A unified neurocomputational model for spatial and linguistic representations

Tatsuya Haga[1,2,*], Yohei Oseki[3], and Tomoki Fukai[1]

[1]Neural Computation and Brain Coding Unit, Okinawa Institute of Science and Technology, Okinawa, Japan
[2]Center for Information and Neural Networks (CiNet), National Institute of Information and Communications Technology, Osaka, Japan
[3]Graduate School of Arts and Sciences, University of Tokyo, Tokyo, Japan
[*]Corresponding author E-mail: tatsuyahaga@nict.go.jp

Abstract

Hippocampus and entorhinal cortex encode spaces by spatially local and hexagonal grid activity patterns (place cells and grid cells), respectively. In addition, the same brain regions also implicate neural representations for non-spatial, semantic concepts (concept cells). However, the relationship between those representations remains to be understood. We propose a unified neurocomputational model for spaces and linguistic concepts, called "disentangled successor information (DSI)", based on reinforcement learning and word embedding models in natural language processing (NLP). DSI generates spatial representations in a 2-dimensional space that correspond to place cells and grid cells. Furthermore, the same model creates concept-specific linguistic representations which resemble concept cells found in hippocampus and entorhinal cortex. Notably, with DSI representations, we could perform analogical inference of spatial contexts and words by the same computational framework, which can be biologically interpreted as partial modulation of assemblies of concept cells and non-grid cells in EC. Our model suggests the existence of a shared computational mechanism behind spatial and conceptual representations in hippocampus and entorhinal cortex.

Introduction

In the brain, place cells in the hippocampus (HPC) and grid cells in the entorhinal cortex (EC) represent spaces by spatially local and hexagonal grid activity patterns, respectively[1–4]. Place cells can directly support spatial recognition and memory, whereas grid cells are considered as the basis of robust spatial navigation and path integration. Theoretically, an animal can estimate the direction to a goal when grid representations of a current position and a goal position are given (vector-based spatial navigation) [5]. Furthermore, self-position can be estimated by integrating self-motions when sensory information is not available (path integration)[6,7]. These functions are bases of robust spatial navigation by animals. In addition, recent experiments suggest that grid representations in EC appear not only for physical space but also for 2-dimensional perceptual or conceptual space (e.g. 2-D visual[8], olfactory[9], social[10] spaces, and pseudowords[11,12] and objects[13] associated to 2-D structures), and those representations are the basis of vector-based conceptual inference. HPC also exhibits neural representations for conceptual spaces[14,15].

Because spatial navigation is a special case of decision making, we can hypothesize that such spatial representations in HPC and EC are designed for efficient reinforcement learning. Previous computational studies suggest that successor representation (SR), which was originally proposed for efficient evaluation of value functions[16,17], can explain various experimental observations of place cells and grid cells[18,19]. Further experiments confirmed the existence of SR-like representations in the brain[19–21]. Furthermore, default representations (DR) based on the theory of linear reinforcement learning[22,23] explains that spatial representations in EC are related to flexible behaviors[24]. These theories provide the formal explanation of the design principle of spatial representations like place cells and grid cells.

However, in HPC and EC, there are also neurons representing non-spatial semantic concepts which are called as "concept cells"[25–27]. Concept cells respond to specific concepts, namely, stimuli related to a specific person, a famous place, or a specific category like "foods" and "clothes", irrespectively of sensory modality (images, written or spoken words). Those cells are activated not only by external stimuli but also by imagery[28] and memory recall[29], suggesting that their activities are not merely sensory responses but conceptual representations of the high-level cognitive process. Because these cells have been found in HPC and EC, it is possible that concept cells can also be understood from the theory of spatial representations. However, how the formal spatial representation model like SR can be extended to such conceptual representations has not been fully understood.

In this paper, we propose a unified neurocomputational model called "disentangled successor information (DSI)" and demonstrate that the spatial representation model based on reinforcement learning can be extended to conceptual representations. Like SR, DSI forms place and grid representations corresponding to place cells and grid cells, supporting near-optimal decision making for spatial navigation. On the other hand, when we apply DSI to languages, DSI forms word representations in which each unit is activated by a specific concept like concept cells. This suggests that neural representations corresponding to place cells, grid cells and concept cells can be generated through the same computational framework.

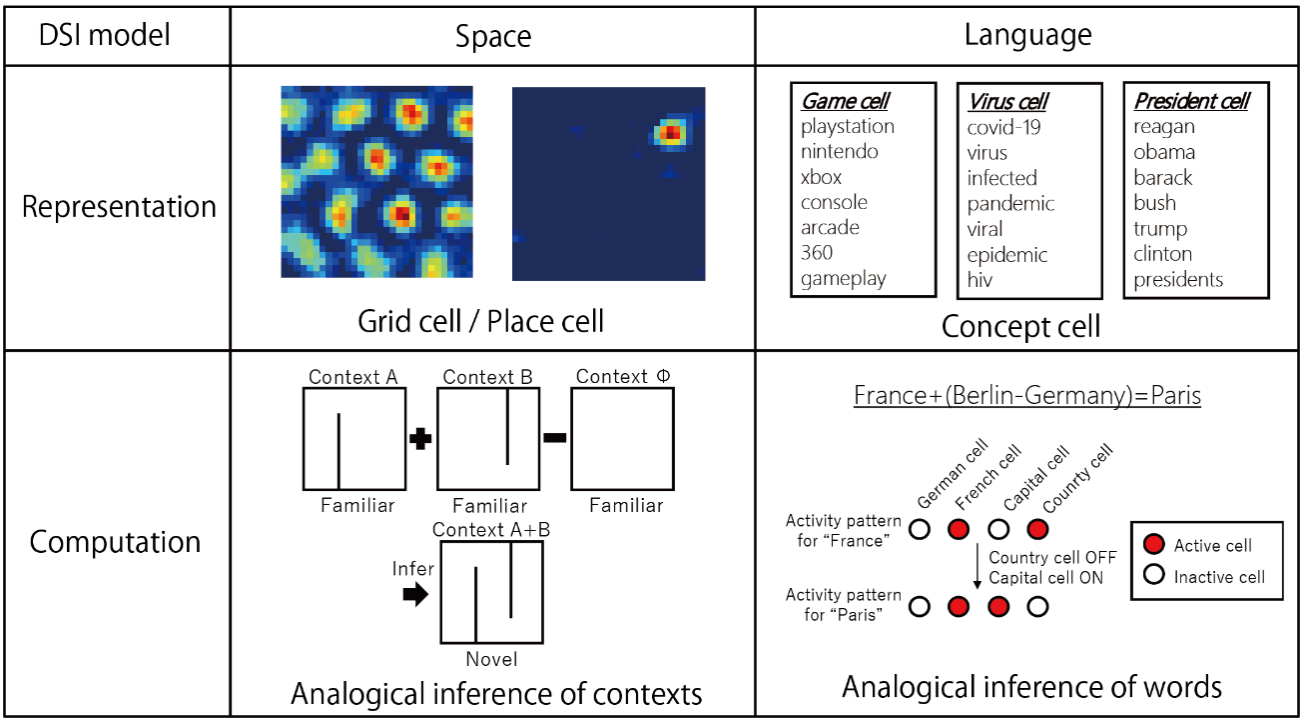

Figure 1: Summary of DSI model.

Furthermore, DSI model offers a shared neural mechanism across spatial and linguistic inference. We show that DSI is mathematically related to word embedding models in natural language processing (NLP)[30–33] and, therefore, DSI can perform analogical inference of words just like those models. Nevertheless, unlike conventional word embedding models, DSI enables inference by switching only a few conceptual units, which can be biologically interpreted as a partial recombination of concept-cell assemblies. Remarkably, we found that the same computational framework enables analogical inference of spatial contexts by combining previously learned spatial representations. As in experiments[34], non-grid spatial representations rather than grid representations were crucial for this computation of spatial contexts. Thus, our model suggests a shared computational principle behind spatial and conceptual inferences as well as similarities between non-grid cells and concept cells in biological computation.

Previous computational studies have revealed how spatial[18,20,24,35–40] and conceptual[41–45] representations emerge and function in the brain. However, computational understanding of the shared mechanism between spatial and conceptual representations has been scarce, especially for highly complex conceptual structures like languages. Although several models have recently worked on this problem[46–49], it is often difficult to discuss computational properties of neural representations other than place cells and grid cells, such as concept cells[25–27] and non-grid spatial representations that represent contextual information in EC[34,50]. Our model explains representations and computations for spaces and linguistic concepts (Figure 1), which are biologically interpretable at unit-level and at population-level. From another perspective, our model suggests that word embedding models in NLP can be grounded on processing in the human brain. Our model reveals a new theoretical relationship between spatial and linguistic representation learning, and supports a hypothesis that there is a shared computational principle behind spatial and conceptual representations in HPC and EC.

Disentangled successor information model

We define our DSI model based on SR model of HPC and EC[16,18,19]. Let us assume N_s discrete states exist in the environment. SR between two states s and s' is defined as

$$SR(s,s') = E\left[\sum_{t=0}^{\infty} \gamma^t \, \delta(s_t, s') | s_0 = s\right] = \sum_{t=0}^{\infty} \gamma^t \, P(s_t = s' | s_0 = s) \tag{1}$$

where $\delta(i,j)$ is Kronecker's delta and γ is a discount factor. Based on SR, we define successor information (SI) and positive successor information (PSI) as

$$SI(s,s') = \log\big(SR(s,s')\big) - \log\big(P(s')\big) \tag{2}$$

$$PSI(s,s') = \max\{SI(s,s'), 0\} \tag{3}$$

We performed dimension reduction of SI matrix using an extension of nonnegative matrix factorization (NMF) [51] with additional constraints (See Methods section for details), then we finally obtain D-dimensional non-negative representation vectors $\vec{x}(s)$ and $\vec{w}(s)$ ($D < N_s$) which satisfies $\vec{x}(s) \cdot \vec{w}(s') \approx PSI(s,s')$. Those vectors are called as disentangled successor information (DSI) but we specifically focus on the vector $\vec{x}(s)$ in this paper (we explain the reason in the following section).

We tested two types of constraints, and we call vectors as DSI-decorr or DSI-sparse depending on settings. The first one, DSI-decorr was constrained by decorrelation and L-2 regularization. The other one, DSI-sparse was obtained under L-1 sparse constraint. Previous theoretical studies have shown that non-negativity, decorrelation (or orthogonality), and regularization are important for generation of grid representations[35–40,52,53]. A previous study also suggests that the combination of non-negativity and sparsity enable generation of word representation vectors with conceptual specificity[54]. Therefore, we can hypothesize that either or both of those constraints achieves generation of biologically plausible spatial representations and disentangled (concept-specific) linguistic representations, which correspond to grid cells and concept cells, respectively. These constraints are biologically plausible because neural activities are basically non-negative and decorrelation and sparsification is possible through lateral inhibition.

DSI relates to spatial navigation and word embedding

Before we empirically show properties of DSI representations, we discuss dual mathematical interpretation of DSI model related to spatial and language processing. On the one hand, DSI approximates value estimation of linear reinforcement learning, supporting goal-directed spatial navigation. On the other hand, the same representation approximates word embedding, supporting language processing.

First, our model approximates value functions of linear reinforcement learning[22–24] in the setting of spatial navigation. Linear reinforcement learning assumes default policy and imposes additional penalty on deviation from default policy, then we can obtain value functions explicitly by solving linear equations. Let us consider a specific condition in which the environment consists of non-terminal states, and a virtual terminal state is

attached to a goal state s_G arbitrarily chosen from non-terminal states. When the agent gets to the goal, it transits to the terminal state and obtain positive reward. Furthermore, we assume that rewards at non-terminal states are uniformly negative so that the agent has to take a short path to goal to maximize reward. In this setting, we can obtain value functions $v^*(s)$ in linear reinforcement learning as

$$\lambda^{-1}v^*(s) = \log SR^d(s, s_G) - \log P(s_G) = SI^d(s, s_G) \approx \vec{x}(s) \cdot \vec{w}(s_G) \tag{4}$$

where $SR^d(s, s_G)$ and $SI^d(s, s_G)$ are SR and SI under the default policy, respectively, and λ is a parameter representing the relative weight of penalty on deviation from default policy (see Methods section for details of derivation). Therefore, SI is proportional to value functions for spatial navigation and inner products of DSI vectors approximates value functions. Based on this interpretation, we basically regard $\vec{x}(s)$ as a representation of each state, and $\vec{w}(s)$ represents a temporary goal. Below, we call $\vec{x}(s)$ as a DSI representation vector unless otherwise specified.

Second, DSI is related to word embedding models in NLP[30–33,54]. In linguistics, pointwise mutual information (PMI) and positive pointwise mutual information (PPMI) are used to measure the degree of coincidence between two words[33]. They are defined as

$$PMI = \log\left(\frac{P(word_i, word_j)}{P(word_i)P(word_j)}\right) \tag{5}$$

$$PPMI = \max\{PMI, 0\} \tag{6}$$

where $P(word_i, word_j)$ is a coincidence probability of two words (in a certain temporal window). It has been proven that dimension reduction of PMI approximates a conventional word embedding model skip-gram[30,31], and similar performance is obtained using PPMI[33]. SI can be written as

$$SI(s, s') = \log SR(s, s') - \log P(s') = \log\left(\frac{\sum_{t=1}^{\infty} P(s_t = s', s_0 = s)}{P(s)P(s')}\right) \tag{7}$$

In this formulation, we can see mathematical similarity between PMI and SI by regarding words as states ($s = word_i, s' = word_j$), thus the correspondence between PPMI and PSI. Because of this relationship, we can expect that DSI, which is obtained through dimension reduction of PSI, has similar properties to word embedding models. The difference is how to count coincidence: the coincidence in SI is evaluated with an asymmetric exponential kernel as in SR, in contrast that a symmetric rectangular temporal window is often used in typical word embedding.

Emergence of spatial representations like grid cells and place cells

Below, we empirically show properties of DSI representations. First, we checked spatial representations that the DSI model forms in a 2-D space. As an environment, we assumed a square room tiled with 30 × 30 discrete states (Figure 2A). We generated sequences of states by random walks in the room, from which we calculated 100-dimensional DSI representation vectors for states (places) (see Methods for the detail of the simulation).

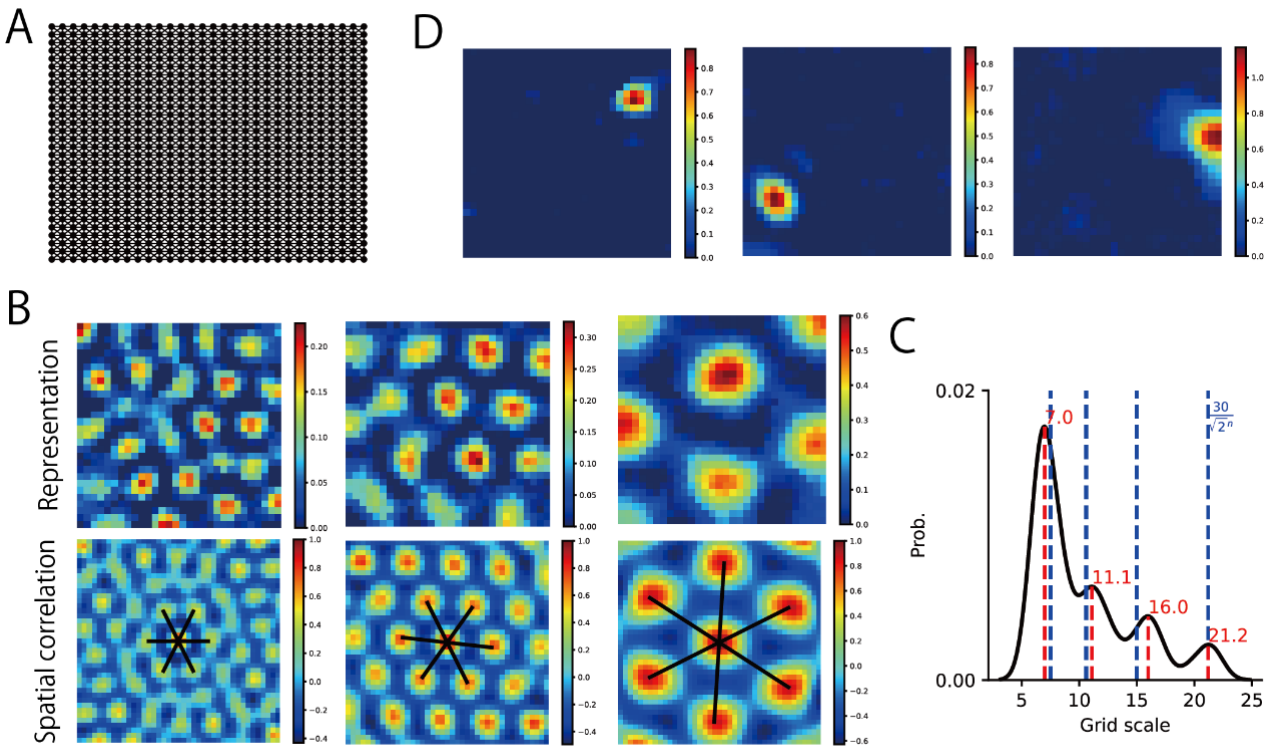

Figure 2: DSI representations for the space. (A) The square room tiled with 30x30 discrete states. (B) Grid-like spatial representations generated by DSI-decorr (upper) and their spatial autocorrelation (lower). (C) The distribution of grid scales of DSI representations. Red and blue broken lines indicate positions of peaks and $\frac{30}{(\sqrt{2})^n}$, respectively. (D) Spatial representations like place cells generated by DSI-sparse.

Here we call each dimension of DSI representation vectors $\vec{x}(s)$ as a neural "unit", and we regard a value in each dimension at each state as a neural activity (or a neural representation). Then, in DSI-decorr vectors, many units exhibited grid activity patterns in the space (Figure 2B). We performed a gridness analysis that has been used in animal experiments[55] and found that 52.6±6.3 % of units (average±std of 5 simulations with different random seeds) were classified as grid cells. Similarly, 52.6±7.7 % of units in $w(s)$ were classified as grid cells. On the other hand, DSI-sparse generated spatially local representations which resemble hippocampal place cells (Figure 2D).

Furthermore, we checked whether spatial representations generated by DSI-decorr reproduce a property of biological grid cells. Actual grid cells in the rat brain exhibit multiple discrete spatial scales and the ratio between grid scales of adjacent modules is $\sqrt{2}$[56]. We constructed a distribution of grid scales of DSI units by kernel density estimation, which revealed that multiple peaks of grid scales existed and the ratio between grid scales of adjacent peaks was approximately $\sqrt{2}$ (Figure 2C). Similar distribution of scales was preserved across simulations with different random seeds (Supplementary Figure 1). These results show that DSI model constructs biologically plausible spatial representations of HPC and EC, depending on the choice of constraints. Below, we mainly show results of DSI-decorr but DSI-sparse also gave qualitatively similar performances in most cases.

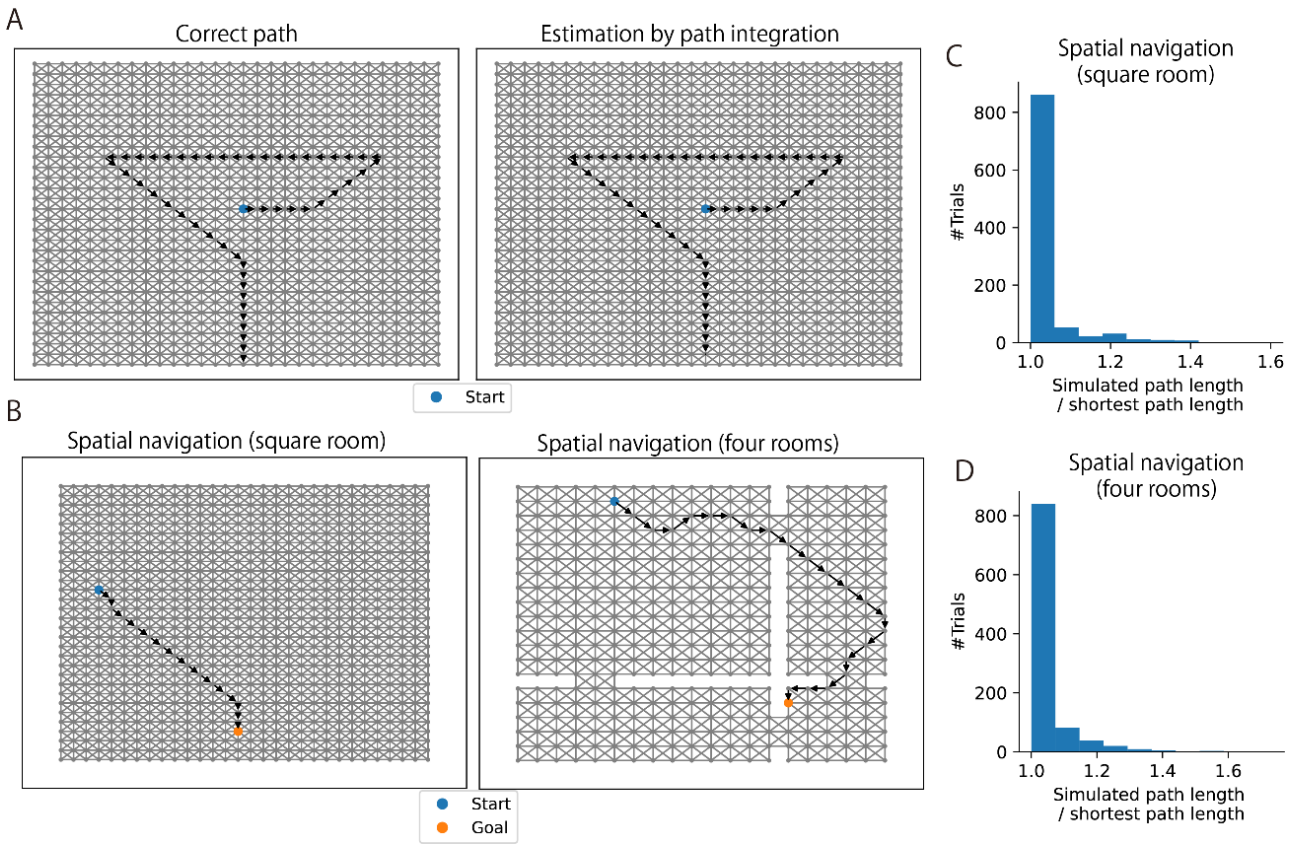

Figure 3: Spatial navigation using DSI representation vectors (decorr-DSI). (A) Path integration using DSI model. (Left) Actual path. (Right) Path estimated from DSI vectors updated by movement information. (B) Example spatial paths obtained by DSI-based navigation. (C) A histogram of path lengths in 1,000 trials of spatial navigation in the square-room environment. Note that a start and a goal were randomly determined in each trial, and we normalized a simulated path length by the shortest path length between the start and the goal. (D) A histogram of path lengths in the four-room environment.

Path integration and spatial navigation by DSI representations

If DSI representations are a plausible model of place cells and grid cells, they should support path integration and spatial navigation, which are important functions of HPC and EC. First, we performed path integration using DSI representations, using movement-conditional recurrent weights to perform path integration (see Methods for details). This strategy has been used in previous studies such as grid cell modeling[39] and action-conditional video prediction[57]. This mechanism is also consistent with a conventional biological model for path integration in which head direction signals activate one of attractor networks specialized for different directional shifts of grid patterns[6,7,36]. As shown in Figure 3A (DSI-decorr) and Supplementary Figure 2 (DSI-sparse), this strategy gave accurate estimation of the spatial path from movement signals. To quantify the accuracy, we evaluated the success rate of estimation of spatial paths generated from random starting points and ten random action sequences. Estimation was correct in 969.4±8.3 trials (DSI-decorr) and 759.2±14.4 trials (DSI-sparse) out of 1,000 trials (average±std of 5 simulations with different random seeds), which suggests that path integration was highly accurate especially with DSI-decorr.

Next, we argue that near-optimal spatial navigation by DSI representations is possible. We assume that a start location and a goal location are randomly given in each trial and an agent has to navigate between them. To solve the task, we defined a vector-based state transition rule that approximates value-based decision making based on the relationship between DSI and value functions (see Methods for details). Because of the approximation error, this rule did not always give optimal navigation (the shortest path from the start to the goal). However, the agent could take relatively short paths (Figure 3B, left and Supplementary Figure 2) and simulated path lengths were close to the shortest path between the start and the goal (Figure 3C and Supplementary Figure 3). Furthermore, the agent could also perform near-optimal spatial navigation in a structure with separated and interconnected rooms (Figure 3B, left 2D, and Supplementary Figure 2). These results suggest that DSI representations can support efficient spatial navigation.

Emergence of concept-specific representations for words

Next, we show that the same DSI model can learn conceptual representations from linguistic inputs. We used text data taken from English Wikipedia, which contains 124M tokens and 9376 words (see Methods section for the detail of preprocessing). To construct DSI representations, we regarded each word as a "state", and considered the text data as a sequence of 9376 states (N_s = 9376). Then, we applied the same learning procedure as in the experiment of 2-D spaces. We obtained 300-dimensional DSI representation vectors for each word.

As in the case of spatial representations, we regard each dimension of representation vectors as a neural unit, and checked how various words activate those units. Specifically, we listed ten words that elicited the highest activities in each unit (TOP-10 words). Then, we found that many units are activated by words related to specific concepts (Figure 4A; other examples in Supplementary Figure 4), which could be named as "game cell" or "president cell", for example. We quantified this conceptual specificity through WordNet-based semantic similarity between words[58]. We compared mean similarity among TOP-10 words and a null distribution of similarity between random word pairs, by which we determined statistically significant concept-specific units and quantified the degree of conceptual specificity of each unit (see Methods section for details). As shown in Figure 4B and 4C, both DSI-decorr and DSI-sparse exhibited the larger number of concept-specific units and higher average conceptual specificity than other well-established word embedding models such as skip-gram and GloVe[30–33]. We also analyzed conceptual specificity of representations in the embedding layer of pretrained BERT model[59,60], which was lower than DSI. This result shows that DSI model forms more concept-specific representations than typical word embedding models in NLP. Remarkably, removal of the non-negativity constraint from DSI-decorr significantly decreased conceptual specificity. This result suggests that non-negativity is an essential constraint for concept-specific word representations as well as hexagonal grid spatial representations[35,36,52], and sparsity is not necessary as proposed in the previous study[54]. It is also notable that continuous bag-of-words (CBOW) model[31] showed almost same conceptual specificity with DSI. However, we see a functional difference between concept-specific representations in DSI and CBOW later.

DSI representation vectors capture the semantic structure of words

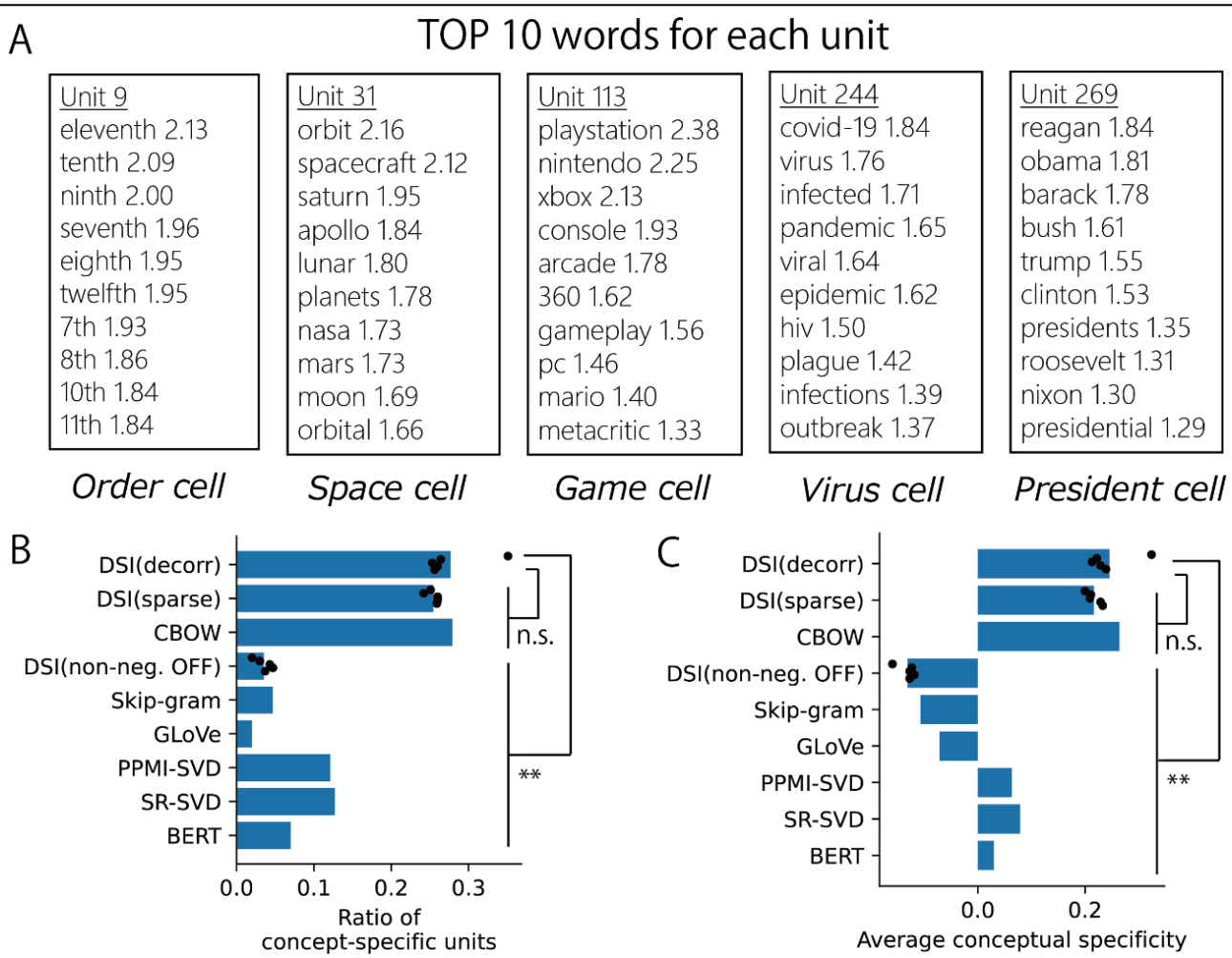

Figure 4: Concept-specific representations formed by decorr-DSI. (A) Ten words that gave the highest activation (TOP-10 words) are shown. We also marked each unit with a descriptive label. (B) Ratio of concept-specific units (dimensions) in word representation vectors obtained by various methods. For DSI, dots indicate 5 trials with different random seeds (different initial values for learning); bars indicate means of those 5 simulations. We compared DSI (decorr) and other DSI models by 2-sample t-tests, and DSI (decorr) and other word embedding methods by 1-sample t-tests. (C) Average conceptual specificity of all units (dimensions) in word representation vectors obtained by various methods. We performed same statistical analyses with (B). **P<0.01; n.s., not significant. All statistical tests were two-sided t-tests and significance thresholds were modified by Bonferroni correction. Details of statistical analyses are shown in Supplementary Table 1.

Concept cells are considered to represent semantic relationships between concepts at the level of population activity patterns[25]. Correspondingly, DSI vectors also represent the semantic structure of words because representation vectors of word embedding models have such property[30–33]. We evaluated how word similarity is captured by cosine similarity of DSI vectors using the evaluation procedure in NLP[30–33]. We calculated cosine similarity between representation vectors of word pairs and evaluated the rank correlation between those cosine similarities and human word similarities (WS353 dataset[61]; 248/345 word pairs were used). As a result,

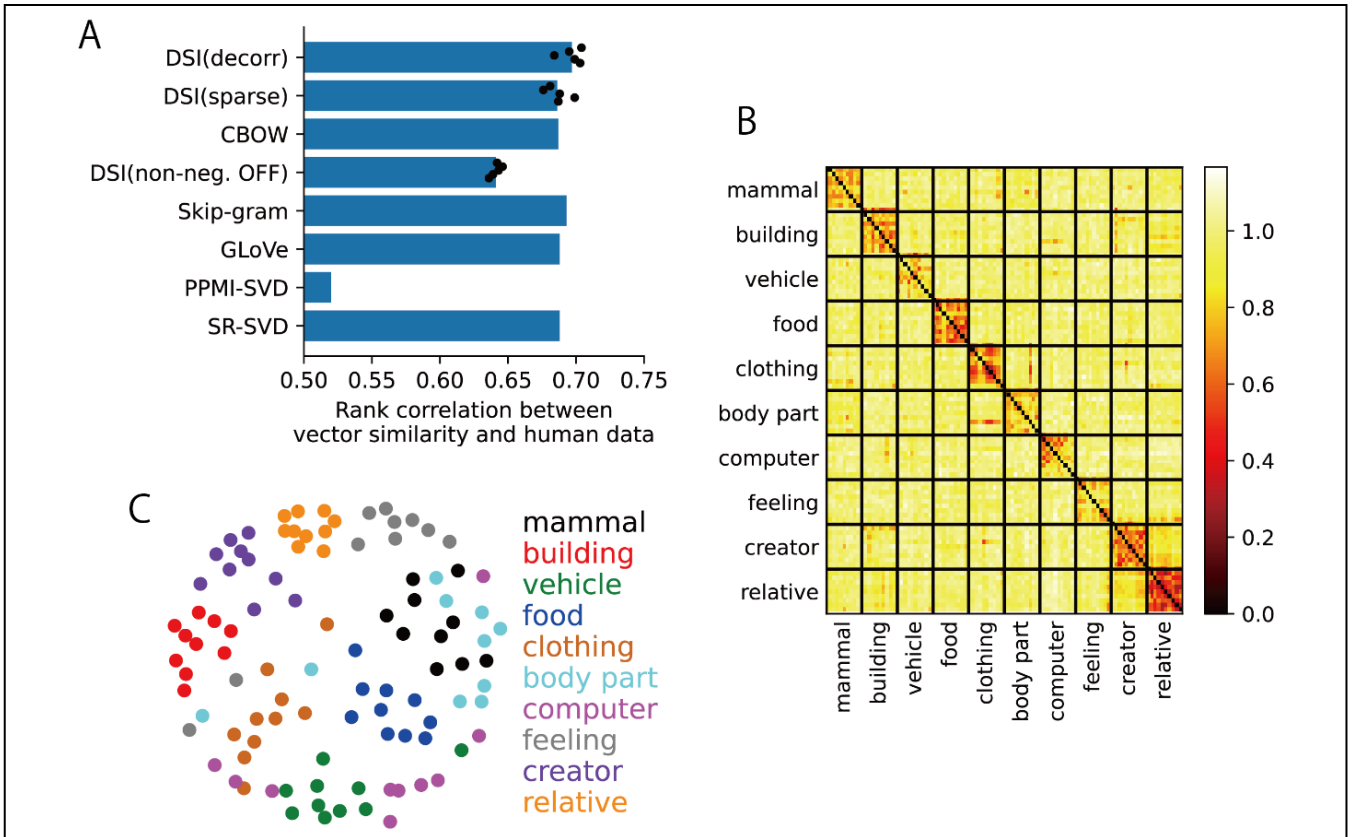

Figure 5: DSI representations capture the semantic structure of words at the population level (DSI-decorr). (A) The rank correlation of word similarity evaluated by word representation vectors (cosine similarity) and humans (WS353 dataset). For DSI, dots indicate 5 trials with different random seeds (different initial values for learning); bars indicate means of those 5 simulations. (B) Dissimilarity matrix between DSI representation vectors for 100 words in 10 semantic categories (DSI-decorr). We selected 10 words in each category. We used same dissimilarity metric with Reber et al. (2019) (1 - Pearson's correlation coefficient). (C) Visualization of the representational structure of DSI using MDS based on the dissimilarity matrix shown in (B). Each color corresponds to a semantic category.

DSI showed high correlations that are comparable to word embedding models (Figure 5A). This indicates that DSI captures the semantic structure of words at the level of population activity patterns.

This property suggests that the representational structure of DSI vectors is consistent with the experimental observation that population-level pattern similarity of concept cell activities represents semantic categories[26]. We considered 10 semantic categories and chose 10 words corresponding to each category (see Methods for the choice of words) and evaluated dissimilarity between DSI representation vectors of those 100 words using the same metric with the experimental work[26] (1 - Pearson's correlation coefficient). Dissimilarity matrix in Figure 5B and Supplementary Figure 5 clearly shows that representations were similar within each semantic category. In the visualization of the representational structure by multidimensional scaling (MDS), we can also see clustering of words corresponding to 10 semantic categories (Figure 5C and Supplementary Figure 5).

Analogical inference by partial recombination of assemblies of conceptual units

So far, we have analyzed how DSI represents spaces and linguistic concepts. In this section, we discuss how DSI can support computation of linguistic concepts. Specifically, we argue that DSI provides a biological interpretable mechanism for inference of words. Previous studies in NLP[30,31] have shown that arithmetic calculation of word representation vectors enables inference of words based on analogical relationships. For example, when we consider an analogical relationship "man is to woman as king is to queen", the calculation of word representation vectors as $\vec{x}(king) + \vec{x}(woman) - \vec{x}(man)$ gives a similar vector to $\vec{x}(queen)$. For this analogical inference task, DSI vectors show comparable performance with conventional word embedding models (evaluation by Mikolov's dataset[30,31]; 3157/19544 questions were used; see Methods section for details) (Figure 6D, success rates at "300 (Full)"). However, unlike the conventional word embedding, each word is represented by a combination of concept-specific units (a population activity pattern of concept cells) in DSI. In this case, we can interpret its analogical inference as a partial recombination of those assemblies to create novel word representations. Below, we give an intuitive explanation of this property of DSI.

First, we analyzed the ratio of each element to the sum of all elements in DSI vectors. We found that even the largest element accounted for 5% of the sum of all elements on average in the case of DSI-decorr (Supplementary Figure 6). This implies that DSI vectors for words are non-sparse and distributed; thus, each word is represented by a combination of multiple conceptual units.

Next, we inspected representations of a set of words as an example: France, Paris, Germany and Berlin. In those words, we can see two analogical relationships (country-capital and French-German relationships). We identified the most active units in DSI vectors for those words and listed the TOP-10 words for identified units. We could find four representative units that we could name as "German cell", "French cell", "country cell" and "capital cell" (Figure 6A). We could see that $\vec{x}(France)$ is represented by the combination of the French cell and the country cell, whereas $\vec{x}(Berlin)$ is represented by the combination of the German cell and the capital cell, and so on (Figure 6B). This example gives a simple interpretation of word similarity in the DSI vector space discussed in the previous section. If words are similar, they share a large number of active conceptual units, like the country cell shared by representations of France and Germany. Thus, semantic similarity between words increases cosine similarity between word vectors.

Then, the difference vectors between words correspond to switching on and off conceptual units. For example, summing the difference vector of $\vec{x}(Germany)$ and $\vec{x}(Berlin)$ can turn on the capital cell and turn off the country cell (Figure 6B). Similarly, transformation from $\vec{x}(France)$ to $\vec{x}(Paris)$ corresponds to the activation of the capital cell and deactivation of the country cell, enabling this transformation by summing the difference of $\vec{x}(Germany)$ and $\vec{x}(Berlin)$. Thus, we can biologically interpret the procedure of analogical inference as a partial recombination of an assembly of concept cells (Figure 6C). If neural circuits in HPC and EC have learned transformation of $\vec{x}(Germany)$ to $\vec{x}(Berlin)$ by sparse switching of concept cells, the same operation enables $\vec{x}(France)$ to $\vec{x}(Paris)$. Here, the mathematical mechanism of the analogical inference is the same as the conventional word embedding models. However, a unique feature of our DSI representations is that those analogical relationships are factorized into separated units, making the modulation of a few units enough to

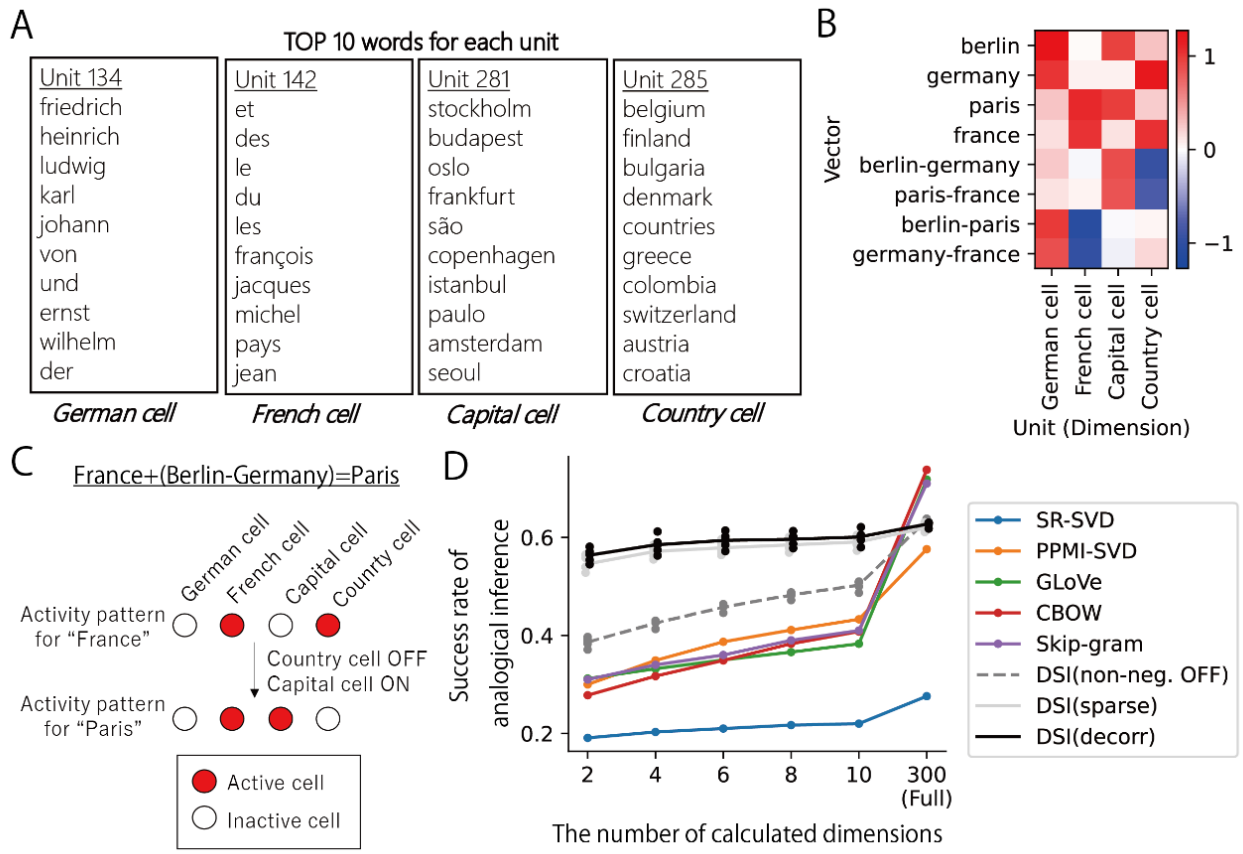

Figure 6: Analogical inference as a sparse recombination of conceptual units. (A) TOP-10 words of four representative units obtained by DSI- decorr and their interpretation. (B) Values of four representative units in DSI vectors for France, Germany, Paris, Berlin, and difference vectors between them. (C) Interpretation of analogical inference as sparse recombination of concept cells. (D) Success rates of analogical inference task (Mikolov's dataset) by calculating the limited number of dimensions in word representation vectors. For DSI, dots indicate 5 trials with different random seeds (different initial values for learning); lines indicate means of those 5 simulations. Success rates of DSI (decorr) were not significantly different from DSI (sparse) ($P>0.05$, 2-sample t-test) but higher than all other methods in the condition of 2, 4, 6, 8, and 10 dimensions. ($P<0.01$, 2-sample t-test for DSI (non-neg. OFF) and 1-sample t-tests for other word embedding methods). All statistical tests were two-sided t-tests and significance thresholds were modified by Bonferroni correction. Details of statistical analyses are shown in Supplementary Table 2.

perform the inference. We speculate that constraints in our dimension reduction method (especially non-negativity) are sufficient conditions to align each semantic factor to each axis of the vector space.

Next, we check whether the property of DSI mentioned above is general across a wide repertoire of words. If each analogical relationship is actually confined to a specific dimension, switching on and off only a few dimensions is enough for the inference, and the calculation of a whole vector is unnecessary. We tested this hypothesis by an analogical inference task using Mikolov's dataset, limiting the number of calculated dimensions. Namely, when we calculated $\vec{x}(France)+\vec{x}(Berlin)-\vec{x}(Germany)$ to obtain $\vec{x}(Paris)$, we

identified a few dimensions that had maximum and minimum values in $\vec{x}(Berlin) - \vec{x}(Germany)$ (that are "country cell" and "capital cell"). We summed only those dimensions to $\vec{x}(France)$. This partial calculation of vectors did not degrade the overall task performance of DSI-decorr and DSI-sparse (even 2 out of 300 dimensions did not largely alter the performance) (Figure 6D). In contrast, the performance of other word embedding models was significantly impaired (Figure 6D). This result suggests that the partial recombination of concept-specific units enabled the analogical inference of various words, and this property is unique to DSI model. Removal of the non-negativity constraint significantly impaired the performance of DSI-decorr, which further suggests the importance of non-negativity (Figure 6D). Notably, CBOW model, which showed high conceptual specificity (Figure 3), did not work properly in this inference task with partial calculations (Figure 6D). This suggests that apparent conceptual specificity is not a sufficient condition for such calculation.

Analogical inference of spatial contexts by modulating non-grid spatial representations

In the previous section, we discussed analogical inference of words. We found that we can apply the same computational framework to spatial representations. Furthermore, our results suggest a novel computational role of non-grid（heterogenous) spatial representations that have been experimentally found in EC[34,50].

First, we constructed DSI representation vectors (DSI-decorr) in spatial contexts A, B, and Φ, which have different barrier layouts (Figure 7A, Supplementary Figure 7). Then, we created composite representation vectors for a novel context A+B by simply combining the spatial representation vectors for familiar contexts A, B, and Φ as "A+B-Φ" (Figure 7A, see Methods for details). We devised this computation based on the analogical inference of words: the relationship of Φ to B corresponds to the relationship of A to A+B (whether a barrier exists at a specific position or not). We tested goal-directed spatial navigation (as described in the previous section) in three spatial contexts A, B, and A+B, by using the learned representations for A, B, and Φ, and those generated for A+B. Naturally, the representation vectors for A and B gave the best performance in contexts A and B, respectively (Figure 7B and 7C). Remarkably, the composite representation vector "A+B-Φ" achieved the best performance in the context A+B (Figure 7B and 7C). This result corresponds to the vector computation for word inference, suggesting that the computational framework for linguistic concepts is also valid for inferring the spatial contexts. We confirmed that this inference also worked in another spatial setting (Supplementary Figure 8). However, the inference did not improve spatial navigation when two barriers were spatially connected in the context A+B (Supplementary Figure 9). We speculate that this simple computational scheme works only if the interaction between barriers is small. Therefore, we conducted further analyses below in the settings used in Figure 7 and Supplementary Figure 8.

We hypothesized that the relationship between spatial contexts is confined to the small number of units (dimensions) as in word representations. If this is the case, representational modulations of only a few units are enough for the inference. We identified a few units that had the largest representational differences (see Methods for the definition of representational distance) between the spatial contexts B and Φ, and calculated only those units to compose representation vectors for the context A+B. We found that minimally 4 units were enough to obtain the asymptotic performance (Figure 7D and 7E). We inspected the spatial representations of these units and found that all of them were non-grid representations in the context Φ (no barriers) where 42.6±5.2 % of units were grid representations according to gridness analyses (Figure 7F). Consistently, we

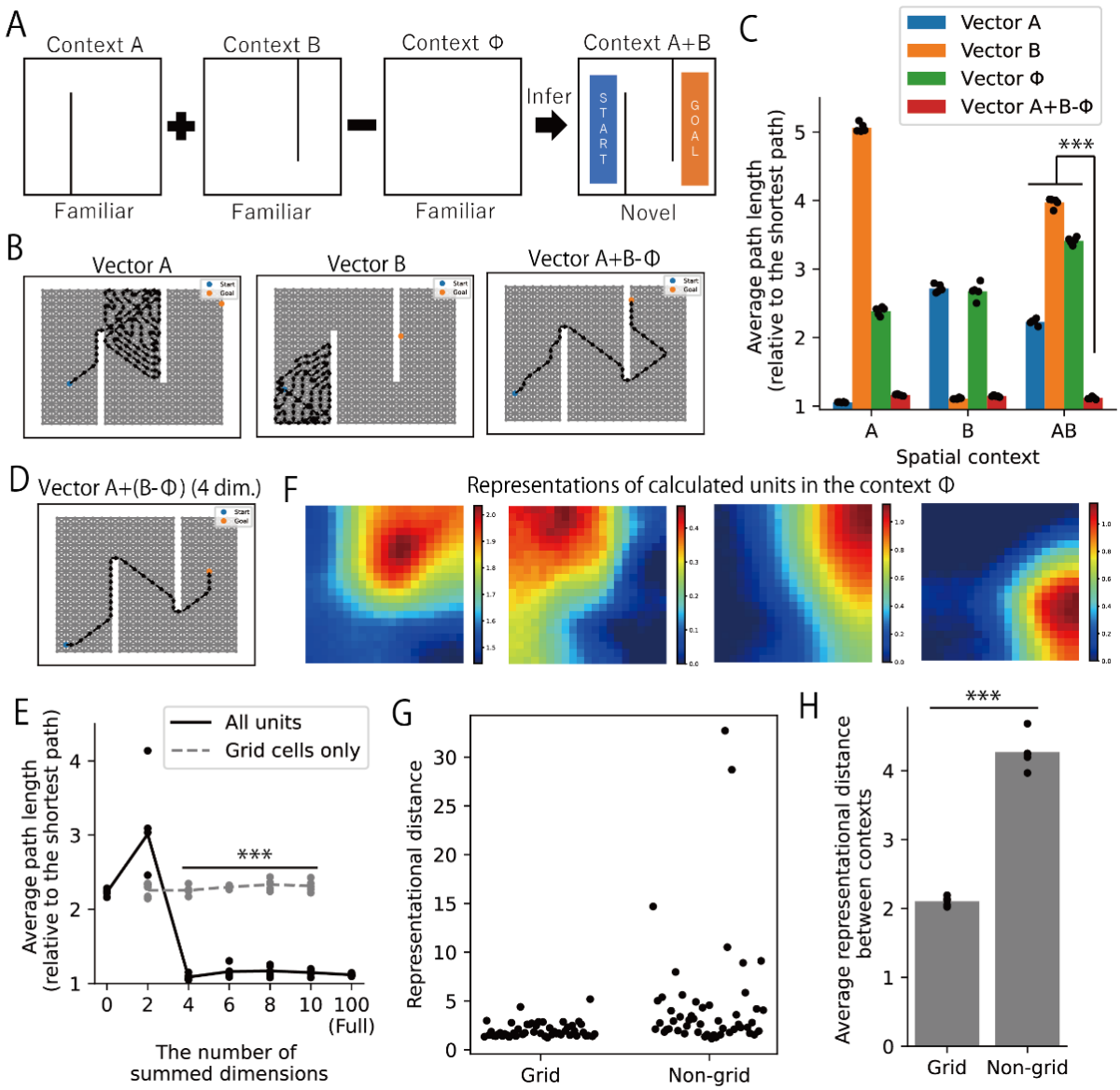

Figure 7: (Legend is in the next page)

found large representational distances selectively in non-grid-type units (Figure 7G), and the average representational distance of non-gird-type units is larger than that of grid-type units (Figure 7H and Supplementary Figure 8). Strikingly, when we restricted calculation of the vectors to grid-type units (excluding non-grid-type units), the obtained composite representation vectors did not improve spatial navigation in the context A+B (Figure 7E). We obtained qualitatively same results in the other spatial setting (Supplementary Figure 8). These results suggest that information of spatial contexts (barrier settings in this case) is confined to the limited number of units that have non-grid spatial representations, and the modulation of non-grid representations is sufficient for inferring novel spatial contexts.

These results imply that the modulation of non-grid spatial representations is more important for coding differences between spatial contexts than that of grid representations. This is consistent with the previous

(Legend of the figure in the previous page)
Figure 7: Composite spatial representations enable navigation in a novel spatial context. (A) We constructed representation vectors for a novel context A+B by arithmetic composition of DSI representation vectors for three familiar contexts A, B, and Φ. The start and the goal in each navigation trial were randomly positioned in the colored area. (B) Example spatial paths by spatial navigation using representation vectors learned in the context A (left), B (middle), and composite representation vectors A+B-Φ (right). (C) Average path lengths in 1,000 trials of spatial navigation under various settings of representation vectors and contexts. Note that we normalized a path length by the shortest path length between the start and the goal in each trial. Dots indicate 5 simulations with different random seeds (different initial values for learning and simulations); bars indicate means of those 5 simulations. (D) An example of spatial navigation performed by representation vectors created by calculating only 4 dimensions of the vector A. (E) Average path lengths by the spatial navigation using the composite representation vectors in which we summed the given number of dimensions. Dots indicate 5 simulations with different random seeds (different initial values for learning and simulations); lines indicate means of those 5 simulations. 2-sample t-tests were performed between the condition in which we calculated only grid cells and the condition in which we calculated all units. (F) Spatial representations of four calculated units in the context Φ in an example of (D). Note that all these units are non-grid representations. (G) Representational distances between the context B and Φ of grid-type and non-grid-type units in a simulation. Each dot indicates a unit. (H) Average representational distances between the context B and Φ of grid-type and non-grid-type units. Dots indicate 5 simulations with different random seeds (different initial values for learning and simulations); bars indicate means of those 5 simulations. ***P<0.001. All statistical tests were two-sided t-tests and significance thresholds were modified by Bonferroni correction. Details of statistical analyses are shown in Supplementary Table 3.

experimental finding that non-grid spatial representations are remapped more significantly than grid cells across different spatial contexts[34]. Our model provides a novel theoretical interpretation for this experimental observation, linking the computation of spatial contexts to that of linguistic concepts.

Discussion

In this paper, we proposed a theoretically interpretable and biologically plausible neural representation model for spatial navigation and linguistic concepts. We demonstrated that our DSI model forms spatially local or hexagonal grid representations for the 2-D space and concept-specific representations for the linguistic inputs, which are assumed to correspond to neural representations in HPC and EC. Furthermore, we showed that SI is mathematically related to reinforcement learning and word embedding, thus DSI representations support spatial navigation and NLP. Finally, the same computational framework based on partial representational modulation enables the inference of words and spatial contexts, which can be biologically interpreted as the function of concept cells and non-grid cells. These results suggest that we can extend the spatial representation model of HPC and EC to learn and compute linguistic concepts, which apparently seems a different computational domain from spatial navigation, in an intuitive and biologically plausible manner.

Theory of grid and non-grid representations in EC

Our model produces biologically plausible grid-like representations in 2-D space, which supports path integration and spatial navigation. Previous studies have revealed that nonnegative and orthogonal constraints are important to obtain realistic grid-like representations[35,36,52]. Furthermore, recurrent neural networks form grid-like representations through learning path integration, and those representations support efficient spatial navigation[36–40]. Some of those models have reproduced experimentally observed scaling ratios between grid cell modules[37]. It has also been shown that a unified model for spatial and non-spatial cognition generates grid representations [46,47,49]. Although our model was built on the basis of those previous works, previous models have not been applied to learning of linguistic concepts or other complex conceptual spaces in real-world data. Furthermore, our results revealed that the computational framework for linguistic concepts (analogical inference) can be applied to spatial representations, and non-grid spatial representations were important for the inference. Our model extended the range of applicability of biologically plausible spatial representation models to linguistic concepts and suggests shared computational mechanism between those two computational domains.

Furthermore, our model suggests the importance of non-grid spatial representations for inference (or switching) of spatial contexts. In EC, there are many neurons that exhibit non-grid spatial representations, and they show large remapping across contexts[34,50]. Previous models have shown that non-grid (heterogenous) spatial representations in EC support precise spatial encoding[36] and reward-based modulation of path integration[40]. Our results give alternative explanation of such remapping: such remapping can support analogical inference of navigation strategy across spatial contexts. However, it is still to be investigated whether this computational framework extends to general contextual changes such as wall colors, odors and reward settings[34,40,50]. We speculate that this is possible if we can appropriately construct disentangled representations for general sensory and task variables [53,62,63].

Implications for conceptual representations in the brain

Models of concept cells have mostly assumed sparsity and clustering to create conceptual specificity[41,42,46,47,54]. In contrast, our results suggest that non-negativity is essential to create functionally useful concept-specific representations. Non-negativity is also crucial to generate biologically plausible grid representations[35,36,52], thus our model suggests a common constraint across spatial and conceptual representations in HPC and EC. A recent theoretical finding suggests that non-negativity and energy minimization enable disentanglement of visual, spatial, and task variables from mixed inputs [53]. Our results suggest that such strategy also works in the case of linguistic concepts, and it enables a simple framework for the inference in the brain.

The biological interpretation of our model predicts a population-level property of concept cells. When a human subject (or an animal) learns a novel word (or an item) that can be interpreted by the composition of previously learned concepts, the representation for the novel item can be created by partial recombination of previous formed concept cells, instead of creating novel concept cells. Concept cells exhibit concept-specific activities, but they can simultaneously work as building blocks to represent specific items as assemblies[25,26]. Thus, recombination of those assemblies is useful for creation (or prediction) of novel conceptual representations. This strategy is beneficial in term of energy efficiency: the brain can reduce the number of neurons for processing by using factorized representations, and switching only a part of assemblies requires a smaller

number of synapses and plastic changes than overall control of the neural population. Our model confirmed that such efficient computational strategy actually works for linguistic concepts learned from the text data in the real world.

We interpreted conceptual units in our model as concept cells in HPC and EC because of the relationship with place cells and grid cells. However, we can also interpret these units as "semantic features" (attributes) in a broad sense[43–45,64]. A previous study has shown that distributed word representations composed of combinations of semantic features are useful for decoding words from fMRI activity patterns[64]. In the experiment, each semantic feature is associated to a brain-wide activity pattern, which implies that conceptual units in our model may also be related to brain areas other than HPC and EC. It has been also shown that skip-gram representations support high-performance decoding of semantic information from fMRI data[65].Although we specifically considered linguistic inputs in our study, semantic features in the brain can also emerge through interactions of multimodal sensory features across brain areas[43–45]. HPC and EC may be involved in that process, and the role of HPC and EC in brain-wide semantic learning should be investigated in the future.

Neural mechanism for natural language processing in the brain

Our model relates NLP, especially word embedding to conceptual representations in HPC and EC. Concept cells respond to words (both auditory inputs and texts)[25,27], and the relationship between HPC and language processing have been experimentally found[66,67]. Regarding word embedding, a previous study showed that hippocampal theta oscillation codes semantic distances between words measured in word2vec (skip-gram) subspace [68]. These experimental results support the contribution of HPC to language processing and word representations.

Recent studies have shown that representations in transformer-based models[69] such as GPT[70] achieve remarkable performance in linear fitting to neural recording during language processing[71,72]. Furthermore, it was recently shown that transformer-based models generate grid-like representations when applied to spatial learning[48]. Similarly to our model, this finding implies the relationship between spatial and linguistic processing in the brain although concept-specific representations has not been found in transformer-based models. A major difference between our DSI model and transformer-based models is that DSI representations are basically fixed (static embedding) whereas transformer-based models flexibly create context-dependent representations (dynamic embedding). Computation of concepts obviously depends on the context; thus activities of concept cells are dependent on contexts such as memory contents and task demands [27]. Therefore, our DSI model should be extended to process context-dependence, hopefully by combination with other models for learning context-dependent latent cognitive states[49,73,74].

Relationship with disentangled visual representation learning

In our dimension reduction method, we introduced non-negativity, sparsity, decorrelation and regularization as constraints. Those constraints are also important for extraction of independent components[53,75,76], and it is known that imposing independence in latent spaces of deep generative models results in the emergence of disentangled representations for visual features[53,63]. As those disentangled visual representations explain single-cell activities in higher-order visual cortex[63], we similarly interpreted disentangled word representations

in our model as concept cells. Concept cells respond to a specific concept, whereas population-level activity patterns represent abstract semantic structures[26]. Such property is consistent with the factorized and distributed nature of disentangled representation vectors. Our model provides the view that concept cells emerge as disentangled representations for semantic concepts.

Methods

Calculation of SR

Throughout this study, we used a direct count method because we performed only offline processing of finite data. In a sequence of states $\{s_1, \dots, s_t, \dots, s_T\}$, we recursively calculated exponential traces of past states $z(s,t) = \sum_{\tau=0}^{t-1} \gamma^{\tau} \delta(s_{t-\tau}, s)$ as

$$z(s,t) = \gamma z(s, t-1) + \delta(s_t, s) \tag{8}$$

and calculated SR from state counts and coincidence counts as

$$SR(s,s') = \frac{\sum_{t=1}^{T} z(s,t)\, \delta(s_t, s')}{\sum_{t=1}^{T} \delta(s_t, s)} \tag{9}$$

Details of dimension reduction for DSI vectors

To obtain DSI-decorr, we iteratively updated D-dimensional vectors $\vec{x}(s)$ and $\vec{w}(s)$ ($D < N_s$) by Nesterov's accelerated gradient descent method[77] to minimize the following objective function, rectifying all elements every iteration ($\forall i, x_i(s) \geq 0, w_i(s) \geq 0$).

$$\begin{aligned} J_{decorr} = \frac{1}{2} \sum_{s,s'} \rho(s,s') \big(PSI(s,s') - \vec{x}(s) \cdot \vec{w}(s') \big)^2 + \frac{1}{2} \beta_{cor} \sum_{i \neq j} \big(Corr(i,j) \big)^2 \\ + \frac{1}{2} \beta_{reg} \sum_{s} (\|\vec{x}(s)\|^2 + \|\vec{w}(s)\|^2) \end{aligned} \tag{10}$$

The first term of this objective function is weighted approximation error minimization, the second term works for decorrelation between dimensions, and the third term regularizes representation vectors. In this function, $\rho(s,s')$ is a weight for the square error

$$\rho(s,s') = \frac{1}{N_s V} \left(\frac{1}{M} PSI(s,s') + \rho_{min} \right) \tag{11}$$

where M and V are mean and variance of PSI, respectively, and ρ_{min} is a small value to avoid zero-weight. $Corr(i,j)$ is a correlation between two dimensions in $\vec{x}(s)$

$$Corr(i,j) = \frac{\sum_s \tilde{x}_i(s)\tilde{x}_j(s)}{\sqrt{\sum_s \left(\tilde{x}_i(s)\right)^2}\sqrt{\sum_s \left(\tilde{x}_j(s)\right)^2}} \tag{12}$$

where $\tilde{x}_i(s) = x_i(s) - \frac{1}{N_s}\sum_s x_i(s)$.

Gradients of the objective functions are

$$\begin{aligned}\frac{\partial J_{decorr}}{\partial x_k(s)} = &-\sum_{s'} \rho(s,s')\left(PSI(s,s') - \vec{x}(s)\cdot\vec{w}(s')\right) w_k(s') \\ &+\beta_{cor}\sum_{j\neq k}\frac{Corr(k,j)\tilde{x}_j(s)}{\sqrt{\sum_s \left(\tilde{x}_k(s)\right)^2}\sqrt{\sum_s \left(\tilde{x}_k(s)\right)^2}} + \beta_{reg}x_k(s)\end{aligned} \tag{13}$$

$$\frac{\partial J_{decorr}}{\partial w_k(s')} = -\sum_{s'} \rho(s,s')\left(PSI(s,s') - \vec{x}(s)\cdot\vec{w}(s')\right) x_k(s) + \beta_{reg}w_k(s') \tag{14}$$

We note that we regarded mean and variance of $x_k(s)$ in the correlation ($\sum_s \left(\tilde{x}_k(s)\right)^2, \sum_s \left(\tilde{x}_k(s)\right)^2$ in Eq. 6) as constants in the calculation of these gradients. Practically, this heuristic did not affect the performance of decorrelation.

To obtain DSI-sparse, we performed the same procedure with DSI-decorr but we changed the objective function to

$$J = \frac{1}{2}\sum_{s,s'} \rho(s,s')\left(PSI(s,s') - \vec{x}(s)\cdot\vec{w}(s')\right)^2 + \beta_{sparse}\sum_{s,k}\left(x_k(s) + w_k(s)\right) \tag{15}$$

We note that the second term of this function is equivalent to a L-1 constraint because $x_k(s)$ and $w_k(s)$ are non-negative. Gradients are

$$\frac{\partial J_{sparse}}{\partial x_k(s)} = -\sum_{s'} \rho(s,s')\left(PSI(s,s') - \vec{x}(s)\cdot\vec{w}(s')\right) w_k(s') + \beta_{sparse} \tag{16}$$

$$\frac{\partial J_{sparse}}{\partial w_k(s')} = -\sum_{s'} \rho(s,s')\left(PSI(s,s') - \vec{x}(s)\cdot\vec{w}(s')\right) x_k(s) + \beta_{sparse} \tag{17}$$

Throughout this paper, the learning rate was 0.05 and the number of iteration was 10000. Parameters were β_{cor}=1, $\beta_{reg} = 0.001$, β_{sparse}=0.0001, and $\rho_{min} = 0.001$.

Mathematical relationship of DSI and reinforcement learning

In this section, we show that our model approximates value functions of linear reinforcement learning[22–24] in the setting of spatial navigation.

In linear reinforcement learning, an agent aims to maximize "gain" instead of reward. Assuming a default policy $\pi^d(s)$ (any policy is available; typically random walk in the case of exploration task), gain function is defined as

$$g(s) = r(s) - \lambda KL\left(\pi(s)|\pi^d(s)\right) \tag{18}$$

where $r(s)$ is expected reward at the state s and $\lambda KL\left(\pi(s)|\pi^d(s)\right)$ is the cost imposed on the difference between the current policy $\pi(s)$ and the default policy $\pi^d(s)$ (λ is a relative weight of the cost). Then, previous works have shown that the optimal policy and corresponding value functions can be determined explicitly by solving linear equations[22–24]. Here we consider an environment that consists of N_N nonterminal states and N_T terminal states. We define two transition probability matrices under the default policy: P_{NT} is a $N_N \times N_T$ matrix for transitions from non-terminal states to terminal states, and P_{NN} is a $N_N \times N_N$ matrix for transitions across non-terminal states. Furthermore, $\vec{r}_N$ and $\vec{r}_T$are vectors of rewards at non-terminal states and terminal states, respectively. In this condition, a vector of value functions under optimal policy $\vec{v}^* = \left(v^*(s_1), \dots, v^*(s_{N_N})\right)$ is obtained as

$$\exp(\lambda^{-1}\vec{v}^*) = \mathrm{M}\mathrm{P}_{NT}\exp(\lambda^{-1}\vec{r}_T) \tag{19}$$

where $\mathrm{M} = (\mathrm{diag}(\exp(-\lambda^{-1}\vec{r}_N)) - \mathrm{P}_{NN})^{-1}$ is DR[24].

To relate $\vec{v}^*$ to SI, we consider a specific condition in which the environment consists of non-terminal states, and a virtual terminal state is attached to a goal state s_G arbitrarily chosen from non-terminal states. When the agent gets to the goal, it transits to the terminal state. Furthermore, we assume that rewards at non-terminal states are uniformly negative and reward at the terminal state is positive so that the agent has to take a short path to goal to maximize reward. Specifically, we assume all elements of $\vec{r}_N$ are r_-, and $r_T = r_+$ where r_- and r_+ is arbitrary negative and positive values, respectively. Then, we obtain

$$\exp(\lambda^{-1}\vec{v}^*) = (\mathrm{I} - \exp(\lambda^{-1}r_-)\,\mathrm{P}_{NN})^{-1}\vec{e}(i_G)\exp\left(\lambda^{-1}(r_+ + r_-)\right) \tag{20}$$

where I is an identity matrix, $\vec{e}(i_G)$ is a standard unit vector in which i_G-th element is 1, i_G is the index of the goal state. Because $0 < \exp(\lambda^{-1}r_-) < 1$, $(\mathrm{I} - \exp(\lambda^{-1}r_-)\,\mathrm{P}_{NT})^{-1}$ is equivalent to a successor representation matrix with a discount factor $\gamma = \exp(\lambda^{-1}r_-)$ [16,18]. Therefore, we obtain

$$\exp\left(\lambda^{-1}v^*(s)\right) = SR^d(s, s_G)\exp\left(\lambda^{-1}(r_+ + r_-)\right) \tag{21}$$

where $SR^d(s, s_G)$ is SR under the default policy. By setting $r_+ = -r_- - \lambda \log P^d(s_G)$ ($P^d(s_G)$ is a probability of visiting the state s_G under the default policy), we finally obtain

$$\lambda^{-1}v^*(s) = \log SR^d(s, s_G) - \log P^d(s_G) = SI^d(s, s_G) \approx \vec{x}(s)\cdot\vec{w}(s_G) \tag{22}$$

where $SI^d(s, s_G)$ is SI under the default policy.

Learning DSI in 2-D spaces

As an environment, we assumed a square room (or interconnected separated rooms) tiled with 30 × 30 discrete states. In each simulation trial, an agent starts at one of those 900 states and transits to one of eight surrounding states each time except that transitions are limited at states along the boundary (the structure was not a torus). Transitions to all directions occur with an equal probability. We performed 500 simulation trials and obtained a sequence of 100,000 time steps in each trial. We calculated occurrence probabilities ($P(s)$) and a successor representation matrix ($SR(s,s')$) of 900 states from those sequences, and calculated PSI and DSI (100-dimensional) as described in the Model section. The discount factor γ was set to 0.99.

We repeated this procedure five times using different random seeds which gave different representations each time, to report reproducibility or statistics of results.

Quantitative evaluation of grid representations

In the section 4.2, we performed the gridness analysis following a previous experimental study[55]. For each unit, we rotated the spatial autocorrelation map (Figure 2B, lower) and calculated correlations between the original and rotated maps. Gridness was defined as the difference between the lowest correlation at 60° and 120° and the highest correlation at 30°, 90° and 150°. A unit was classified as a grid cell when gridness exceeds zero.

In Figure 2C, we constructed a distribution of grid scales. Grid scales were determined as the median of distances between the central peak and the six closest peaks (vertices of inner hexagon) in the spatial autocorrelation map. The kernel function for kernel density estimation was Gaussian with a standard deviation 1.

Path integration by DSI vectors

We made an estimate of the next representation vector $\vec{z}_{t+1}$ by linear transformation of the current representation vector $\vec{x}(s_t)$ as

$$\vec{z}_{t+1} = \mathrm{M}(a_t)\vec{x}(s_t) \tag{23}$$

where a_t represents a movement (one of eight directional movements in this study) and $\mathrm{M}(a_t)$ is movement-conditional recurrent weight matrix. Here, $\vec{x}(s_t)$ was a DSI representation vector, and we optimized the matrix $\mathrm{M}(a_t)$ by minimizing prediction error $\|\vec{z}_{t+1} - \mathrm{M}(a_t)\vec{x}(s_t)\|_2^2$ by stochastic gradient descent during random walk on the state transition graph (20 simulation trials of 100,000 time steps). After optimization, we set an initial state s_0 and a sequence of movements $\{a_0, a_1, \cdots, a_{T-1}\}$, and performed path integration by recursive estimation $\vec{z}_{t+1} = \mathrm{M}(a_t)\vec{z}_t$. We determined a position at each time step by searching a state representation vector that has minimum Euclidian distance with the estimated vector ($s_t = \mathrm{argmin}_s \|\vec{x}(s) - \vec{z}_t\|_2$).

Goal-directed spatial navigation by DSI vectors

We assume that a start location (state s_{init}) and a goal location (state s_G) are randomly given in each trial such that the shortest path length is minimally 10, and an agent has to navigate between them. Suppose that the agent exists at a state s, and a set of neighboring states of s is $A(s)$. Given the goal representation vector $w(s_G)$, a value function of a neighboring state $s_{next} \in A(s)$ is estimated by $x(s_{next}) \cdot w(s_G)$, and the agents

transits to the state that has a maximum value. Because of the relationship between value functions and DSI, this strategy approximate value-based decision making for goal-directed navigation.

However, because of approximation, the agent sometimes deviated from optimal paths. Specifically, the agent was sometimes trapped in local loops of a few adjacent states, which significantly impaired the performance. To avoid that, we heuristically introduced a "familiarity" variable $f(s)$ to each state. This variable was initially set to zero and incremented by one each time the agent visit the state s. Then, value was evaluated by $x(s_{next}) \cdot w(s_G)$- $f(s)$, which facilitates escape from local loops. Note that this strategy alone was not enough for appropriate goal-directed navigation (see results in Figure 7 for example).

In the evaluation, we stopped simulation at 100 time steps in each trial, and we considered the path length was 100 in the trial.

Learning DSI from text data

We used text data taken from English Wikipedia dump (enwiki-latest-pages-articles, 22-May-2020). We first generated text files from raw data using wikiextrator (https://github.com/attardi/wikiextractor). We tokenized texts by nltk Punkt sentence tokenizer, and randomly sampled 100,000 articles containing 1,000 tokens at minimum. We lowercased all characters and removed punctuation characters in the data. After that, we selected words that appeared more than 1,000 times in the data and substituted all other rare words by <unk> symbol. Finally, we obtained data that contains 124M tokens and 9376 words (i.e. 9376 states).

The procedure of calculating DSI was same as the case of 2-D spaces except that the discount factor γ was set to 0.9. The setting of other parameters was the same as the experiment of 2-D spaces. Using the preprocessed data, we repeated learning of DSI vectors five times using different random seeds (i.e. different initial values for optimization) which gave different representations each time, to report reproducibility or statistics of results.

Comparison with other word embedding methods

We compared DSI with other word embedding models applied to the same text data under same dimensionality (300 dimensions). For skip-gram and continuous-bag-of-words (CBOW)[30,31], we used implementation in Python Gensim library. For GLoVe[32], we used implementation by Pennington et al. (https://github.com/stanfordnlp/GLoVe). PPMI-SVD[33] and SR-SVD[18] were implemented by ourselves. The window size of coincidence count was set to 10 in all models, and other parameters followed default settings or parameters described in original papers. The discount rate of SR-SVD were same as SR used for calculation of DSI ($\gamma = 0.9$). For BERT, we took representations in a pretrained model (bert-base-uncased in Hugging Face transformers)[59,60], thus training data and dimensionality (768 dimensions) were different from our setting.

Quantitative evaluation of conceptual specificity

In the section 5.2, conceptual specificity of each unit was evaluated using WordNet database[58]. In WordNet, a word belongs to several synsets (sets of cognitive synonyms), and semantic similarity of two synsets can be

evaluated from the shortest path length between them in the WordNet structure (we used path similarity function in nltk library). We defined similarity of two words as the highest similarity among all combinations of synsets of those words. We calculated mean similarity of all combinations of TOP-10 words (ten words that highly activated the unit; Figure 4A) that are available in WordNet. We evaluated only units which had at least five TOP-10 words available in WordNet. Furthermore, we randomly generated 1,000 pairs of words available in WordNet, and generated a null distribution of similarity between words. We defined a significance threshold of similarity as a 95 percentile of the null distribution, and a unit was classified as a significantly concept-specific unit if mean similarity of TOP-10 words exceeded the threshold. Furthermore, we quantitatively defined a conceptual specificity of each unit as

$$\frac{s_{unit}}{s_{null}} - 1 \tag{24}$$

where s_{unit} is mean similarity of TOP-10 words and s_{null} is the mean of the null distribution. This quantity becomes zero if similarity between TOP-10 words is not different from random pairs, and becomes positive if TOP-10 words are semantically similar. This conceptual specificity was averaged over all evaluated units.

Of note, we found several non-significant units exhibit conceptual specificity according to manual inspection (see Supplementary Figure 4). This is probably because of the limitation of knowledge covered by WordNet. Therefore, we suppose that the current evaluation method tends to underestimate the number of concept-specific units. However, the comparison across models was fair because we used the same procedure and criteria for all models.

Evaluation of the semantic structure of DSI vectors

To analyze the semantic structure in representation vectors, we considered ten semantic categories (“mammal”, “building”, “vehicle”, “food”, “clothing”, “body part”, “computer”, “feeling”, “creator”, “relative”) and chose ten words for each category as follows.

“mammal”: dog, cat, bull, bear, elephant, fox, horse, lion, rat, tiger

“building”: building, church, theater, hall, school, hotel, house, library, mansion, palace

“vehicle”: vehicle, ambulance, bike, aircraft, bus, car, helicopter, locomotive, rocket, ship

“food”: food, bread, cheese, candy, rice, beer, coffee, milk, tea, wine

“clothing”: clothing, shirt, hat, belt, costume, dress, wear, cap, coat, crown

“body part”: arm, bone, lung, ear, eye, finger, foot, hair, kidney, leg

“computer”: computer, software, internet, code, access, server, website, pc, node, portal

“feeling”: feeling, anger, anxiety, comfort, confusion, emotion, enthusiasm, happiness, joy, love

"creator": creator, architect, artist, composer, designer, producer, filmmaker, photographer, musician, painter

"relative": husband, brother, aunt, cousin, daughter, father, grandfather, grandmother, mother, sister

We took DSI representation vectors for these 100 words and computed dissimilarity (1-Pearson's correlation coefficient). MDS was performed using the same dissimilarity metric.

Analogical inference of words

In Mikolov's dataset, many sets of four words are given. There is a relationship "WORD1 is to WORD2 as WORD3 is to WORD4" (i.e. king is to queen as brother is to sister). Then, an expected relationship in the vector space is WORD2-WORD1=WORD4-WORD3. In this study, we performed inference of WORD4 by WORD3+WORD2-WORD1. We regarded an inference was correct if the actual vector of WORD4 had the largest cosine similarity to the inferred vector among all word representation vectors (except those for WORD1, WORD2, and WORD3). If the number of words is 10,000, a chance level of the correct answer rate is 0.01%. Therefore, the performance shown in this study (more than 50%) is far above the chance level.

When we performed analogical inference by calculation of the limited number of dimensions (partial recombination), we first calculated $\vec{x}(WORD2) - \vec{x}(WORD1)$ and then identified the n largest and the n smallest elements in that vector. We summed those $2n$ out of 300 elements to $\vec{x}(WORD3)$ (in other words, other elements of $\vec{x}(WORD2) - \vec{x}(WORD1)$ were substituted by zero before summation). Correctness of the inference was evaluated by the same way with the original task.

Analogical inference of spatial contexts

We considered a 2-D space tiled with 21x21 (441) states and four spatial contexts (barrier layouts) A, B, Φ, A+B (Supplementary Figure 7). We define separated states in contexts A, B, Φ, A+B as s_i^A, s_i^B, s_i^Φ and s_i^{A+B} where i is a positional index which indicates a same position in all contexts ($i = 1,2,\ldots,441$). We constructed representation vectors for contexts A, B, Φ through direct experiences, then we created $\vec{x}(s_i^{A+B})$ and $\vec{w}(s_i^{A+B})$ as

$$\vec{x}(s_i^{A+B}) = \vec{x}(s_i^A) + \vec{x}(s_i^B) - \vec{x}(s_i^\Phi) \tag{25}$$

$$\vec{w}(s_i^{A+B}) = \vec{w}(s_i^A) + \vec{w}(s_i^B) - \vec{w}(s_i^\Phi) \tag{26}$$

We performed spatial navigation in a given context using one of four representations $\{\vec{x}(s_i^A), \vec{w}(s_i^A)\}$, $\{\vec{x}(s_i^B), \vec{w}(s_i^B)\}$, $\{\vec{x}(s_i^\Phi), \vec{w}(s_i^\Phi)\}$, and $\{\vec{x}(s_i^{A+B}), \vec{w}(s_i^{A+B})\}$ for corresponding positions, following the vector-based decision making rule described in the previous section ("Goal-directed spatial navigation by DSI vectors").

To learn representations in contexts A, B, and Φ, we sampled sequences of $\{s_i^A\}_{i=1,\cdots,441}$, $\{s_i^B\}_{i=1,\cdots,441}$, and $\{s_i^\Phi\}_{i=1,\cdots,441}$ as in the previous section ("Learning DSI in 2-D spaces"). However, we performed random walk in the 2-D space while switching between context A, B, and Φ, that is, state transitions to the same position in

other contexts occurred every 5,000 time steps (transition between s_i^A and s_i^B, and s_i^Φ). It means that we assumed that the setting of barriers can change during the experience, and this transition was introduced to associate the same position in different contexts. We performed 500 simulation trials and obtained a sequence of 100,000 time steps in each trial. From sampled sequences, we calculated SR for all combinations of $\{s_i^A\}_{i=1,\cdots,441}$, $\{s_i^B\}_{i=1,\cdots,441}$, and $\{s_i^\Phi\}_{i=1,\cdots,441}$ and calculated 100-dimensional DSI vectors for 1,324 (441×3) states by simultaneous compression of all states (dimension reduction of a 1,324×1,324 PSI matrix). The discount factor γ was set to 0.99.

When we performed the inference by calculation of the limited number of dimensions, we first identified the n units (dimensions) that had the largest representational distances between the context B and Φ. The representational distance of the unit k (k-th dimension of vectors $x_k(s)$) was defined as

$$\sum_i \left(x_k(s_i^B) - x_k(s_i^\Phi)\right)^2 \tag{27}$$

We calculated $x_k(s_i^{A+B}) = x_k(s_i^A) + x_k(s_i^B) - x_k(s_i^\Phi)$ only for those n dimensions, and we set $x_k(s_i^{A+B}) = x_k(s_i^A)$ in other dimensions.

Statistical analysis

To ensure reproducibility, we showed results obtained by five repetitions of simulations with different random seeds (different initial values for learning and simulations) in Figure 4, Figure 6, Figure 7, Supplementary Figure 8 and 9. We applied two-sided t-tests to those five samples assuming normality of data, and detailed values of t-statistic and p-values are summarized in Supplementary Table 1, 2, and 3. Significance thresholds were modified by Bonferroni correction.

Code and data availability

Python codes for simulations and analyses are available at https://github.com/TatsuyaHaga/DSI_codes . We also used external codes at https://github.com/stanfordnlp/GLoVe (GLoVe) and https://github.com/attardi/wikiextractor (wikiextrator). Text data were taken from https://dumps.wikimedia.org/enwiki/latest/ but currently the version 22-May-2020 is not available. We share the version we used upon request. WS353 dataset is available at http://alfonseca.org/eng/research/wordsim353.html. Mikolov's dataset is available at https://aclweb.org/aclwiki/Google_analogy_test_set_(State_of_the_art).

Acknowledgments

We thank Ryo Yoshida for supporting the analysis of representations in BERT model. We also thank Robert M Mok for fruitful discussion. This research was partially supported by JSPS KAKENHI Grant Numbers 21K15611, 19H04994, 18H05213, and JST PRESTO Grant Number JPMJPR21C2.

Author contributions

T.H.: conceptualization, formal analysis, funding acquisition, investigation, methodology, software, visualization, writing-original draft, writing-review & editing. Y.O.: conceptualization, funding acquisition, supervision, writing-review & editing. T.F.: conceptualization, funding acquisition, resources, supervision, writing-review & editing.

Competing interests

Authors declare no competing interests.

References

1. Hafting, T., Fyhn, M., Molden, S., Moser, M. B. & Moser, E. I. Microstructure of a spatial map in the entorhinal cortex. *Nature* **436**, 801–806 (2005).
2. Doeller, C. F., Barry, C. & Burgess, N. Evidence for grid cells in a human memory network. *Nature* **463**, 657–661 (2010).
3. Jacobs, J. *et al.* Direct recordings of grid-like neuronal activity in human spatial navigation. *Nat Neurosci* **16**, 1188–1190 (2013).
4. O'Keefe, J. & Dostrovsky, J. The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat. *Brain Res* **34**, 171–175 (1971).
5. Bush, D., Barry, C., Manson, D. & Burgess, N. Using Grid Cells for Navigation. *Neuron* **87**, 507–520 (2015).
6. McNaughton, B. L., Battaglia, F. P., Jensen, O., Moser, E. I. & Moser, M. B. Path integration and the neural basis of the 'cognitive map'. *Nat Rev Neurosci* **7**, 663–678 (2006).
7. Burak, Y. & Fiete, I. R. Accurate path integration in continuous attractor network models of grid cells. *PLoS Comput Biol* **5**, (2009).
8. Constantinescu, A. O., O'reilly, J. X. & Behrens, T. E. J. Organizing conceptual knowledge in humans with a gridlike code. *Science (1979)* **352**, 1464–1468 (2016).
9. Bao, X. *et al.* Grid-like Neural Representations Support Olfactory Navigation of a Two-Dimensional Odor Space. *Neuron* **102**, 1066-1075.e5 (2019).
10. Park, S. A., Miller, D. S. & Boorman, E. D. Inferences on a multidimensional social hierarchy use a grid-like code. *Nat Neurosci* **24**, 1292–1301 (2021).
11. Viganò, S. & Piazza, M. Distance and direction codes underlie navigation of a novel semantic space in the human brain. *Journal of Neuroscience* **40**, 2727–2736 (2020).
12. Viganò, S., Rubino, V., Soccio, A. Di, Buiatti, M. & Piazza, M. Grid-like and distance codes for representing word meaning in the human brain. *Neuroimage* **232**, (2021).
13. Theves, S., Fernandez, G. & Doeller, C. F. The Hippocampus Encodes Distances in Multidimensional Feature Space. *Current Biology* **29**, 1226-1231.e3 (2019).
14. Aronov, D., Nevers, R. & Tank, D. W. Mapping of a non-spatial dimension by the hippocampal-entorhinal circuit. *Nature* **543**, 719–722 (2017).

15. Knudsen, E. B. & Wallis, J. D. Hippocampal neurons construct a map of an abstract value space. *Cell* **184**, 4640-4650.e10 (2021).
16. Dayan, P. Improving Generalization for Temporal Difference Learning: The Successor Representation. *Neural Comput* **5**, 613–624 (1993).
17. 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 Comput Biol* **13**, (2017).
18. Stachenfeld, K. L., Botvinick, M. M. & Gershman, S. J. The hippocampus as a predictive map. *Nat Neurosci* **20**, 1643–1653 (2017).
19. Momennejad, I. Learning Structures: Predictive Representations, Replay, and Generalization. *Curr Opin Behav Sci* **32**, 155–166 (2020).
20. Momennejad, I. *et al.* The successor representation in human reinforcement learning. *Nat Hum Behav* **1**, 680–692 (2017).
21. Garvert, M. M., Dolan, R. J. & Behrens, T. E. A map of abstract relational knowledge in the human hippocampal–entorhinal cortex. *Elife* **6**, (2017).
22. Todorov, E. Linearly-solvable Markov decision problems. in *Advances in Neural Information Processing Systems 19* 1369–1376 (2006).
23. Todorov, E. Efficient computation of optimal actions. *Proc Natl Acad Sci U S A* **106**, 11478–11483 (2009).
24. Piray, P. & Daw, N. D. Linear reinforcement learning in planning, grid fields, and cognitive control. *Nat Commun* **12**, (2021).
25. Quiroga, R. Q. Concept cells: the building blocks of declarative memory functions. *Nat Rev Neurosci* **13**, 587–597 (2012).
26. Reber, T. P. *et al.* Representation of abstract semantic knowledge in populations of human single neurons in the medial temporal lobe. *PLoS Biol* **17**, (2019).
27. Bausch, M. *et al.* Concept neurons in the human medial temporal lobe flexibly represent abstract relations between concepts. *Nat Commun* **12**, (2021).
28. Kreiman, G., Koch, C. & Fried, I. Imagery neurons in the human brain. *Nature* **408**, 357–361 (2000).
29. Gelbard-Sagiv, H., Mukamel, R., Harel, M., Malach, R. & Fried, I. Internally Generated Reactivation of Single Neurons in Human Hippocampus During Free Recall. *Science (1979)* **322**, 96–101 (2008).
30. Mikolov, T., Chen, K., Corrado, G. & Dean, J. Distributed Representations of Words and Phrases and their Compositionality. *ArXiv* (2013).
31. Mikolov, T., Chen, K., Corrado, G. & Dean, J. Efficient Estimation of Word Representations in Vector Space. *ArXiv* (2013).
32. Pennington, J., Socher, R. & Manning, C. Glove: Global Vectors for Word Representation. in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* 1532–1543 (Association for Computational Linguistics, 2014). doi:10.3115/v1/D14-1162.
33. Levy, O. & Goldberg, Y. Neural Word Embedding as Implicit Matrix Factorization. in *Advances in Neural Information Processing Systems 7* 2177–2185 (2014).
34. Diehl, G. W., Hon, O. J., Leutgeb, S. & Leutgeb, J. K. Grid and Nongrid Cells in Medial Entorhinal Cortex Represent Spatial Location and Environmental Features with Complementary Coding Schemes. *Neuron* **94**, 83-92.e6 (2017).

35. Dordek, Y., Soudry, D., Meir, R. & Derdikman, D. Extracting grid cell characteristics from place cell inputs using non-negative principal component analysis. (2016) doi:10.7554/eLife.10094.001.
36. 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).
37. Banino, A. *et al.* Vector-based navigation using grid-like representations in artificial agents. *Nature* **557**, 429–433 (2018).
38. Cueva, C. J. & Wei, X.-X. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. in *International Conference on Learning Representations* (2018).
39. Gao, R., Xie, J., Zhu, S.-C. & Wu, Y. N. Learning Grid Cells as Vector Representation of Self-Position Coupled with Matrix Representation of Self-Motion. in *International Conference on Learning Representations* (2019).
40. Nayebi, A. *et al.* Explaining heterogeneity in medial entorhinal cortex with task-driven neural networks. in *Advances in Neural Information Processing Systems* 12167–12179 (2021).
41. Waydo, S. & Koch, C. Unsupervised Learning of Individuals and Categories from Images. *Neural Comput* **20**, 1165–1178 (2008).
42. Calvo Tapia, C., Tyukin, I. & Makarov, V. A. Universal principles justify the existence of concept cells. *Sci Rep* **10**, (2020).
43. Ralph, M. A. L., Jefferies, E., Patterson, K. & Rogers, T. T. The neural and computational bases of semantic cognition. *Nature Reviews Neuroscience* vol. 18 42–55 Preprint at https://doi.org/10.1038/nrn.2016.150 (2016).
44. Rogers, T. T. & McClelland, J. L. Précis of Semantic Cognition: A Parallel Distributed Processing Approach. *Behavioral and Brain Sciences* **31**, 689–714 (2008).
45. Saxe, A. M., McClelland, J. L. & Ganguli, S. A mathematical theory of semantic development in deep neural networks. *Proc Natl Acad Sci U S A* **166**, 11537–11546 (2019).
46. Mok, R. M. & Love, B. C. A multilevel account of hippocampal function in spatial and concept learning: Bridging models of behavior and neural assemblies. *Sci Adv* **9**, (2023).
47. Mok, R. M. & Love, B. C. A non-spatial account of place and grid cells based on clustering models of concept learning. *Nat Commun* **10**, (2019).
48. Whittington, J. C. R., Warren, J. & Behrens, T. E. J. Relating transformers to models and neural representations of the hippocampal formation. in *International Conference on Learning Representations* (2022).
49. 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).
50. Huang, X. *et al.* Distinct spatial maps and multiple object codes in the lateral entorhinal cortex. *Neuron* (2023) doi:10.1016/j.neuron.2023.06.020.
51. Lee, D. D. & Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature* **401**, 788–791 (1999).
52. Sorscher, B., Mel, G. C., Ganguli, S. & Ocko, S. A. A unified theory for the origin of grid cells through the lens of pattern formation. in *Advances in Neural Information Processing Systems* 10003–10013 (2019).

53. Whittington, J. C. R., Dorrell, W., Ganguli, S., Timothy, S. & Behrens, E. J. Disentanglement with Biological Constraints: A Theory of Functional Cell Types. in *The Eleventh International Conference on Learning Representations* (2022).
54. Murphy, B., Talukdar, P. P. & Mitchell, T. Learning Effective and Interpretable Semantic Models using Non-Negative Sparse Embedding. in *Proceedings of COLING 2012* 1933–1950 (2012).
55. Sargolini, F. *et al.* Conjunctive Representation of Position, Direction, and Velocity in Entorhinal Cortex. *Science (1979)* **312**, 758–762 (2006).
56. Stensola, H. *et al.* The entorhinal grid map is discretized. *Nature* **492**, 72–78 (2012).
57. Oh, J., Guo, X., Lee, H., Lewis, R. & Singh, S. Action-Conditional Video Prediction using Deep Networks in Atari Games. in *Advances in Neural Information Processing Systems* 2863–2871 (2015).
58. Princeton University. About Wordnet. *https://wordnet.princeton.edu/* (2010).
59. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *ArXiv* (2018).
60. Wolf, T. *et al.* Transformers: State-of-the-Art Natural Language Processing. in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* 38–45 (2020). doi:10.18653/v1/2020.emnlp-demos.6.
61. Agirre, E. *et al.* A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches. in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics* 19–27 (2009).
62. Higgins, I. *et al.* beta-VAE: LEARNING BASIC VISUAL CONCEPTS WITH A CONSTRAINED VARIATIONAL FRAMEWORK. in *International Conference on Learning Representations* (2017).
63. Higgins, I. *et al.* Unsupervised deep learning identifies semantic disentanglement in single inferotemporal face patch neurons. *Nat Commun* **12**, (2021).
64. Mitchell, T. M. *et al.* Predicting Human Brain Activity Associated with the Meanings of Nouns. *Science (1979)* **320**, 1191–1195 (2008).
65. Nishida, S. & Nishimoto, S. Decoding naturalistic experiences from human brain activity via distributed representations of words. *Neuroimage* **180**, 232–242 (2018).
66. Duff, M. C. & Brown-Schmidt, S. The hippocampus and the flexible use and processing of language. *Front Hum Neurosci* (2012) doi:10.3389/fnhum.2012.00069.
67. Piai, V. *et al.* Direct brain recordings reveal hippocampal rhythm underpinnings of language processing. *Proc Natl Acad Sci U S A* **113**, 11366–11371 (2016).
68. Solomon, E. A., Lega, B. C., Sperling, M. R. & Kahana, M. J. Hippocampal theta codes for distances in semantic and temporal spaces. *Proc Natl Acad Sci U S A* **116**, 24343–24352 (2019).
69. Vaswani, A. *et al.* Attention Is All You Need. in *Advances in Neural Information Processing Systems* 5998–6008 (2017).
70. Brown, T. B. *et al.* Language Models are Few-Shot Learners. *ArXiv* (2020).
71. Goldstein, A. *et al.* Shared computational principles for language processing in humans and deep language models. *Nat Neurosci* **25**, 369–380 (2022).
72. Schrimpf, M. *et al.* The neural architecture of language: Integrative modeling converges on predictive processing. *Proceedings of the National Academy of Sciences* **118**, (2021).

73. Uria, B. *et al.* A model of egocentric to allocentric understanding in mammalian brains. *biorxiv* (2020) doi:10.1101/2020.11.11.378141.
74. George, D. *et al.* Clone-structured graph representations enable flexible learning and vicarious evaluation of cognitive maps. *Nat Commun* **12**, (2021).
75. Oja, E. & Plumbley, M. Blind Separation of Positive Sources by Globally Convergent Gradient Search. *Neural Comput* **16**, 1811–1825 (2004).
76. Hyvärinen, A. & Oja, E. Independent component analysis: algorithms and applications. *Neural Networks* **13**, 411–430 (2000).
77. Nesterov, Y. E. A method of solving a convex programming problem with convergence rate O(1/k^2). *Dokl Akad Nauk* **269**, 543–547 (1983).