\frac{1}{2}}} \tag{6}$$

Supplementary Figs. 1–5 provide a walkthrough explanation of how this term works and expand on the logic of how constraining the network's topology can serve as a prior for intra-network communication in sparse networks. Supplementary Fig. 6 specifically highlights the role that communicability has within the network optimization process. Note that in equation (6), S is a diagonal matrix with the degree of unit i ($\deg_i$) on the diagonal (that is, the node strength), which simply acts as a normalization term preventing any one single edge having undue influence[19]. This is explained in Supplementary Figs. 4 and 5.

Importantly, as all terms (W, D, C) are element-wise multiplied within the regularization term, they are all minimized as part of the training process. Note, it is possible, in principle, to parameterize each part of the above equation to vary the extent to which each term influences network outcomes. However, in this work, we focus on establishing the role of all in tandem. Future work could look to establish models with greater parameterization to establish optimal relative magnitudes.

Task paradigm

The task that networks are presented with is a one-choice inference task requiring networks to remember and integrate information (Fig. 1f). On an abstract level, networks needed to first store a stimulus, integrate it with a second stimulus and make a predefined correct choice. More specifically, networks first observe stimulus A for 20 time steps, followed by a delay for 10 time steps, followed by stimulus B for 20 steps. Agents must then make one choice. This set-up can be interpreted as a one-step navigation task, where agents are presented with the goal location (stimulus A) followed by possible choice directions (stimulus B). The choice to be made is the one moving closer to the goal. Extended

Data Table 1 outlines all possible trials and defines whether the given trial is included in the regular version of the task used in the main text.

All stimuli are one-hot encoded with a vector of eight binary digits. The first four define the goal locations and only one of the four digits would be set to one during the goal presentation. The second four binary digits each stand in for one allowed choice direction and two choice directions would be set to one during the choice options presentation. Gaussian noise with a standard deviation of 0.05 is added to all inputs.

This task design is a simplified version of a multi-step maze navigation task we have recorded in macaques. A harder version of the task with an extended set of trials is equivalent to the first choice monkeys face in their version of the task. We use the full set of trials for a control calculation in Supplementary Fig. 8. After this first choice, the monkeys then continue the task with a further step to reach the goal and collect the reward. As the goal of this study was to establish the emerging features of seRNNs, here we focus just on the first choice and leave questions relating to the multi-step task to future investigations.

RNN modelling

All recurrent neural networks in this project have 100 units in the hidden layer and are defined by the same basic set of equations:

$$h_t = \text{ReLU}\left(W_x x_t + W_h h_{t-1} + b_h\right) \tag{7}$$

$$\hat{y}_t = \sigma\left(W_y h_t + b_y\right) \tag{8}$$

Here x_t is the input vector at time t (1×8), W_x is the input layer weight matrix (8×100) (Xavier initialization), h_{t-1} is the activation of hidden layer at time $t-1$ (1×100) (zeros initialization), W_h is the hidden layer weight matrix (100×100) (orthogonal initialization), b_h is the bias of hidden layer (1×100) (zeros initialization), h_t is the activation of hidden layer at time t (1×100) (zeros initialization), W_y is the output layer weight matrix (100×8) (Xavier initialization), b_y is the bias of network output (1×8) (zeros initialization), σ is the softmax activation function and $\hat{y}_t$ is the network output/prediction.

Networks differ in terms of which regularization was applied to its hidden layer and with which regularization strength. Networks are optimized to minimize a cross entropy loss on task performance combined with the regularization penalty using the Adam optimizer (hyperparameters: learning rate 0.001, beta_1 0.9, beta_2 0.999, epsilon 1×10^{-7}) for 10 epochs. Note that the network's choice is only read out once, at the very end of the trial. Each epoch consists of 5,120 problems, batched in blocks of 128 problems.

Regularization strength set-up and network selection

The most critical parameter choice in our analyses is the regularization strength. As shown across analyses (for example, Fig. 6), the strength of the regularization has a major influence on all metrics analysed here. While the L1 regularization and the purely Euclidean regularization could be matched by average strength of regularization of the hidden layer, the communicability term of seRNNs makes this challenging due to it being dependent on the current state of the hidden layer and hence changing throughout training. To match the spectrum of regularization strengths in L1 and seRNNs, we used a functional approach. As performance in the task starts to break down as networks become too sparse to effectively remember past stimuli, we matched regularization strength using task performance before looking at any of the other structural or functional metrics. Specifically, we set the regularization spectrum on a linear scale and chose the boundary values so that task performance started to deteriorate half-way through the set of networks (so around the 500th network for the sets of 1,000 networks).

To make both groups comparable, we focus our analyses on networks that achieve >90% task accuracy. For the L1 networks, these were 47.9% of all trained networks and for seRNN networks 39%. Note that this difference in percentages is not meaningful per se and could

be eliminated by matching the regularization spectra of both groups more closely. As we focus our analyses on highly functional networks with high task accuracy, matching the regularization spectra of both groups would have not influenced the results. The code repository has an overview file with regularization strengths chosen for different network types. We hope that future implementations of the seRNNs can provide a method for more precise numerical matching between regularization strengths.

Topological analysis

Graph theory network statistics were calculated using the Brain Connectivity Toolbox[71], and the mathematical formalisms are provided. All network statistics were calculated on the hidden RNN weight matrix and all edges were enforced to be the absolute value of the element. When the measure in question was binary (for example, small-worldness) a proportional threshold was applied, taking the top 10% of these absolute connections.

Modularity. The modularity statistic, Q, quantifies the extent to which the network can be subdivided into clearly delineated groups:

$$Q = \frac{1}{l} \sum_{i,j \in N} \left(a_{i,j} - \frac{k_i k_j}{l}\right) \delta_{m_i m_j}, \tag{9}$$

where l is number of connections, N is the total number of nodes, a_{ij} is the connection status between nodes i and j ($a_{i,j} = 1$ when i and j are connected) and $a_{i,j} = 0$ otherwise, where k_i and k_j are the total number of connections (degrees) of nodes i and j. m_i is the module containing node i, and $\delta_{m_i m_j} = 1$ if $m_i = m_j$, and 0 otherwise. In this work, we tested the modularity using the default resolution parameter of 1.

Small-worldness. Small-worldness refers to a graph property where most nodes are not neighbours of one another, but the neighbours of nodes are likely to be neighbours of each other. This means that most nodes can be reached from every other node in a small number of steps. It is given by:

$$\sigma = \frac{c/c_{\text{rand}}}{l/l_{\text{rand}}}, \tag{10}$$

where c and c_{rand} are the clustering coefficients, and l and l_{rand} are the characteristic path lengths of the respective tested network and a random network with the same size and density of the empirical network. Networks are generally considered as small-world networks at $\sigma > 1$. In our work, we computed the random network as the mean statistic across a distribution of $n = 1,000$ random networks. The characteristic path length is given by:

$$L_i = \frac{1}{n} \sum_{i \in N} \frac{\sum_{j \in N, j \neq i} d_{i,j}}{n - 1} \tag{11}$$

Generative network modelling

We use a technique called generative network modelling to investigate whether the connectome of networks can be recreated by unsupervised wiring rules. The idea is to start from an empty network and probabilistically add connections-based simple wiring equations. The wiring equations are based on the topological structure of the existing network. We follow the approach outlined in refs. 15,35. We provide an overview of this approach in 'Generative network modelling of RNNs' in Supplementary Information.

Decoding

To analyse the internal function of our trained recurrent neural networks, we record the hidden state activity of every unit while the

network solves a set of 640 trials. Each trial is constituted of 50 steps (as shown in Fig. 1e). For decoding, the activity is averaged in step windows of 5, so that there is a total of 10 time windows. In animal electrophysiology, researchers often look at the explained variance per task variable per unit. To allow for comparison of our networks with findings in the literature, we wanted to extract the same metric. Given the nature of our task, the variables used to predict unit activity (goal, choice options, correct choice) are highly correlated, so that the standard decoding with analysis of variance (ANOVA) would give biased results. Instead, we used a decoding algorithm based on L1 regression, as follows.

(1) Apply cross-validated L1 regression with *k*-fold cross validation (5 folds) to set alpha term with best cross-validation performance.
(2) Split the dataset via repeated *k*-fold (3 folds, 2 repeats).
(3) On each (train, test) dataset:

(a) Train L1 regression with the pre-set alpha term.
(b) Calculate explained variance in test dataset including all predictor variables.
(c) Iteratively set all values of a given set of predictors (for example, all goal predictors) to 0 and recalculate the explained variance and calculate the drop of explained variance per predictor group.
(d) Take mean of drop of explained variance for each group across splits of dataset.

This algorithm results in every unit in every network being assigned an explained variance number for every task variable. Note that the decoding cannot reliably work in networks that make too many errors, so that we functionally analyse only networks with a task performance of 90% or above.

Spatial permutation test
To examine the spatial clustering of decoded task information of neuronal ensembles within the RNNs, we constructed a simple spatial permutation test as follows.

(1) Considering a single RNN hidden layer at a particular task time window (note, explained variances change over the course of the task), for each unit, compute the relative preference for goal versus choice explained variance for each unit. This is calculated as the goal explained variance minus the choice explained variance.
(2) Between all *n* 'goal' units (that is, positive difference from step 1), compute the Euclidean distance weighted by the decoding for goal information. This, therefore, captures the spatial proximity between goal units weighted by the magnitude of their 'goal' information. Average this matrix to compute a summary statistic. This is the observed statistic.
(3) Then repeat this procedure for 1,000 times, but for a random set of *n* units taken from the 3D grid space. These 1,000 summary statistics constitutes the null distribution.
(4) Compute a permuted *P* value (P_{perm}), which is simply the location in which the observed statistic (step 2) sits within the null distribution (step 3) normalized to the range [0 1]. This value subsequently corresponds to how clustered or distributed the observed goal decoding information is clustered in space relative to random chance. A small P_{perm} means that information is clustered more than chance and vice versa.
(5) Do steps 1–4, but between all 'choices' units (that is, negative difference from step 1).
(6) Redo steps 1–5 for all desired time windows that have been decoded. In the current work, we calculated P_{perm} values for time window 3, time window 6 and time window 9 to reflect different aspects of the task over the sequence of the task.

The above steps were done for all functional RNNs (>90% accuracy) for L1 and seRNNs. We presented distributions of these P_{perm} values for goals and choices to highlight how goal and choices information is clustered, distributed or random at key points in the sequence of the task. To ensure that we did not bias our findings, we further computed a slight variation of the above statistical test, which allows us to assess the clustering of coding information independently (that is, without computing relative goal versus choice coding, as in step 1 above). As cluster size was now not determined by the direction of coding (as it was previously), we instead used the 50 units with the highest variance-explained values for a given variable. This was selected because this approximately mirrors the cluster sizes achieved in the primary functional clustering analysis. Mirroring the permutation testing approach, we calculated P_{perm} by ranking the mean Euclidean distance between these units (top 50% coding neurons) in a null distribution of Euclidean distance between 1,000 permuted samples of 50 units. This was done for goal and choice options (to assess replication). This test is advantageous in that it allows for testing variables independently, but disadvantageous in that it does not directly incorporate the coding magnitude into the test statistics. These findings are given in Supplementary Fig. 12.

Reporting summary
Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability
No unique data were used in the production of this paper. Reference values were extracted from the respective cited reference. All data shown in the figures are based on simulations, which are described in the 'Code availability' section. The data files generated with simulations that underlie the figures are available in the CodeOcean capsule belonging to this paper (https://doi.org/10.24433/CO.3539394.v2)[72].

Code availability
We provide detailed walkthroughs for the training of our recurrent neural networks alongside all the code used to create the plots in this paper on CodeOcean (https://doi.org/10.24433/CO.3539394.v2)[72]. We provide additional example implementations of seRNNs on GitHub. As new implementations of seRNNs become available, we will add them to this paper's GitHub repository alongside the implementation used for this project. The GitHub repository is https://github.com/8erberg/spatially-embedded-RNN.

References
1. Fair, D. A. et al. Functional brain networks develop from a 'local to distributed' organization. *PLoS Comput. Biol.* **5**, e1000381 (2009).
2. Kaiser, M. Mechanisms of connectome development. *Trends Cogn. Sci.* **21**, 703–717 (2017).
3. Bosman, C. & Aboitiz, F. Functional constraints in the evolution of brain circuits. *Front. Neurosci.* **9**, 303 (2015).
4. Heuvel van den, M. P., Bullmore, E. T. & Sporns, O. Comparative connectomics. *Trends Cogn. Sci.* **20**, 345–361 (2016).
5. Hiratani, N. & Latham, P. E. Developmental and evolutionary constraints on olfactory circuit selection. *Proc. Natl Acad. Sci. USA* **119**, e2100600119 (2022).
6. Mišić, B. et al. Network-level structure–function relationships in human neocortex. *Cereb. Cortex* **26**, 3285–3296 (2016).
7. Smith, V. et al. Fluid intelligence and naturalistic task impairments after focal brain lesions. *Cortex* **146**, 106–115 (2022).
8. Doerig, A. et al. The neuroconnectionist research programme. *Nat. Rev. Neurosci.* https://doi.org/10.1038/s41583-023-00705-w (2023).
9. Kaiser, M. & Hilgetag, C. C. Modelling the development of cortical systems networks. *Neurocomputing* **58–60**, 297–302 (2004).

10. Mante, V., Sussillo, D., Shenoy, K. V. & Newsome, W. T. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature* **503**, 78–84 (2013).

11. Ali, A., Ahmad, N., De Groot, E., Johannes Van Gerven, M. A. & Kietzmann, T. C. Predictive coding is a consequence of energy efficiency in recurrent neural networks. *Patterns* **3**, 100639 (2022).

12. Barthélemy, M. Spatial networks. *Phys. Rep.* **499**, 1–101 (2011).

13. Bassett, D. S. & Stiso, J. Spatial brain networks. *C. R. Phys.* **19**, 253–264 (2018).

14. Bullmore, E. & Sporns, O. The economy of brain network organization. *Nat. Rev. Neurosci.* **13**, 336–349 (2012).

15. Akarca, D. et al. Homophilic wiring principles underpin neuronal network topology in vitro. Preprint at *bioRxiv* https://doi.org/10.1101/2022.03.09.483605 (2022).

16. Song, H. F., Kennedy, H. & Wang, X.-J. Spatial embedding of structural similarity in the cerebral cortex. *Proc. Natl Acad. Sci. USA* **111**, 16580–16585 (2014).

17. Avena-Koenigsberger, A., Misic, B. & Sporns, O. Communication dynamics in complex brain networks. *Nat. Rev. Neurosci.* **19**, 17–33 (2018).

18. Laughlin, S. B. & Sejnowski, T. J. Communication in neuronal networks. *Science* **301**, 1870–1874 (2003).

19. Crofts, J. J. & Higham, D. J. A weighted communicability measure applied to complex brain networks. *J. R. Soc. Interface* **6**, 411–414 (2009).

20. Griffa, A. et al. The evolution of information transmission in mammalian brain networks. Preprint at *bioRxiv* https://doi.org/10.1101/2022.05.09.491115 (2022).

21. Seguin, C., Mansour L, S., Sporns, O., Zalesky, A. & Calamante, F. Network communication models narrow the gap between the modular organization of structural and functional brain networks. *NeuroImage* **257**, 119323 (2022).

22. Estrada, E. & Hatano, N. Communicability in complex networks. *Phys. Rev. E* **77**, 036111 (2008).

23. Betzel, R. F., Medaglia, J. D. & Bassett, D. S. Diversity of meso-scale architecture in human and non-human connectomes. *Nat. Commun.* **9**, 346 (2018).

24. Bertolero, M. A., Yeo, B. T. T. & D'Esposito, M. The modular and integrative functional architecture of the human brain. *Proc. Natl Acad. Sci. USA* **112**, E6798–E6807 (2015).

25. Park, H.-J. & Friston, K. Structural and functional brain networks: from connections to cognition. *Science* **342**, 1238411 (2013).

26. Sporns, O. & Betzel, R. F. Modular brain networks. *Annu. Rev. Psychol.* **67**, 613–640 (2016).

27. Bassett, D. S. & Bullmore, E. T. Small-world brain networks revisited. *Neuroscientist* **23**, 499–516 (2017).

28. Sporns, O. & Zwi, J. D. The small world of the cerebral cortex. *Neuroinformatics* **2**, 145–162 (2004).

29. Johnston, W. J., Palmer, S. E. & Freedman, D. J. Nonlinear mixed selectivity supports reliable neural computation. *PLoS Comput. Biol.* **16**, e1007544 (2020).

30. Rigotti, M. et al. The importance of mixed selectivity in complex cognitive tasks. *Nature* **497**, 585–590 (2013).

31. Hilger, K., Ekman, M., Fiebach, C. J. & Basten, U. Intelligence is associated with the modular structure of intrinsic brain networks. *Sci. Rep.* **7**, 16088 (2017).

32. Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* **393**, 440–442 (1998).

33. Bassett, D. S. & Sporns, O. Network neuroscience. *Nat. Neurosci.* **20**, 353–364 (2017).

34. Betzel, R. F. et al. Generative models of the human connectome. *NeuroImage* **124**, 1054–1064 (2016).

35. Akarca, D., Vértes, P. E., Bullmore, E. T. & Astle, D. E. A generative network model of neurodevelopmental diversity in structural brain organization. *Nat. Commun.* **12**, 4216 (2021).

36. Vértes, P. E. et al. Simple models of human brain functional networks. *Proc. Natl Acad. Sci. USA* **109**, 5868–5873 (2012).

37. Kanwisher, N. Functional specificity in the human brain: a window into the functional architecture of the mind. *Proc. Natl Acad. Sci. USA* **107**, 11163–11170 (2010).

38. Thompson, W. H. & Fransson, P. Spatial confluence of psychological and anatomical network constructs in the human brain revealed by a mass meta-analysis of fMRI activation. *Sci. Rep.* **7**, 44259 (2017).

39. Waskom, M. L. & Wagner, A. D. Distributed representation of context by intrinsic subnetworks in prefrontal cortex. *Proc. Natl Acad. Sci. USA* **114**, 2030–2035 (2017).

40. Ji, J. L. et al. Mapping the human brain's cortical-subcortical functional network organization. *NeuroImage* **185**, 35–57 (2019).

41. Steinmetz, N. A., Zatka-Haas, P., Carandini, M. & Harris, K. D. Distributed coding of choice, action and engagement across the mouse brain. *Nature* **576**, 266–273 (2019).

42. Wallach, A., Melanson, A., Longtin, A. & Maler, L. Mixed selectivity coding of sensory and motor social signals in the thalamus of a weakly electric fish. *Curr. Biol.* **32**, 51–63.e3 (2022).

43. Hirokawa, J., Vaughan, A., Masset, P., Ott, T. & Kepecs, A. Frontal cortex neuron types categorically encode single decision variables. *Nature* **576**, 446–451 (2019).

44. Whittington, J. C. R., Dorrell, W., Ganguli, S. & Behrens, T. E. J. Disentanglement with biological constraints: a theory of functional cell types. In *The Eleventh International Conference on Learning Representations* (ICLR, 2023).

45. Fusi, S., Miller, E. K. & Rigotti, M. Why neurons mix: high dimensionality for higher cognition. *Curr. Opin. Neurobiol.* **37**, 66–74 (2016).

46. Bernardi, S. et al. The geometry of abstraction in the hippocampus and prefrontal cortex. *Cell* **183**, 954–967.e21 (2020).

47. Erez, Y. et al. Integrated neural dynamics for behavioural decisions and attentional competition in the prefrontal cortex. *Eur. J. Neurosci.* **56**, 4393–4410 (2022).

48. Attwell, D. & Laughlin, S. B. An energy budget for signaling in the grey matter of the brain. *J. Cereb. Blood Flow Metab.* **21**, 1133–1145 (2001).

49. Beggs, J. M. The criticality hypothesis: how local cortical networks might optimize information processing. *Phil. Trans. R. Soc. Math. Phys. Eng. Sci.* **366**, 329–343 (2008).

50. Cohen, J. R. & D'Esposito, M. The segregation and integration of distinct brain networks and their relationship to cognition. *J. Neurosci.* **36**, 12083–12094 (2016).

51. O'Byrne, J. & Jerbi, K. How critical is brain criticality? *Trends Neurosci.* **45**, 820–837 (2022).

52. Yan, C. & He, Y. Driving and driven architectures of directed small-world human brain functional networks. *PLoS ONE* **6**, e23460 (2011).

53. Suárez, L. E., Richards, B. A., Lajoie, G. & Misic, B. Learning function from structure in neuromorphic networks. *Nat. Mach. Intell.* **3**, 771–786 (2021).

54. Lindsay, G. W., Rigotti, M., Warden, M. R., Miller, E. K. & Fusi, S. Hebbian learning in a random network captures selectivity properties of the prefrontal cortex. *J. Neurosci.* **37**, 11021–11036 (2017).

55. Finzi, D., Margalit, E., Kay, K., Yamins, D. L. & Grill-Spector, K. Topographic DCNNs trained on a single self-supervised task capture the functional organization of cortex into visual processing streams. In *NeurIPS 2022 Workshop SVRHM* (2022); https://openreview.net/forum?id=E1iY-d13smd

56. Damicelli, F., Hilgetag, C. C. & Goulas, A. Brain connectivity meets reservoir computing. *PLOS Comput. Biol.* **18**, e1010639 (2022).

57. Goulas, A., Damicelli, F. & Hilgetag, C. C. Bio-instantiated recurrent neural networks: integrating neurobiology-based network topology in artificial networks. *Neural Netw.* **142**, 608–618 (2021).

58. Zador, A. et al. Catalyzing next-generation artificial intelligence through NeuroAI. *Nat. Commun.* **14**, 1597 (2023).

59. Moons, L. & De Groef, L. Molecular mechanisms of neural circuit development and regeneration. *Int. J. Mol. Sci.* **22**, 4593 (2021).

60. Perez-Nieves, N., Leung, V. C. H., Dragotti, P. L. & Goodman, D. F. M. Neural heterogeneity promotes robust learning. *Nat. Commun.* **12**, 5791 (2021).

61. Baxter, R. A. & Levy, W. B. Constructing multilayered neural networks with sparse, data-driven connectivity using biologically-inspired, complementary, homeostatic mechanisms. *Neural Netw.* **122**, 68–93 (2020).

62. Chechik, G., Meilijson, I. & Ruppin, E. Synaptic pruning in development: a computational account. *Neural Comput.* **10**, 1759–1777 (1998).

63. Johnson, M. H. Interactive specialization: a domain-general framework for human functional brain development? *Dev. Cogn. Neurosci.* **1**, 7–21 (2011).

64. Siugzdaite, R., Bathelt, J., Holmes, J. & Astle, D. E. Transdiagnostic brain mapping in developmental disorders. *Curr. Biol.* **30**, 1245–1257.e4 (2020).

65. Bassett, D. S. et al. Efficient physical embedding of topologically complex information processing networks in brains and computer circuits. *PLoS Comput. Biol.* **6**, e1000748 (2010).

66. Sperry, M. M., Telesford, Q. K., Klimm, F. & Bassett, D. S. Rentian scaling for the measurement of optimal embedding of complex networks into physical space. *J. Complex Netw.* **5**, 199–218 (2017).

67. Hardt, M. & Recht, B. *Patterns, Predictions, and Actions* (Princeton Univ. Press, 2022).

68. Kietzmann, T. C. et al. Recurrence is required to capture the representational dynamics of the human visual system. *Proc. Natl Acad. Sci. USA* **116**, 21854–21863 (2019).

69. Yang, G. R., Joglekar, M. R., Song, H. F., Newsome, W. T. & Wang, X.-J. Task representations in neural networks trained to perform many cognitive tasks. *Nat. Neurosci.* **22**, 297–306 (2019).

70. Tibshirani, R. Regression shrinkage and selection via the LASSO. *J. R. Stat. Soc. Ser. B* **58**, 267–288 (1996).

71. Rubinov, M. & Sporns, O. Complex network measures of brain connectivity: uses and interpretations. *NeuroImage* **52**, 1059–1069 (2010).

72. Achterberg, J., Akarca, D., Strouse, D. J., Duncan, J. & Astle, D. E. Capsule for Achterberg & Akarca, et al: Spatially-embedded recurrent neural networks reveal widespread links between structural and functional neuroscience findings. *CodeOcean* https://doi.org/10.24433/CO.3539394.v2 (2023).

Acknowledgements

Author contributions

J.A., D.A., D.J.S., J.D. and D.E.A. were involved in the conceptualization and methodological developments of the study. J.A. and D.A. conducted the analyses. J.A., D.A., D.J.S., J.D. and D.E.A interpreted the results. J.A. and D.A. wrote the paper in consultation with J.D. and D.E.A.

Competing interests

The authors declare no competing interests.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-023-00748-9.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-023-00748-9.

Correspondence and requests for materials should be addressed to Jascha Achterberg or Danyal Akarca.

Peer review information *Nature Machine Intelligence* thanks Bratislav Misic, and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Extended Data Table 1 | List of all problems that make up the task that networks are trained on

Problem	Stimulus A	Stimulus B	Correct Choice	Regular Task
1	(Right, up)	Left, right	Right	No
2	(Right, up)	Right, down	Right	Yes
3	(Right, up)	Up, down	Up	No
4	(Right, up)	Up, left	Up	Yes
5	(Right, down)	Left, right	Right	Yes
6	(Right, down)	Up, right	Right	No
7	(Right, down)	Up, down	Down	No
8	(Right, down)	Left, down	Down	Yes
9	(Left, up)	Left, right	Left	Yes
10	(Left, up)	Left, down	Left	No
11	(Left, up)	Up, down	Up	No
12	(Left, up)	Up, right	Up	Yes
13	(Left, down)	Left, right	Left	No
14	(Left, down)	Up, left	Left	Yes
15	(Left, down)	Up, down	Down	No
16	(Left, down)	Right, down	Down	Yes

nature portfolio

Corresponding author(s): Jascha Achterberg, Danyal Akarca

Last updated by author(s): Sep 14, 2023

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☐	☒	A description of all covariates tested
☐	☒	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒	For null hypothesis testing, the test statistic (e.g. F, t, r) with confidence intervals, effect sizes, degrees of freedom and P value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☐	☒	Estimates of effect sizes (e.g. Cohen's d, Pearson's r), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	The data used in our manuscript was generated using Python (3.7.9) and Tensorflow (2.3.0) as outlined in the manuscripts methods. The GitHub repository and Code Ocean Capsule linked in the manuscript provides a working version of the new Recurrent Neural Networks developed in this manuscript.
Data analysis	Our data analysis was based on Python (3.9.1), Numpy (1.19.2), statsmodels (0.12.1) scikit-learn (0.23.2), scipy (1.5.2), pandas (1.2.1), Matlab 2020b; Brain Connectivity Toolbox 2019 brain-connectivity-toolbox.net

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:
- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

We provide detailed walkthroughs for the training of our recurrent neural networks alongside all the code used to create the plots in this manuscript on CodeOcean

Human research participants

Policy information about studies involving human research participants and Sex and Gender in Research.

Reporting on sex and gender	NA
Population characteristics	NA
Recruitment	NA
Ethics oversight	NA

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences   ☐ Behavioural & social sciences   ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	We sample 1000 networks per group (seRNN vs L1 networks), so 2000 networks in total. We sample an additional set of 1000 networks for supplementary control analyses (seRNN [standard], seRNN [hard task], seRNN [random inputs], seRNN [Distance only], seRNN [Communicability only], seRNN [Short delay period], seRNN [Long delay period], seRNN [Low noise], seRNN [High noise], L1; 100 networks per group).
Data exclusions	In this manuscript we only analyse networks which reach > 90% task accuracy. Note that we do not apply an accuracy criteria for the seRNN [random inputs] control group used in supplementary control analyses.
Replication	As we based our analysis on large groups of independently trained networks we already show that our results replicate across networks. Our results are limited to a subset of the parameter space as discussed in the manuscript. The exact training parameters are available in the Code Ocean Capsule.
Randomization	NA
Blinding	NA

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging