



OPEN

## Neural network based successor representations to form cognitive maps of space and language

Paul Stoewer<sup>1,2</sup>, Christian Schlieker<sup>1,2</sup>, Achim Schilling<sup>1,3</sup>, Claus Metzner<sup>3,4</sup>, Andreas Maier<sup>2</sup> & Patrick Krauss<sup>1,2,3,5</sup>✉

How does the mind organize thoughts? The hippocampal-entorhinal complex is thought to support domain-general representation and processing of structural knowledge of arbitrary state, feature and concept spaces. In particular, it enables the formation of cognitive maps, and navigation on these maps, thereby broadly contributing to cognition. It has been proposed that the concept of multi-scale successor representations provides an explanation of the underlying computations performed by place and grid cells. Here, we present a neural network based approach to learn such representations, and its application to different scenarios: a spatial exploration task based on supervised learning, a spatial navigation task based on reinforcement learning, and a non-spatial task where linguistic constructions have to be inferred by observing sample sentences. In all scenarios, the neural network correctly learns and approximates the underlying structure by building successor representations. Furthermore, the resulting neural firing patterns are strikingly similar to experimentally observed place and grid cell firing patterns. We conclude that cognitive maps and neural network-based successor representations of structured knowledge provide a promising way to overcome some of the shortcomings of deep learning towards artificial general intelligence.

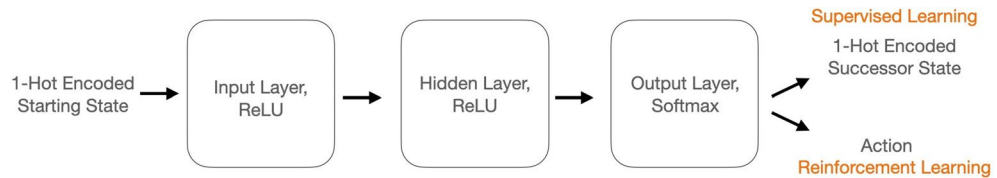
Cognitive maps are mental representations that serve an organism to acquire, code, store, recall, and decode information about the relative locations and features of objects<sup>1</sup>. Electrophysiological research in rodents suggests that the hippocampus<sup>2</sup> and the entorhinal cortex<sup>3</sup> are the neurological basis of cognitive maps. There, highly specialised neurons including place<sup>4</sup> and grid cells<sup>5</sup> support map-like spatial codes, and thus enable spatial representation and navigation<sup>6</sup>, and furthermore the construction of multi-scale maps<sup>7,8</sup>. Also human fMRI studies during virtual navigation tasks have shown that the hippocampal and entorhinal spatial codes, together with areas in the frontal lobe, enable route planning during navigation<sup>9</sup>, e.g. detours<sup>10</sup>, shortcuts or efficient novel routes<sup>11</sup>, and in particular hierarchical spatial planning<sup>12</sup> based on distance preserving representations<sup>13</sup>.

Recent human fMRI studies even suggest that these map-like representations might not be restricted to physical space, i.e. places and spatial relations, but also extend to more abstract relations like in social and conceptual spaces<sup>14–16</sup>, thereby contributing broadly to other cognitive domains<sup>17</sup>, and thus enabling navigation and route planning in arbitrary abstract cognitive spaces<sup>18</sup>.

The hippocampus also plays a crucial role in episodic and declarative memory<sup>19,20</sup>. Furthermore, the hippocampal formation, as a hub in brain connectivity<sup>21</sup>, receives highly processed information via direct and indirect pathways from a large number of multi-modal areas of the cerebral cortex including language related areas in the frontal, temporal, and parietal lobe<sup>22</sup>. Finally, some findings indicate that the hippocampus even contributes to the coding of narrative context<sup>23,24</sup>, and that memory representations, similar to the internal representation of space, systematically vary in scale along the hippocampal long axis<sup>25</sup>. This scale might be used for goal directed navigation with different horizons<sup>26</sup> or even encode information from smaller episodes to more complex concepts<sup>27</sup>. This geometry can also be modeled in artificial neural networks when performing an abstraction task<sup>28</sup>. Cognitive maps therefore enable flexible planning through re-mapping of place cells and through the continuous (re-)scaling, generalization or detailed representation of information<sup>29</sup>.

A number of computational models try to describe the hippocampal-entorhinal complex. For instance, the Tolman–Eichenbaum Machine describes hippocampal and entorhinal cell types and allows flexible transfer of structural knowledge<sup>30</sup>. Another framework that aims to describe the firing patterns of place cells in the

<sup>1</sup>Cognitive Computational Neuroscience Group, University Erlangen-Nuremberg, Erlangen, Germany. <sup>2</sup>Pattern Recognition Lab, University Erlangen-Nuremberg, Erlangen, Germany. <sup>3</sup>Neuroscience Lab, University Hospital Erlangen, Erlangen, Germany. <sup>4</sup>Biophysics Lab, University Erlangen-Nuremberg, Erlangen, Germany. <sup>5</sup>Linguistics Lab, University Erlangen-Nuremberg, Erlangen, Germany. ✉email: patrick.krauss@fau.de



**Figure 1.** A scheme of the used Neural Network for the supervised and reinforcement learning task. The network receives the one hot encoded starting state of the environment as input. It uses one hidden layer with a ReLU activation function and the output is a softmax layer. In the case of the spatial exploration and the linguistic task, the output of the network is the one-hot encoded successor state of the corresponding input. In case of the spatial navigation task, which uses reinforcement learning, the output of the network is the action depending on the input state.

hippocampus uses the successor representation (SR) as a building block for the construction of cognitive or predictive maps<sup>31,32</sup>. The hierarchical structure in the entorhinal cortex can also be modeled by means of multi-scale successor representations<sup>33</sup>. Here, SR can be for example learned with a feature set of boundary vector cells<sup>34</sup> or with a sequence generation model inspired by the entorhinal–hippocampal circuit<sup>35</sup>.

To further investigate both the biological plausibility and potential machine learning applications of multi-scale SR and cognitive maps, we developed a neural network based simulation of place cell behavior under different circumstances. In particular, we trained a neural network to learn the SR for a simulated spatial environment and a navigation task in a virtual maze as proposed by Alvernhe et al.<sup>36</sup>. In addition, we investigated if the applicability of our model extends from space to language as the hippocampus is known to also contribute to language processing<sup>37,38</sup>. Therefore, we created a model to learn a simplified artificial language. In particular, the model's task was to learn the underlying grammatical structure in terms of SR of words by observing exemplary input sentences only.

## Methods

**Successor representation.** The developed model is based on the principle of the successor representation (SR). As proposed by Stachenfeld et al. the SR can model the firing patterns of the place cells in the hippocampus<sup>32</sup>. The SR was originally designed to build a representation of all possible future rewards  $V(s)$  that may be achieved from each state  $s$  within the state space over time<sup>39</sup>. The future reward matrix  $V(s)$  can be calculated for every state in the environment whereas the parameter  $t$  indicates the number of time steps in the future that are taken into account, and  $R(s_t)$  is the reward for state  $s$  at time  $t$ . The discount factor  $\gamma^{0,1}$  reduces the relevance of states  $s_t$  that are further in the future relative to the respective initial state  $s_0$  (cf. Eq. 1).

$$V(s) = E \left[ \sum_{t=0}^{\infty} \gamma^t R(s_t) | s_0 = s \right] \quad (1)$$

Here,  $E$  denotes the expectation value.

The future reward matrix  $V(s)$  can be re-factorized using the SR matrix  $M$ , which can be computed from the state transition probability matrix  $T$  of successive states (cf. 2). In case of supervised learning, the environments used for our model operate without specific rewards for each state. For the calculation of these SR we choose  $R(s_t) = 1$  for every state.

$$V(s) = \sum_{s'} M(s, s') R(s') \quad M = \sum_{t=0}^{\infty} \gamma^t T^t \quad (2)$$

**Spatial exploration task.** *Spatial environment.* The spatial environment created in our framework is designed as a discrete grid-like room which can be freely explored by the agent. The neighboring states of a particular state are defined as direct successor states. Walls and barriers are not counted as possible successor state for a neighboring initial state. The squared room consists of 100 states arranged as a  $10 \times 10$  rectangular grid (cf. Fig. 3).

*Neural network architecture.* To be able to learn the SR by just observing the environment, we set up three-layered neural networks that learn the transition probabilities of the different environments (Fig. 1). The input to the network is the momentary state encoded as one-hot vectors. Thus the number of neurons in the input layer is 100 for the exploration task. The hidden layer neurons have a ReLU activation function, whereas the number of neurons is equal to the input state. The softmax output layer gives a probability distribution for all successor states. Therefore it has also a size of 100. Note that, even though the number of neurons in the hidden layer equals the number of neurons of both, the input and output layer, it can still be considered as an information bottleneck since the  $100 \times 100$  state transition matrix containing 10,000 entries is compressed to a 100-dimensional representation.

The architecture of the network was evaluated and the performance for different parameters was compared. We tested different sizes of the hidden layers (cf. Fig. 2) regarding the performance of the neural network. However the performance was not greatly affected by different architectures (Fig. 2).

**Training parameters.** The training and test data set is created by sampling trajectories through the spatial structure of the simulated environment. First, a random starting state is chosen as input and subsequently another random possible successor state is chosen from its possible neighbors as desired output. Walls are excluded as input states. For the experiment we sampled 10,000 state-successor state pairs, and trained for 5000 epochs. We used cross entropy as a loss function and the Adam optimizer with a learning rate of 0.01.

**Spatial navigation task.** *Spatial environment.* The environment for the spatial navigation task is again a discrete grid-like room, where neighboring states of a particular state are defined as direct successor states. Walls and barriers are not counted as possible successor state for a neighboring initial state. The maze was represented as a  $15 \times 15$  rectangular grid, whereas only 94 states from all 225 states were “allowed” states that could be observed by the agent (cf. Fig. 5).

*Reinforcement learning architecture.* In an attempt to reproduce experimental data, we simulated a maze which is described above, as proposed by Alvernhe et al.<sup>36</sup>. Furthermore, a reward system is required for reinforcement learning (RL). Our RL approach enables us to define rewards in the spatial environments, which we use to simulate the food trays of the original experiment. The network structure is again a three-layered network, with a ReLU activation function for the hidden layer neurons and a softmax output layer, which yields the probabilities for the next actions. A DQN agent can choose from several actions depending on the number of the neighboring states belonging to the current state. If the agent chooses a wall state during training, the momentary training run is terminated and a new random starting state is chosen randomly.

The architecture of the neural network for the RL approach was also evaluated. In particular, the number of neurons in the hidden layer was tested. The architecture did not significantly influence the average received reward during training (cf. Fig. 2).

*Training parameters.* Training was performed for 10,000 epochs, with warm up steps of 300, an ADAM optimizer with a learning rate of 0.001, maximum number of steps per episode of 30 and the DQN agent used the greedy policy.

**Linguistic structure inference task.** *Language environment.* Additionally, we set up a state space with a non-spatial structure, i.e. a linguistic environment. The environment consists of 40 discrete states representing the vocabulary. Each state corresponds to a particular word, and each word belongs to one of the five different word classes: adjectives, verbs, nouns, pronouns and question words. The transition probabilities between subsequent words are defined according to a simplified syntax which consists of three types of linguistic constructions: an adjective-noun construction (cf. rule 3a), a descriptive construction (cf. rule 3b) and an interrogative construction (cf. rule 3c).

$$\text{adjective} \rightarrow \text{noun} \quad (3a)$$

$$\text{pronoun} \rightarrow \text{verb} \rightarrow \text{adjective} \quad (3b)$$

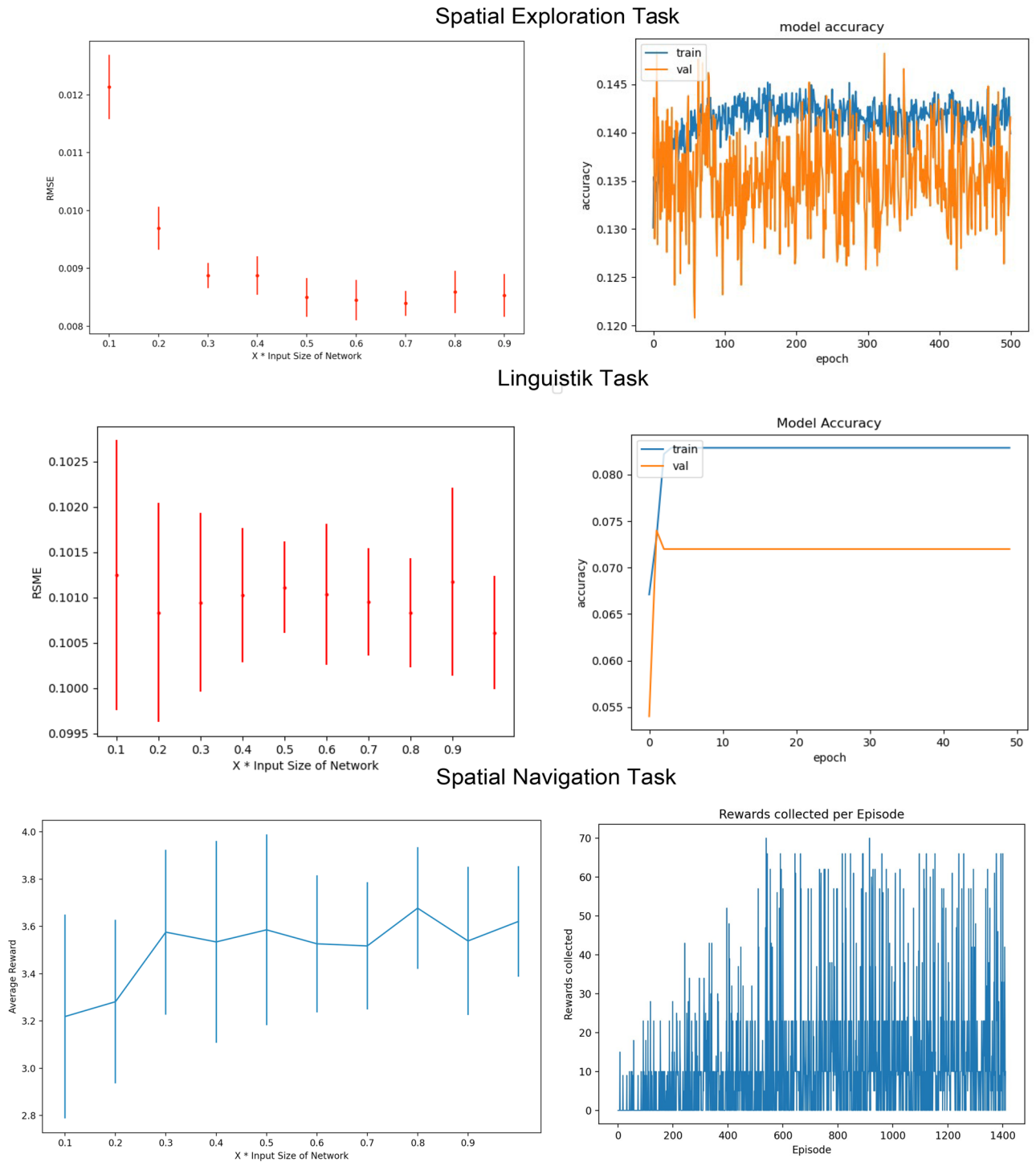
$$\text{question} \rightarrow \text{pronoun} \rightarrow \text{verb} \quad (3c)$$

The syntax rules, i.e. constructions, determine the transition probabilities for randomly chosen starting states and a word from the picked word group is set as label for the training data. The individual words from the successor word class are chosen with equal probability. The constructed sentences have therefore no particular meaning.

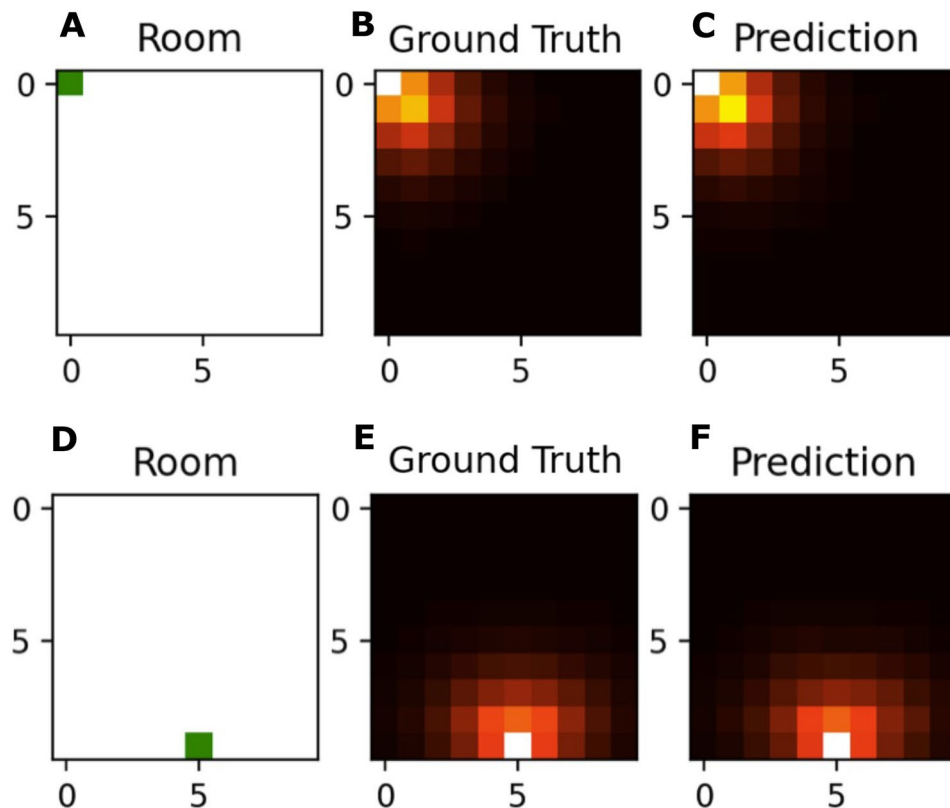
*Neural network architecture.* For this task the same three layer architecture as in the spatial exploration task was used. Different sizes of the hidden layer did not influence the performance (cf. Fig. 2). The network consisted of an input layer with the size of 40, a hidden layer with the same size using a ReLU activation function and a softmax output layer of size 40. The input and output are encoded as one-hot encoded vectors representing a word of the vocabulary.

*Training parameters.* For the language data set, 5,000 training and test samples were generated and the network was trained for 50 epochs. Cross entropy is again defined as loss function and the Adam optimizer with a learning rate of 0.01 is used.

**Transition probability and successor representation matrix.** After the training process, the network can predict all probabilities of successor states for any given initial state. Concatenating the predictions of all states leads to the transition probability (TP) matrix of our environments, which we use to calculate the SR matrix (cf. Equation 2). In case of the supervised learning approach (spatial exploration task and language task), the output of the network is a vector shaped like a row of the respective environment’s TP or SR matrix and can therefore directly be used to fill the TP or SR matrix, respectively. The reinforcement learning network however



**Figure 2.** Evaluation of the models with regard to architecture and performance during training. Top: Evaluations for the spatial exploration model. The network’s architecture is tested depending on the size of the hidden layer. Different sizes depending on the input size in comparison to the RMSE of the predicted transition probability matrix and the ground truth matrix are shown. The error decreases with the size of the hidden layer until 30% of the input size. After that the error saturates. The accuracy of the model during training and validation increases shortly at the beginning from 0.13 to 0.14, afterwards it stagnates. The low accuracy of the model is connected to the potential 9 different successor states of each starting state which are randomly sampled as label. However the RMSE is low in all cases. Middle: In the linguistic task the hidden layer size does not play an important role. The RMSE stays similar for all configurations. The accuracy for the model also jumps at the beginning to around 0.08. The low accuracy can be again explained by up to possible ten randomly sampled successor states. However the RSME regarding the ground truth is low again. Bottom: The architecture for the spatial navigation task also does not influence the performance much. The average collected reward increases until 30% of the input size and subsequently saturates. During training, the model improves the received reward continuously until around 600 episodes.



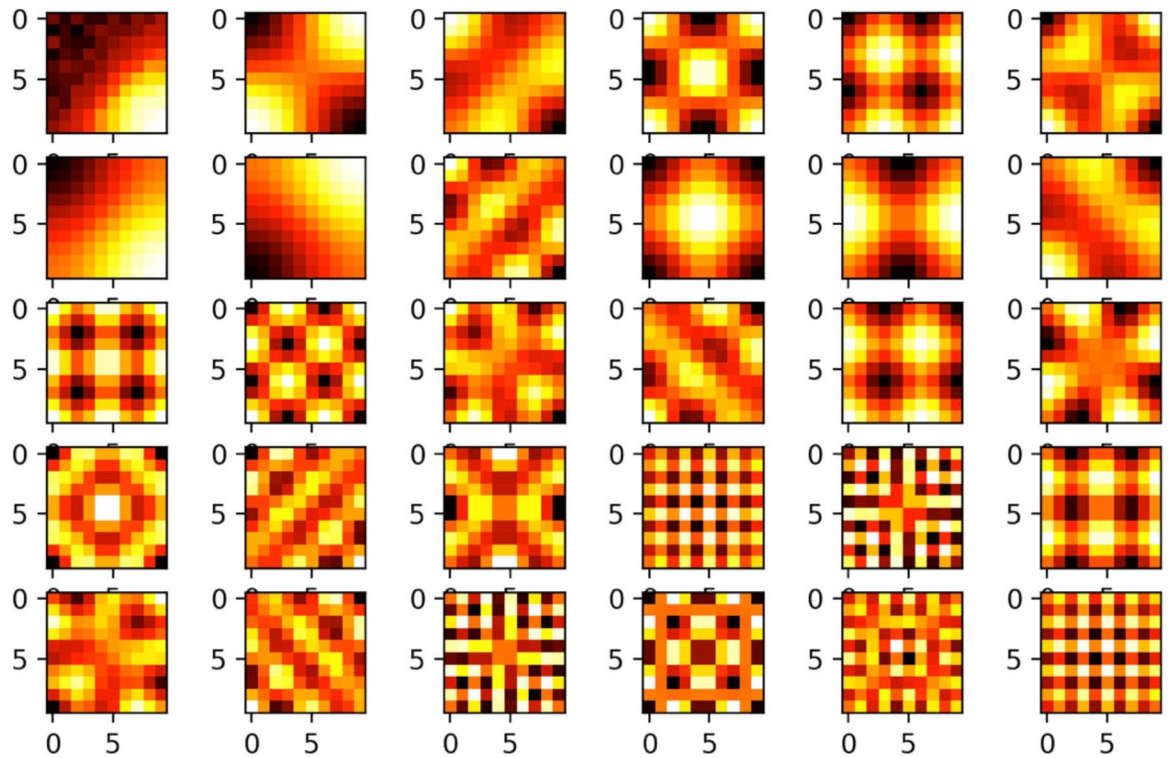
**Figure 3.** Supervised learning to explore a spatial environment: The SR for a 2D squared environment is learned with a supervised neural network. The small green squares indicate two sample starting positions (A, D). The corresponding SR are calculated and serve as ground truth (B, E). The neural network learns the transition probabilities for its direct neighbors and estimates the SR for a sequence length of  $t = 10$  (C, F).

only yields the probabilities for the direct successors of a given state, which therefore need to be further extended to a vector containing all possible states of the environment.

**Reproducing experimental grid cell firing patterns.** After training the network, the resulting SR matrices are evaluated. Therefore, each state encoded as one-hot vector is fed in as input to the network, and the resulting softmax output vectors are concatenated to build the SR matrix. The resulting SR matrix can be used to calculate their Eigendecomposition. The different Eigenvectors can be ordered according to their size, and are subsequently reshaped to fit the shape of the corresponding state space, i.e. simulated environment. The reshaped Eigenvectors are supposed to form grid like patterns, and to be a representation of the grid cells' receptive fields<sup>32</sup>.

**Multi-dimensional scaling.** A frequently used method to generate low-dimensional embeddings of high-dimensional data is t-distributed stochastic neighbor embedding (t-SNE)<sup>40</sup>. However, in t-SNE the resulting low-dimensional projections can be highly dependent on the detailed parameter settings<sup>41</sup>, sensitive to noise, and may not preserve, but rather often scramble the global structure in data<sup>42,43</sup>. In contrast, multi-Dimensional-Scaling (MDS)<sup>44–47</sup> is an efficient embedding technique to visualize high-dimensional point clouds by projecting them onto a 2-dimensional plane. Furthermore, MDS has the decisive advantage that it is parameter-free and all mutual distances of the points are preserved, thereby conserving both the global and local structure of the underlying data.

When interpreting patterns as points in high-dimensional space and dissimilarities between patterns as distances between corresponding points, MDS is an elegant method to visualize high-dimensional data. By color-coding each projected data point of a data set according to its label, the representation of the data can be visualized as a set of point clusters. For instance, MDS has already been applied to visualize for instance word class distributions of different linguistic corpora<sup>48</sup>, hid-den layer representations (embeddings) of artificial neural networks<sup>49,50</sup>, structure and dynamics of recurrent neural networks<sup>51–53</sup>, or brain activity patterns assessed during e.g. pure tone or speech perception<sup>48,54</sup>, or even during sleep<sup>55,56</sup>. In all these cases the apparent compactness and mutual overlap of the point clusters permits a qualitative assessment of how well the different classes separate.



**Figure 4.** Grid cell-like Eigenvectors of the SR matrix: The firing patterns of grid cells in the entorhinal cortex are proposed to represent the Eigenvectors of the SR matrix<sup>32</sup>. For the squared room depicted in Fig. 3, the Eigenvectors for the first 30 Eigenvalues of the SR matrix are shown (re-shaped to the shape of the squared environment). Indeed, they resemble grid cell-like firing patterns. Furthermore, the grids vary in orientation and scaling, as observed in electrophysiological experiments in rodents.

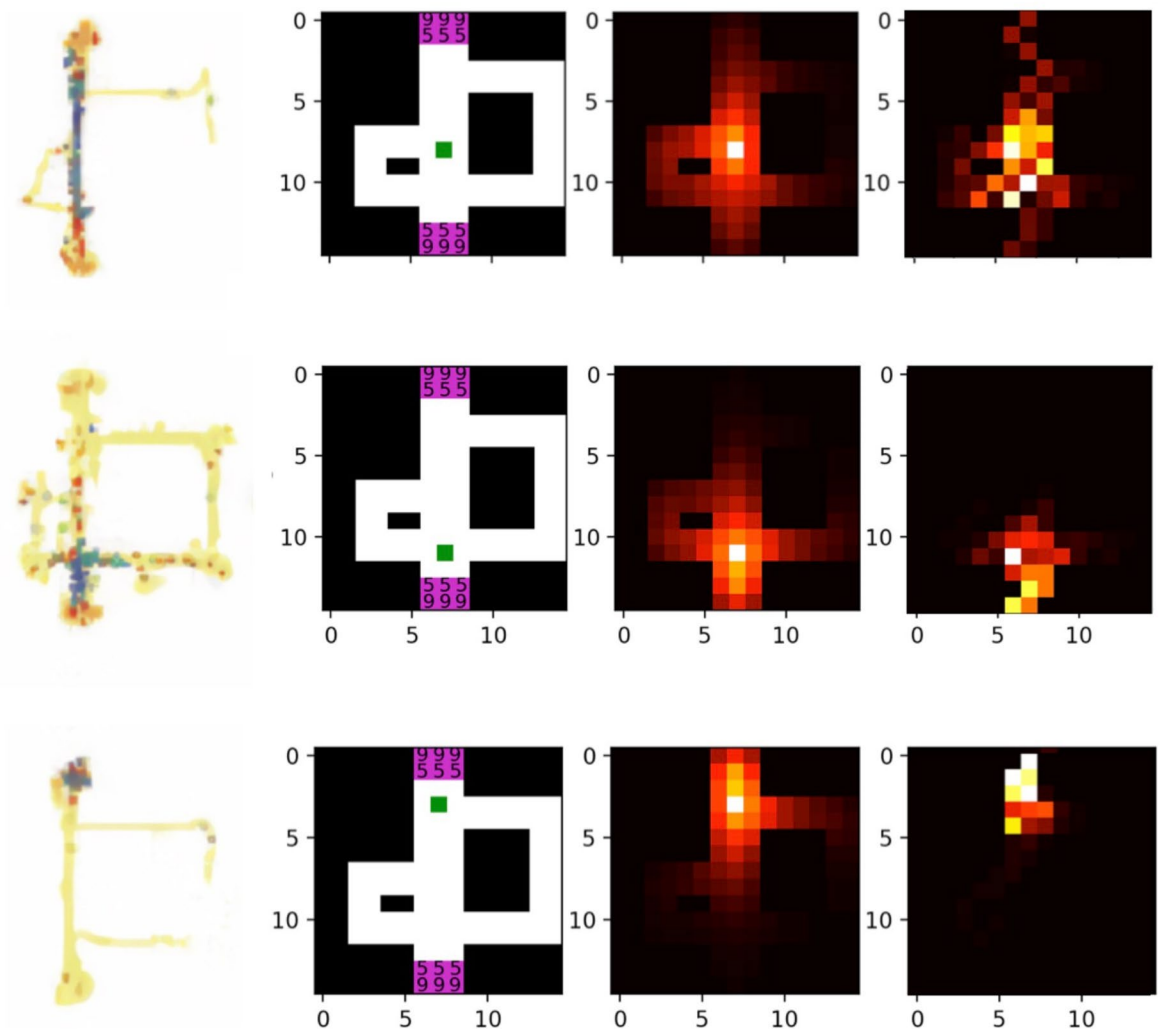
**Code implementation.** The models were coded in Python. The neural networks were designed using the Keras<sup>57</sup> and Keras-RL<sup>58</sup> libraries. Mathematical operations were performed with numpy<sup>59</sup> and scikit-learn<sup>60</sup> libraries. Visualizations were realised with matplotlib<sup>61</sup> and networkX<sup>62</sup>.

## Results

**Spatial environment.** *Supervised learning reproduces basic firing patterns of place cells in rodents.* In the supervised learning approach, the transition probabilities between neighboring places and hence the SR is learned by randomly observing places (states) and exploring their potential successors. The accuracy during training reaches around 0.14, which is the theoretical maximum, since there are nine potential successor states for each input state (cf. Fig. 2). Note that, the chance level for the accuracy would be 0.01 as there are 100 output neurons. In the simplest case of a 2D square environment without any obstacles, the transition probabilities from any starting place to all its eight neighbors are identical, i.e. uniformly distributed. Places at the walls (corners) of the room however only have five (three) neighboring states, so the transition probabilities corresponding to the remaining neighboring states representing those walls are zero, respectively. The resulting successor representations learned by the neural network are almost identical to the ground truth (cf. Fig. 3). Furthermore, these firing patterns are strikingly similar to those of place cells in rodents. They reflect the environment's spatial structure depending on obstacles and room shape<sup>63</sup>, i.e. the intensity of the firing patterns is centered around the starting position (as this is also the most probable next state) and directed away from any walls into the open space (cf. Figs. 1, 3), as described e.g. in<sup>64</sup>, and found experimentally<sup>65</sup>.

**Eigenvectors of learned SR resemble firing patterns of grid cells in rodents.** Stachenfeld et al.<sup>64</sup> propose that the grid-like firing patterns of grid cells in the entorhinal cortex of rodents<sup>66</sup> may be explained by an Eigendecomposition of the SR matrix, whereas each individual grid cell would correspond to one Eigenvector. To test this assumption in the context of our framework, we calculated the Eigenvectors of the learned SR matrix as shown in Fig. 3, and reshaped them to the shape of the environment. We find that this procedure actually leads to grid cell-like firing patterns (cf. Fig. 4). The first 30 Eigenvectors ordered by increasing value of the corresponding Eigenvalues are shown in Fig. 4. As known from neurobiology<sup>66</sup>, the grid-like patterns vary in orientation and mesh size (i.e. frequency). In particular, the smaller the corresponding Eigenvalue of the Eigenvector, the smaller the mesh size, i.e. more fine-grained the resulting grid, becomes (cf. Fig. 4). Furthermore it is known that, the individual orientation of the grid cells' firing patterns follows no particular order, whereas the mesh size of the grids varies systematically along the long axis of the entorhinal cortex<sup>67</sup>. This feature, especially, is thought to enable multi-scale mapping, route planning and navigation<sup>33</sup>.

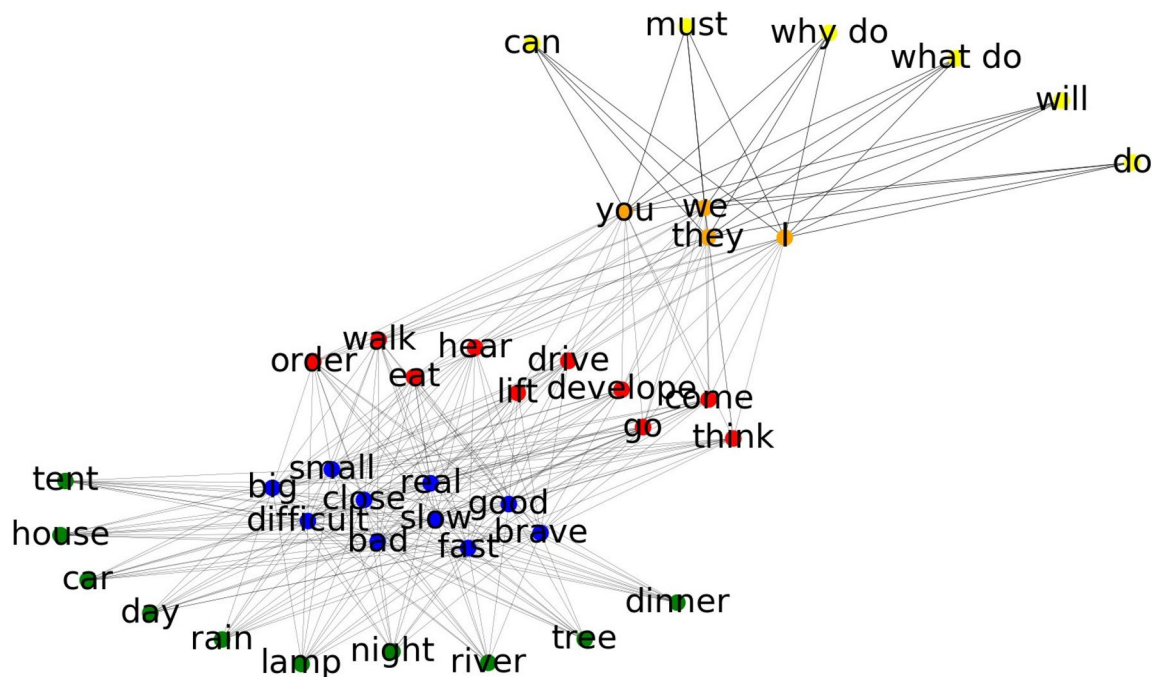
## Experiment Room Ground Truth Prediction



**Figure 5.** Reinforcement learning to navigate a spatial environment: We reproduced the rat maze experiment published by Alvernhe et al.<sup>36</sup>. (left column). Therefore, we simulated the corresponding maze environment, small green squares indicate three different sample starting positions (second column). Based on the transition probabilities to neighboring states we calculated the successor representation of the maze as ground truth (third column). The predicted SR of the trained network, i.e. the firing patterns of the artificial place cells are very similar to the underlying ground truth (right column).

**Reinforcement learning reproduces basic firing patterns of place cells in rodents.** In contrast to a goal-free random walk in order to explore a novel environment (as in the previous setting), navigation is usually driven by a specific goal or reward<sup>68</sup> like, e.g. food. The task is therefore ideally suited for reinforcement learning (RL)<sup>69</sup>. In our simulation, we reproduced a classical rodent maze experiment presented by Alvernhe et al.<sup>36</sup>. As in the supervised learning setting, the successor representations learned in the RL setting are very similar to the ground truth (cf. Fig. 5). Also, the resulting place cell firing patterns closely resemble those of place cells in rodents during maze navigation tasks. The average reward increases systematically during the training process (cf. Fig. 2). The SR place fields are clearly different from those obtained without any reward in the goal-free exploration task (cf. Fig. 3). A position close to a reward state is associated with highly localized firing patterns, whereas the highest successor probabilities are directed towards the reward states. Furthermore, places in the middle of the maze are associated with firing patterns that are stretched parallel to the orientation of the maze's main corridor. Highest successor probabilities are localized around the starting position, but still also in reach of the reward states. The side arms of the maze are a detour to the goal (reward position), and are therefore mainly ignored by the network, i.e. associated with the lowest successor state probabilities.

**Linguistic structures.** *Linguistic constructions define a network-like linguistic map.* Cognitive maps are however not restricted to physical space. On the contrary, cognitive maps may also be applied to arbitrary ab-



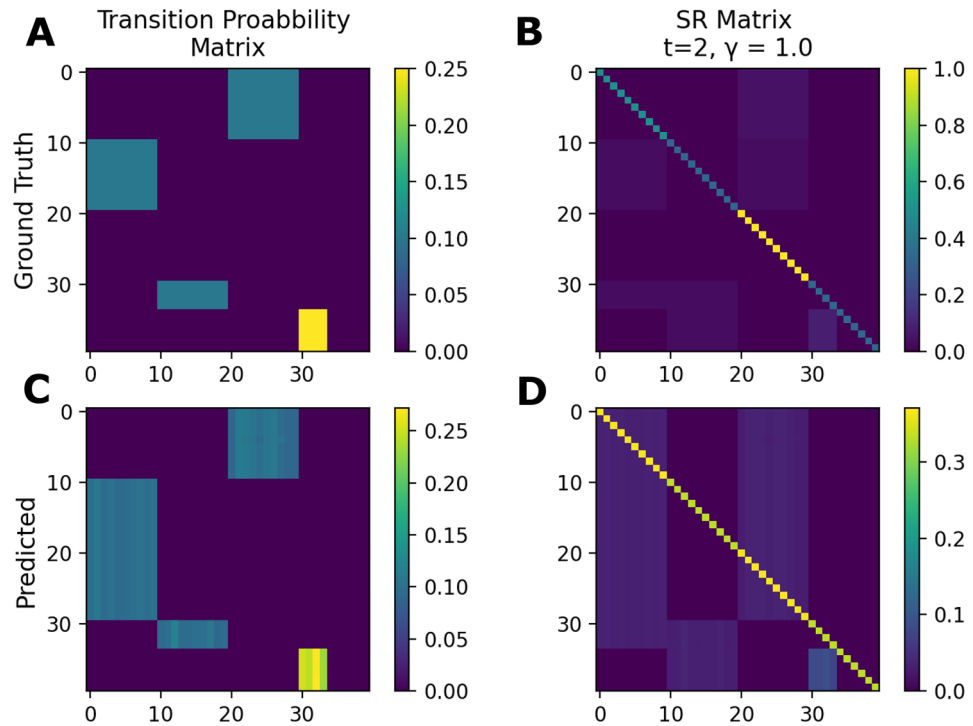
**Figure 6.** Network-like map of linguistic constructions: The simplified language model consists of five different word classes and three linguistic constructions defining allowed word class transitions. The word transition matrix can be visualized as a graph or network-like map, whereas each word corresponds to a node, and edges represent possible word transitions. Different node colors indicate different word classes. Note that, edges for transition probabilities smaller than  $10^{-4}$  are not shown for better readability.

stract and complex state spaces. In general, any state space can be represented as a graph. In this case, nodes correspond to states, and edges to state transitions. A prime example of such graph-like (or network-like) state space representations is language. In cognitive linguistics, there is an overall agreement on the fact, that language is represented as a network in the human mind<sup>70–76</sup>, whereas the nodes correspond to linguistic units at different hierarchical levels from phonemes, through words, to idioms and abstract argument structure constructions<sup>70</sup>. In particular, “the nodes at one level of analysis are networks at another level of analysis”<sup>77</sup>. Hence, multi-scale SR<sup>33</sup> appears to be an ideal theoretical framework to explain language representation and processing in the human mind, whereas the systematically varying grid-scale along the long axis of the entorhinal cortex<sup>67</sup> might explain its implementation in the human brain. To investigate this hypothesis, we constructed as a first step a simplified language as described in detail in the “Methods” section. The lexicon together with the three linguistic constructions result in a network-like linguistic map (cf. Fig. 6), that has to be learned by the neural network.

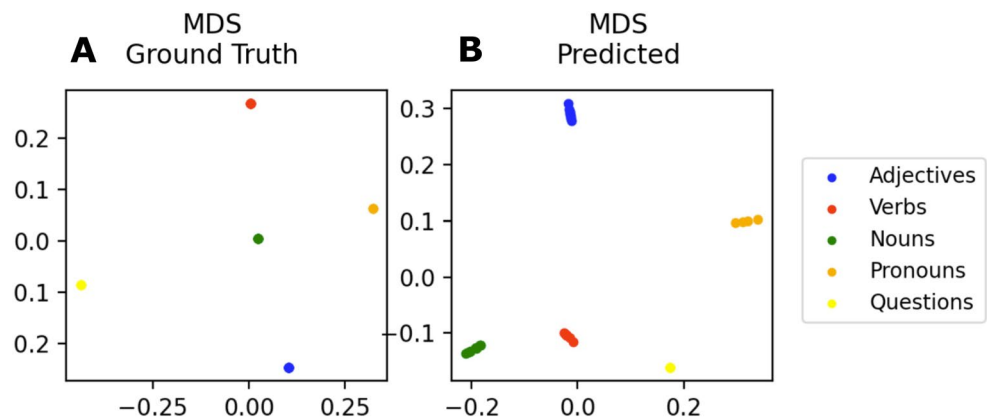
*The neural network learns state TP and SR matrices.* The learned behaviour of the network in the state space can be displayed as a state TP or SR matrix. The accuracy of the prediction of the network reaches around 0.085, which is close to the theoretical maximum of 0.1 given the probability of each state’s potential 10 successor states. (cf. Fig. 2). Here, the chance level of the accuracy would be 0.025. Therefore, after training, the TP matrix which is predicted by the network is very similar to the ground truth (cf. Fig. 7A,C). However the network also predicts adjectives (states 0–10) as successors of nouns (states 20–30), even though this transition does not explicitly exist in the three pre-defined constructions. Consequently, the network’s SR matrix is also slightly different from the ground truth (cf. Fig. 7B,D).

*Word classes spontaneously emerge as clusters in the TP and SR vector space.* The transition probabilities from a given word to all other 40 words (rows in the TP matrix), as well as the corresponding successor probabilities (rows in the SR matrix) can be represented as vectors, and hence may be interpreted as points in a 40-dimensional TP or SR space respectively, whereas each word corresponds to a particular point. In order to further investigate the properties of these high-dimensional representations, we visualize both the TP and the SR space using multi-dimensional scaling. In particular, the 40-dimensional TP and SR vector representations of each word are projected onto a two-dimensional plane as described in detail in the “Methods” section. By color-coding each word according to its word class, we observed putative clustering of the vocabulary. Remarkably, the words actually cluster according to their word classes (cf. Figs. 8,9), even though, this information was not provided (e.g. as an additional label for each word) to the neural network at any time during training.





**Figure 7.** Word transition probability and word successor representation matrices: After training on the linguistic data, the neural network predicts the word transition probabilities (TP) and the word successor representations (SR). As ground truth, the calculated TP matrix (A) and the SR matrix (B) for  $t=2$  and  $\gamma=1$  are shown. The corresponding predictions learned by the network are very similar to the ground truth in both cases (C, D). States (0–39) correspond to words. 0–9: adjectives, 10–19: verbs, 20–29: nouns, 30–34: pronouns, 35–40: question words.

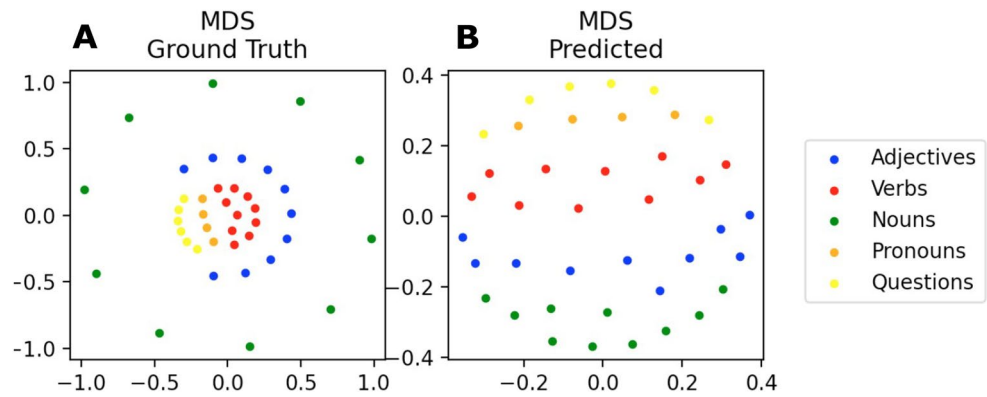


**Figure 8.** MDS of the word transition probability vectors: a two-dimensional projection of the 40-dimensional word TP vectors (rows in TP matrix) for the calculated ground truth (A) and the learned TP matrix (B). In both cases, the words build clearly separated, dense clusters according to the respective word class. Different colors correspond to word classes. Note that, scaling of the axes is in arbitrary units since coordinates have no particular meaning other than indicating the relative positions of the projected vectors.

## Discussion

In this study, we demonstrated that efficient successor representations can be learned by artificial neural networks in different scenarios. The emerging representations share important properties with network-like cognitive maps, enabling e.g. navigation in arbitrary abstract and conceptual spaces, and thereby broadly supporting domain-general cognition, as proposed by Bellmund et al.<sup>18</sup>.

In particular, we created a model, which can learn the SR for spatial and non-spatial environments. The model successfully reproduced experimentally observed firing patterns of place and grid cells in simulated spatial environments in two different scenarios. First, an exploration task based on supervised learning in a squared



**Figure 9.** MDS of the word successor representation vectors: a two-dimensional projection of the 40-dimensional word SR vectors (rows in SR matrix) for the calculated ground truth (A) and the learned SR matrix (B). In both cases, the words cluster according to the respective word class. Different colors correspond to word classes again. However, the resulting clusters are less dense and located closer to each other than for the TP vectors. Since SR vectors cover several time steps, whereas TP vectors only cover a single time step in the future, this result is intuitive. Note that, scaling of the axes is in arbitrary units since coordinates have no particular meaning other than indicating the relative positions of the projected vectors.

room without any obstacles, and second, a navigation task based on reinforcement learning in a simulated maze. Furthermore our neural network model inferred the underlying word classes of a simplified artificial language framework just by observing sequences of words, and without any prior knowledge about actual word classes.

The involvement of the entorhinal–hippocampal complex—as being the most probable candidate structure underlying network-like cognitive maps and multi-scale navigation<sup>15,16,29,33,64</sup>—in language processing has already been experimentally demonstrated<sup>37,38</sup>. Our study further supports, in particular, the involvement of place cells, as being the nodes of the “language network” as suggested in cognitive linguistics<sup>70–76</sup>. Early language acquisition, especially, is driven by passive listening<sup>78</sup> and implicit learning<sup>79</sup>. Our model replicates learning by listening and therefore resembles a realistic scenario.

The varying grid cell scaling along the long axis of the entorhinal cortex is known to be associated with hierarchical memory content<sup>25</sup>. The Eigenvectors of the SR matrix are strikingly similar to the firing patterns of grid cells, and therefore provide a putative explanation of the computational mechanisms underlying grid cell coding. These multi-scale representations are perfectly suited to map hierarchical linguistic structures from phonemes through words to sentences, and even beyond, like e.g. events or entire narratives. Indeed, recent neuroimaging studies provide evidence for the existence of “event nodes” in the human hippocampus<sup>23</sup>.

Since our neural network model is able to learn the underlying structure of a simplified language, we speculate that also the human hippocampal–entorhinal complex similarly encodes the complex linguistic structures of the languages learned by a given individual. Therefore learning further languages of similar structure as previously learned languages might be easier due to the fact that the multi-scale representations and cognitive maps in the hippocampus can be more easily transferred and re-mapped<sup>15,16</sup>, i.e. re-used, in major parts to the new language.

Whether the hippocampus is actually involved in multi-scale representation and processing of linguistic structures across several hierarchies needs to be verified experimentally and theoretically. Neuroimaging studies during natural language perception and production, like for instance listening to audiobooks<sup>48,80</sup>, need to be performed. Only continuous, connected speech and language provides such corpus-like rich linguistic structures<sup>48</sup>, being crucial to assess putative multi-scale processing. Additionally, further theoretical studies are needed to extend the presented model, and to apply it to more complex and naturalistic linguistic tasks, like e.g. word prediction in a natural language scenario.

As recently suggested, the neuroscience of spatial navigation might be of particular importance for artificial intelligence research<sup>81</sup>. A neural network implementation of hippocampal successor representations, especially, promises advances in both fields. Following the research agenda of Cognitive Computational Neuroscience proposed by Kriegeskorte et al.<sup>82</sup>, neuroscience and cognitive science benefit from such models by gaining deeper understanding of brain computations<sup>50,83,84</sup>. Conversely, for artificial intelligence and machine learning, neural network-based multi-scale successor representations to learn and process structural knowledge (as an example of neuroscience-inspired artificial intelligence<sup>85</sup>), might be a further step to overcome the limitations of contemporary deep learning<sup>86–89</sup> and towards human-level artificial general intelligence.

### Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 2 April 2022; Accepted: 15 June 2022

Published online: 04 July 2022

## References

1. Tolman, E. C. Cognitive maps in rats and men. *Psychol Rev.* **55**(4), 189 (1948).
2. O'Keefe, J. & Nadel, L. *The Hippocampus as a Cognitive Map* (Oxford University Press, Oxford, 1978).
3. Moser, E. I., Moser, M.-B. & McNoughton, B. L. Spatial representation in the hippocampal formation: A history. *Nat. Neurosci.* **20**(11), 1448–1464 (2017).
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. [https://doi.org/10.1016/0006-8993\(71\)90358-1](https://doi.org/10.1016/0006-8993(71)90358-1) (1971).
5. Hafting, T., Fyhn, M., Molden, S., Moser, M.-B. & Moser, E. I. Microstructure of a spatial map in the entorhinal cortex. *Nature* **436**(7052), 801–806 (2005).
6. Moser, E. I., Kropff, E. & Moser, M.-B. Place cells, grid cells, and the brain's spatial representation system. *Annu. Rev. Neurosci.* **31**, 69–89 (2008).
7. Geva-Sagiv, M., Las, L., Yovel, Y. & Ulanovsky, N. Spatial cognition in bats and rats: From sensory acquisition to multiscale maps and navigation. *Nat. Rev. Neurosci.* **16**(2), 94–108 (2015).
8. Kunz, L. *et al.* Mesoscopic neural representations in spatial navigation. *Trends Cogn. Sci.* **23**(7), 615–630 (2019).
9. Spiers, H. J. & Maguire, E. A. Thoughts, behaviour, and brain dynamics during navigation in the real world. *Neuroimage* **31**(4), 1826–1840 (2006).
10. Spiers, H. J. & Gilbert, S. J. Solving the detour problem in navigation: A model of prefrontal and hippocampal interactions. *Front. Hum. Neurosci.* **9**, 125 (2015).
11. Hartley, T., Maguire, E. A., Spiers, H. J. & Burgess, N. The well-worn route and the path less traveled: Distinct neural bases of route following and wayfinding in humans. *Neuron* **37**(5), 877–888 (2003).
12. Balaguer, J., Spiers, H., Hassabis, D. & Summerfield, C. Neural mechanisms of hierarchical planning in a virtual subway network. *Neuron* **90**(4), 893–903 (2016).
13. Morgan, L. K., MacEvoy, S. P., Aguirre, G. K. & Epstein, R. A. Distances between real-world locations are represented in the human hippocampus. *J. Neurosci.* **31**(4), 1238–1245 (2011).
14. Epstein, R. A., Patai, E. Z., Julian, J. B. & Spiers, H. J. The cognitive map in humans: Spatial navigation and beyond. *Nat. Neurosci.* **20**(11), 1504–1513 (2017).
15. Park, S. A., Miller, D. S., & Boorman, E. D. Inferences on a multidimensional social hierarchy use a grid-like code. *bioRxiv* 2020–05 (2021).
16. Park, S. A., Miller, D. S., Nili, H., Ranganath, C. & Boorman, E. D. Map making: Constructing, combining, and inferring on abstract cognitive maps. *BioRxiv* 810051 (2020).
17. Schiller, D. *et al.* Memory and space: Towards an understanding of the cognitive map. *J. Neurosci.* **35**(41), 13904–13911 (2015).
18. Bellmund, J. L. S., Gärdenfors, P., Moser, E. I., & Doeller, C. F. Navigating cognition: Spatial codes for human thinking. *Science* **362**(6415), 1–11 (2018).
19. Tulving, E. & Markowitsch, H. J. Episodic and declarative memory: Role of the hippocampus. *Hippocampus* **8**(3), 198–204 (1998).
20. Reddy, L. *et al.* Human hippocampal neurons track moments in a sequence of events. *J. Neurosci.* **41**(31), 6714–6725 (2021).
21. Battaglia, F. P., Benchenane, K., Sirota, A., Pennartz, C. M. A. & Wiener, S. I. The hippocampus: Hub of brain network communication for memory. *Trends Cogn. Sci.* **15**(7), 310–318 (2011).
22. Hickok, G. & Poeppel, D. Dorsal and ventral streams: A framework for understanding aspects of the functional anatomy of language. *Cognition* **92**(1–2), 67–99 (2004).
23. Milivojevic, B., Varadinov, M., Grabovetsky, A. V., Collin, S. H. P. & Doeller, C. F. Coding of event nodes and narrative context in the hippocampus. *J. Neurosci.* **36**(49), 12412–12424 (2016).
24. Morton, N. W. & Preston, A. R. Concept formation as a computational cognitive process. *Curr. Opin. Behav. Sci.* **38**, 83–89 (2021).
25. Collin, S. H. P., Milivojevic, B. & Doeller, C. F. Memory hierarchies map onto the hippocampal long axis in humans. *Nat. Neurosci.* **18**(11), 1562–1564 (2015).
26. Brunec, I. K. & Momennejad, I. Predictive representations in hippocampal and prefrontal hierarchies. *bioRxiv* 786434 (2019).
27. Milivojevic, B. & Doeller, C. F. Mnemonic networks in the hippocampal formation: From spatial maps to temporal and conceptual codes. *J. Exp. Psychol. General* **142**(4), 1231 (2013).
28. Bernardi, S. *et al.* The geometry of abstraction in the hippocampus and prefrontal cortex. *Cell* **183**(4), 954–967 (2020).
29. Momennejad, I. Learning structures: Predictive representations, replay, and generalization. *Curr. Opin. Behav. Sci.* **32**, 155–166 (2020).
30. Whittington, J. C. R. *et al.* The Tolman–Eichenbaum machine: Unifying space and relational memory through generalization in the hippocampal formation. *Cell* **183**(5), 1249–1263 (2020).
31. Stachenfeld, K. L., Botvinick, M. & Gershman, S. J. Design principles of the hippocampal cognitive map. *Adv. Neural Inf. Process. Syst.* **27**, 2528–2536 (2014).
32. Stachenfeld, K. L., Botvinick, M. M. & Gershman, S. J. The hippocampus as a predictive map. *Nat. Neurosci.* **20**(11), 1643 (2017).
33. Momennejad, I. & Howard, M. W. Predicting the future with multi-scale successor representations. *bioRxiv* (2018).
34. De Cothi, W. & Barry, C. Neurobiological successor features for spatial navigation. *BioRxiv* 789412 (2019).
35. McNamee, D. C., Stachenfeld, K. L., Botvinick, M. M. & Gershman, S. J. Flexible modulation of sequence generation in the entorhinal–hippocampal system. *Nat. Neurosci.* **24**(6), 851–862 (2021).
36. Alvernhe, A., Save, E. & Poucet, B. Local remapping of place cell firing in the Tolman detour task. *Eur. J. Neurosci.* **33**, 1696–705 (2011).
37. Piai, V., *et al.* Direct brain recordings reveal hippocampal rhythm underpinnings of language processing. In *Proceedings of the National Academy of Sciences of the United States of America* Vol. 113 11366–11371 (2016).
38. Covington, N. V. & Duff, M. C. Expanding the language network: Direct contributions from the hippocampus. *Trends Cogn. Sci.* **20**, 869–870 (2016).
39. Dayan, P. Improving generalization for temporal difference learning: The successor representation. *Neural Comput.* **5**(4), 613–624 (1993).
40. Van der Maaten, L. & Hinton, G. Visualizing data using t-sne. *J. Mach. Learn. Res.* **9**(11), 2579–2605 (2008).
41. Wattenberg, M., Viégas, F. & Johnson, I. How to use t-sne effectively. *Distill* **1**(10), e2 (2016).
42. Vallejos, C. A. Exploring a world of a thousand dimensions. *Nat. Biotechnol.* **37**(12), 1423–1424 (2019).
43. Moon, K. R. *et al.* Visualizing structure and transitions in high-dimensional biological data. *Nat. Biotechnol.* **37**(12), 1482–1492 (2019).
44. Torgerson, W. S. Multidimensional scaling: I. Theory and method. *Psychometrika* **17**(4), 401–419 (1952).
45. Kruskal, J. B. Nonmetric multidimensional scaling: A numerical method. *Psychometrika* **29**(2), 115–129 (1964).
46. Kruskal, J. B. *Multidimensional scaling*. Vol. 11 (Sage, 1978).
47. Cox, M. A. A. & Cox, T. F. Multidimensional scaling. In *Handbook of data visualization* 315–347. (Springer, 2008).
48. Schilling, A. *et al.* Analysis of continuous neuronal activity evoked by natural speech with computational corpus linguistics methods. *Lang. Cogn. Neurosci.* **36**(2), 167–186 (2021).
49. Schilling, A., Maier, A., Gerum, R., Metzner, C. & Krauss, P. Quantifying the separability of data classes in neural networks. *Neural Netw.* **139**, 278–293 (2021).

50. Krauss, P. *et al.* Analysis and visualization of sleep stages based on deep neural networks. *Neurobiol. Sleep Circadian Rhythms* **10**, 100064 (2021).
51. Krauss, P., Zankl, A., Schilling, A., Schulze, H. & Metzner, C. Analysis of structure and dynamics in three-neuron motifs. *Front. Comput. Neurosci.* **13**, 5 (2019).
52. Krauss, P., Prebeck, K., Schilling, A. & Metzner, C. "Recurrence resonance" in three-neuron motifs. *Front. Computat. Neurosci* vol 64. (2019).
53. Krauss, P., Prebeck, K., Schilling, A. & Metzner, C. "Recurrence Resonance" in Three-Neuron Motifs. *Front. Comput. Neurosci.* **13**(64). <https://doi.org/10.3389/fncom.2019.00064> (2019).
54. Krauss, P. *et al.* A statistical method for analyzing and comparing spatiotemporal cortical activation patterns. *Sci. Rep.* **8**(1), 1–9 (2018).
55. Krauss, P. *et al.* Analysis of multichannel EEG patterns during human sleep: A novel approach. *Front. Hum. Neurosci.* **12**, 121 (2018).
56. Traxdorf, M., Krauss, P., Schilling, A., Schulze, H. & Tziridis, K. Microstructure of cortical activity during sleep reflects respiratory events and state of daytime vigilance. *Somnologie* **23**(2), 72–79 (2019).
57. Fran, cois Chollet *et al.* Keras, 2015.
58. Matthias Plappert. keras-rl, 2016.
59. Harris, C. R. *et al.* Array programming with NumPy. *Nature* **585**(7825), 357–362 (2020).
60. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
61. Hunter, J. D. Matplotlib: A 2d graphics environment. *Comput. Sci. Eng.* **9**(3), 90–95 (2007).
62. Hagberg, A. A., Schult, D. A. & Swart, P. J. Exploring network structure, dynamics, and function using networkx. In *Proceedings of the 7th Python in Science Conference* (eds Varoquaux, G., Vaught, T. & Millman, J.) 11–15 (Pasadena, CA, USA, 2008).
63. Krupic, J., Bauza, M., Burton, S., Barry, C. & O'Keefe J. Grid cell symmetry is shaped by environmental geometry. *Nature* **518** (2015).
64. Stachenfeld, K. L., Botvinick, M. M. & Gershman, S. J. The hippocampus as a predictive map. *Nat. Neurosci.* **21**(6), 232–235 (2017).
65. O'Keefe, J. & Burgess, N. Geometric determinants of the place fields hippocampal neurons. *Nature* **381**, 425–428 (1996).
66. Fyhn, M., Hafting, T., Witter, M. P., Moser, E. I. & Moser, M.-B. Grid cells in mice. *Hippocampus* **18**(12), 1230–1238 (2008).
67. Brun, V. H. *et al.* Progressive increase in grid scale from dorsal to ventral medial entorhinal cortex. *Hippocampus* **18**(12), 1200–1212 (2008).
68. Gerum, R. C., Erpenbeck, A., Krauss, P. & Schilling, A. Sparsity through evolutionary pruning prevents neuronal networks from overfitting. *Neural Netw.* **128**, 305–312 (2020).
69. Sutton, R. S. & Barto, A. G. *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, 2018).
70. Goldberg, A. E. *Constructions: A construction Grammar Approach to Argument Structure* (University of Chicago Press, 1995).
71. Goldberg, A. E. Constructions: A new theoretical approach to language. *Trends Cogn. Sci.* **7**(5), 219–224 (2003).
72. Goldberg, A. E. *Constructions at Work: The Nature of Generalization in Language* (Oxford University Press on Demand, 2006).
73. Goldberg, A. & Goldberg, A. E. *Explain me this* (Princeton University Press, Princeton, 2019).
74. Diessel, H., Dabrowska, E. & Divjak, D. Usage-based construction grammar. *Cogn. Linguist.* **2**, 50–80 (2019).
75. Diessel, H. A dynamic network approach to the study of syntax. *Front. Psychol.* 3196 (2020).
76. Levshina, N. The grammar network: How linguistic structure is shaped by language use by holger diessel. *Language* **97**(4), 825–830 (2021).
77. Diessel, H. A dynamic network approach to the study of syntax *Front. Psychol.* **11**, 604853 (2020).
78. Kuhl, P. K. Brain mechanisms in early language acquisition. *Neuron* **67**, 713–727 (2010).
79. Dabrowska, E. Implicit lexical knowledge. *Linguistics* **52**(1), 205–223 (2014).
80. Garibyan, A., Schilling, A., Boehm, C., Zankl, A. & Krauss, P. Neural correlates of linguistic collocations during continuous speech perception. *bioRxiv* (2022).
81. Bermudez-Contreras, E., Clark, B. J. & Wilber, A. The neuroscience of spatial navigation and the relationship to artificial intelligence. *Front. Comput. Neurosci.* **14**, 63 (2020).
82. Kriegeskorte, N. & Douglas, P. K. Cognitive computational neuroscience. *Nat. Neurosci.* **21**(9), 1148–1160 (2018).
83. Schilling, A., Gerum, R., Zankl, A., Schulze, H., Metzner, C., & Krauss, P. Intrinsic noise improves speech recognition in a computational model of the auditory pathway. *bioRxiv* (2020).
84. Krauss, P., Tziridis, K., Schilling, A. & Schulze, H. Cross-modal stochastic resonance as a universal principle to enhance sensory processing. *Front. Neurosci.* **12**, 578 (2018).
85. Hassabis, D., Kumaran, D., Summerfield, C. & Botvinick, M. Neuroscience-inspired artificial intelligence. *Neuron* **95**(2), 245–258 (2017).
86. Marcus, G. Deep learning: A critical appraisal. arXiv preprint [arXiv:1801.00631](https://arxiv.org/abs/1801.00631) (2018).
87. Yang, Z., Schilling, A., Maier, A. & Krauss, P. Neural networks with fixed binary random projections improve accuracy in classifying noisy data. In *Bildverarbeitung für die Medizin 2021* 211–216 (Springer, 2021).
88. Gerum, R. C. & Schilling, A. Integration of leaky-integrate-and-fire neurons in standard machine learning architectures to generate hybrid networks: A surrogate gradient approach. *Neural Comput.* **33**(10), 2827–2852 (2021).
89. Maier, A. K. *et al.* Learning with known operators reduces maximum error bounds. *Nat. Mach. Intell.* **1**(8), 373–380 (2019).

## Acknowledgements

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation): grant KR5148/2-1 to PK—project number 436456810.

## Author contributions

P.S. and C.S. performed computer simulations. P.K. and A.M. designed and supervised the study. A.S. and C.M. provided analysis tools. All authors discussed the results and wrote the manuscript.

## Funding

Open Access funding enabled and organized by Projekt DEAL.

## Competing interests

The authors declare no competing interests.

## Additional information

**Correspondence** and requests for materials should be addressed to P.K.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

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



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022