Learning, Storing, and Disentangling Correlated Patterns in Neural Networks

Xiaolong Zou[1], Zilong Ji[2], Xiao Liu[2], Tiejun Huang[1], Yuanyuan Mi[3]
Dahui Wang[2,4], and Si Wu[1](✉)

[1] School of Electronics Engineering and Computer Science,
IDG/McGovern Institute for Brain Research,
Peking University, Beijing 100871, China
siwu@pku.edu.cn

[2] State Key Laboratory of Cognitive Neuroscience & Learning, Beijing Normal University, Beijing 100875, China

[3] Institute for Neurointelligence, School of Medicine,
Chongqing University, Chongqing, China

[4] School of Systems Science, Beijing Normal University, Beijing 100875, China

Abstract. The brain encodes object relationship using correlated neural representations. Previous studies have revealed that it is a difficult task for neural networks to process correlated memory patterns; thus, strategies based on modified unsupervised Hebb rules have been proposed. Here, we explore a supervised strategy to learn correlated patterns in a recurrent neural network. We consider that a neural network not only learns to reconstruct a memory pattern, but also holds the pattern as an attractor long after the input cue is removed. Adopting backpropagation through time to train the network, we show that the network is able to store correlated patterns, and furthermore, when continuously morphed patterns are presented, the network acquires the structure of a continuous attractor neural network. By inducing spike frequency adaptation in the neural dynamics after training, we further demonstrate that the network has the capacities of anticipative tracking and disentangling superposed patterns. We hope that this study gives us insight into understanding how neural systems process correlated representations for objects.

Keywords: Neural network · Correlated patterns
Continuous attractor neural network · Backpropagation through time
Spike frequency adaptation

1 Introduction

In reality, the brain needs to encode not only the identities of objects, e.g., whether an animal is cat or dog, but also the relationships between objects, e.g., cat and dog are both mammalian but belong to different categories. The experimental data has indicated that the categorical relationships between objects are

X. Zhu and Z. Ji—Equal contribution.

L. Cheng et al. (Eds.): ICONIP 2018, LNCS 11303, pp. 506–514, 2018.
https://doi.org/10.1007/978-3-030-04182-3_44

encoded in the correlated neural representations of the objects, in term of that for objects belonging to the same category, their neural representations have larger correlations than those of objects belonging to different categories [1,2]. Interestingly, in an artificial deep neural network (DNN) trained by ImageNet, the correlation between object representations (measured by the overlap between activities in the representation layer, i.e., the one before the read-out layer) also reflects the semantic similarity between the objects [3,4]. To understand how a neural system encodes the relationship between objects, it is important to understand how neural networks learn, store, and retrieve correlated memory patterns.

A large volume of theoretical studies has, however, pointed out that it is not a trivial task for a neural network to process correlated memory patterns [5–8]. These studies, which are based on the classical Hopfield model that constructs neuronal connections according to the unsupervised Hebb rule, have shown that the correlations between patterns deteriorate memory retrieval dramatically, leading to that the Hopfield network is unable to support a large memory capacity [5]. To overcome this flaw, several strategies have been proposed, which include: (1) a novelty-based method [6], which considers that neuronal connections are modified only when a novel pattern is presented (the novelty is defined according to that the pattern can be retrieved or not by the current network structure); (2) a popularity-based method [7], which modifies the Hebb rule by reducing the contributions of those popular neurons that are active in many memory patterns to avoid overwhelmed learning of the connections of those neurons; (3) an orthogonalization-based method [8], which orthogonalizes correlated patterns before applying the Hebb rule. All these methods are based on the unsupervised Hebb learning, and each of them works well in certain circumstances, but their biological plausibility has yet been properly justified.

In the present study, we explore the possibility of using a supervised strategy to train a recurrent neural network to learn correlated patterns. Specifically, we consider a computational task, in which the network not only learns to reconstruct the presented input pattern, but also holds the pattern as persistent activity long after the input is removed. Mathematically, this requires that the network holds the pattern as an attractor of its dynamics. We use backpropagation through time (BPTT) [9] to train the network and demonstrate that the network learns to store a number of highly correlated patterns. Moreover, we find that when a set of continuously morphed patterns are presented, the network acquires the structure of a continuous attractor neural network (CANN), a canonical model for neural information processing [10]. After training, we induce spike frequency adaptation (SFA), a popular negative feedback modulation [11], in the neural dynamics, and find that the network holds interesting computational properties, including anticipative tracking and the capacity of disentangling superposed patterns. We hope this study enriches our knowledge of how neural systems process correlated representations for objects.

2 The Model

As illustrated in Fig. 1, the network model we consider consists of three layers: input, recurrent, and output layers. Neurons in the recurrent layer are connected recurrently, whose dynamics are written as follows,

$$\tau \frac{du_i(t)}{dt} = -u_i(t) + \sum_{j=1}^{N} W_{ij}^{rec} x_j(t) + \sum_{k=1}^{N_{in}} W_{ik}^{in} I_k(t) + b_i + \sigma \xi_i(t), \quad (1)$$

$$x_i(t) = \tanh\left[u_i(t)\right], \quad (2)$$

where u_i, for $i = 1, \ldots, N$, is the synaptic input received by neuron i in the recurrent layer and x_i the activity of the neuron. τ is the time constant. N is the number of neurons in the recurrent layer. W_{ij}^{rec} denotes the recurrent connection strength from neuron j to i, W_{ik}^{in} the feedforward connection strength from input component k to neuron i, I_k the external input, and N_{in} the input dimension. b_i is a biased constant input received by neuron i. $\xi_i(t)$ represents Gaussian white noise of zero mean and unit variance, and σ the noise strength.

The neurons in the output layer read-out information by combining the neuronal activities in the recurrent layer linearly, which are written as

$$y_i(t) = \sum_{j=1}^{N} W_{ij}^{out} x_j(t), \quad (3)$$

where y_i, for $i = 1, \ldots, N_{out}$, represents the activity of neuron i in the output layer, and W_{ij}^{out} the read-out connection strength from neuron j in the recurrent layer to neuron i in the output layer. $N_{out} = N_{in}$ holds in our model.

The Learning Procedure

Our goal is to train the network, such that the network holds the predefined memory patterns as attractors of its dynamics. To achieve this goal, we construct a learning task, which requires that the network output not only reconstructs the given input pattern, but also holds the pattern long after the input cue is removed. Mathematically, these two conditions enforce that the network learns to hold the input pattern as an attractor of its dynamics, which mimics the persistent activity observed in working memory in neural systems [12].

Let us consider that the network learns to memorize M patterns, referred to as $\mathbf{P}^i$, for $i = 1, \ldots, M$, hereafter. Denote T_{sti} the duration of presenting each memory pattern as an external input to the network, T_{seq} the duration of the network holding the memory pattern, and $T_{sti} << T_{seq}$ is imposed. For a memory pattern $\mathbf{P}^i$, the corresponding external input to the network is given by, $\mathbf{I}^i(t) = \mathbf{P}^i + \eta^i(t)$, for $0 < t < T_{sti}$ and otherwise $\mathbf{I}^i(t) = 0$. Here, η^i represent Gaussian white noises, which have the same dimensionality as the input and its elements are independently sampled from Gaussian distributions of zero mean and variances uniformly distributed in the range of $[0, 1]$. These noises are essential for robust learning. Denote the network output to be $\mathbf{Y}^i(t)$,

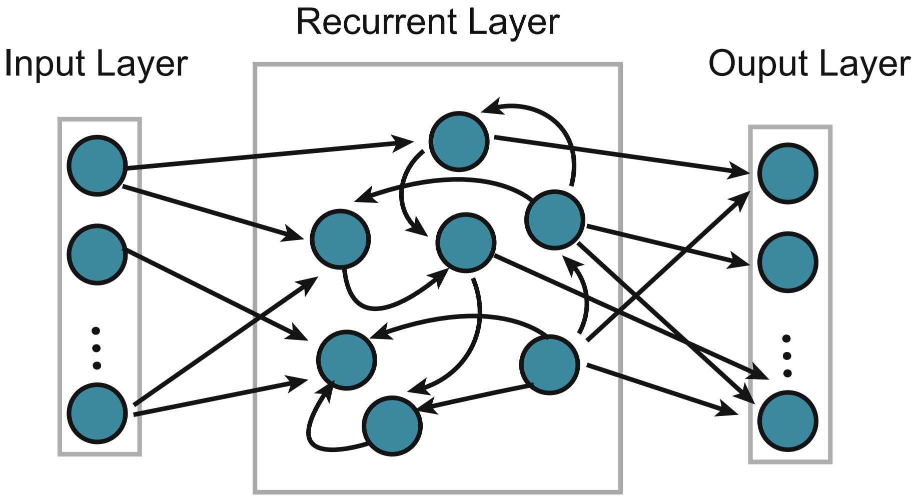

Fig. 1. Network structure. The network contains an input, a recurrent, and an output layers. Neurons in the recurrent layer are connected recurrently.

for $0 < t < T_{seq}$, when the pattern $\mathbf{P}^i$ is presented. The objective function of the learning task is written as,

$$L = \sum_{i=1}^{M} \int_0^{T_{seq}} \left[\mathbf{Y}^i(t) - \mathbf{P}^i\right]^2 dt, \quad \mathbf{I}^i(t) \neq 0, \text{ for } 0 < \mathrm{t} < \mathrm{T}_{\mathrm{sti}}, \tag{4}$$

where $\mathbf{Y}^i(t) = f\left[\mathbf{I}^i(t)\right]$ represents the nonlinear function implemented by the network. We use the Euler method to discrete the network dynamics and the objective function, and adopt backpropagation through time (BPTT) to optimize the network parameters, including the connection weights $\mathbf{W}^{in}, \mathbf{W}^{rec}, \mathbf{W}^{out}$ and the bias terms $\mathbf{b}$. Before training, $\mathbf{W}^{out}$ are initialized to be zeros, $\mathbf{W}^{in}$ a Gaussian distribution of zero mean and unit variance, $\mathbf{W}^{rec}$ an orthogonal and normalized matrix, and $\mathbf{b}$ zeros.

Spike Frequency Adaptation

After training, we add spike frequency adaptation (SFA) in the neural dynamics to induce extra computational properties of the network. With SFA, Eq. (1) becomes

$$\tau \frac{du_i(t)}{dt} = -u_i(t) + \sum_{j=1}^{N} W_{ij}^{rec} x_j(t) + \sum_{k=1}^{N_{in}} W_{ik}^{in} I_k(t) + b_i + \sigma\xi_i(t) - v_i(t), \tag{5}$$

$$\tau_v \frac{dv_i(t)}{dt} = -v_i(t) + m u_i(t), \tag{6}$$

where $v_i(t)$ is the current induced by SFA, a negative feedback modulation widely observed in neural systems [11], whose effect is to suppress neuronal responses when they are too strong. τ_v is the time constant of SFA, and $\tau_v >> \tau$ implies that SFA is a slow process compared to neural firing. The parameter m controls the amplitude of SFA.

3 Results

3.1 Learning to Memorize Correlated Patterns

To demonstrate that our model is able to learn correlated memory patterns, we chose handwriting digit numbers as the inputs (see Fig. 2). These image patterns are highly correlated (overlapped) and hence can not be memorized by the conventional Hopfield model. We test three unsupervised strategies, and found that the orthogonalization-based method accomplished the task, but the other two, novelty-based and popularity-based, failed. Our supervised strategy accomplished the task successfully (Fig. 2).

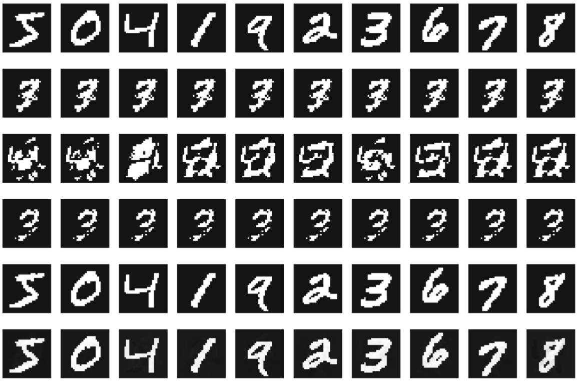

Fig. 2. Retrieval performances of different learning methods. From top to bottom: the original ten digit numbers from the dataset of Mnist, the retrieval of the conventional Hopfield model, the retrieval of the popularity-based method, the retrieval of the novelty-based method, the retrieval of the orthogonalization-based method, and the retrieval of our approach. Parameters: $\tau = 5, T_{sti} = 3, T_{seq} = 30, \sigma = 0.01, N = 200, N_{in} = N_{out} = 784$.

3.2 Disentangling Superposed Memory Patterns

After training the network to memorize ten digit numbers, we add SFA in the neural dynamics (see Eq. (5, 6)). In a real neural system, this corresponds to that during learning, SFA is either frozen or too slow and can be ignored compared to the fast synaptic plasticity. We check the network responses when an image of superposed two digit numbers is presented. As illustrated in Fig. 3, the network outputs the two digit numbers alternatively over time. The underlying mechanism is that: (1) through training, the network has learned to memorize the two digit numbers as its attractors; (2) when the ambiguous image is presented, the network receives two competing input cues and evolves into one of two attractors depending on biases; (3) because of the negative feedback from SFA, the network state becomes unstable gradually, and under the competition from the other cue, the network state moves into the other attractor; (4) this progress goes on, and the network state oscillates between two attractors. Our study suggests that a neural network can use negative feedback such as SFA to disentangle correlated patterns.

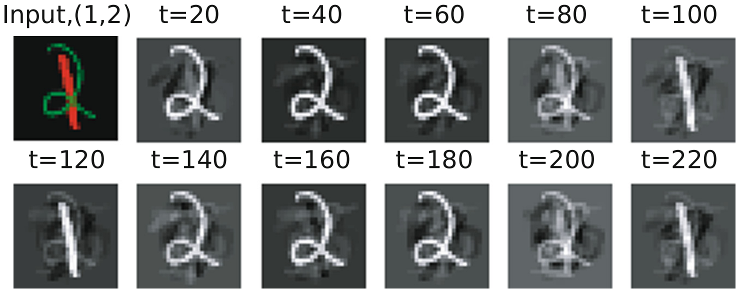

Fig. 3. Disentangling superposed correlated memory patterns. The network was first trained to memorize ten digit numbers as in Fig. 2. For convenience, we use colors to differentiate different digit numbers, but in practice gray images are used. The input is the superposed images of 1 and 2. The evolving of the network output over time is presented. Parameters are: $m = 3.4, \tau_v = 30$. Other parameters are the same as in Fig. 2.

3.3 Learning a CANN

We show that when continuously morphed patterns are memorized, the network acquires the structure of a CANN. Figure 4A displays the set of continuously morphed gaussian bumps to be memorized by the network. After training, the network leans to store each of them as attractors, in terms of that: (1) the network evolves to one-to-one mapped stationary state when each of gaussian bumps is presented (Fig. 4A); and (2) the network remains to be at the active state long after the input is removed (Fig. 4B).

Properties of the Network

We check that the learned network indeed has the good computational properties of CANNs. Figure 5 shows that the network has the properties: (1) mental rotation [13,14], the network exhibits the mental rotation behavior when the external inputs abrupt change (Fig. 5A); (2) travelling wave [15], the network holds a self-sustained travelling wave when SFA is strong enough (Fig. 5B); (3) anticipation tracking [15], the network is able to track a moving input anticipatively if SFA is strong enough (Fig. 5C).

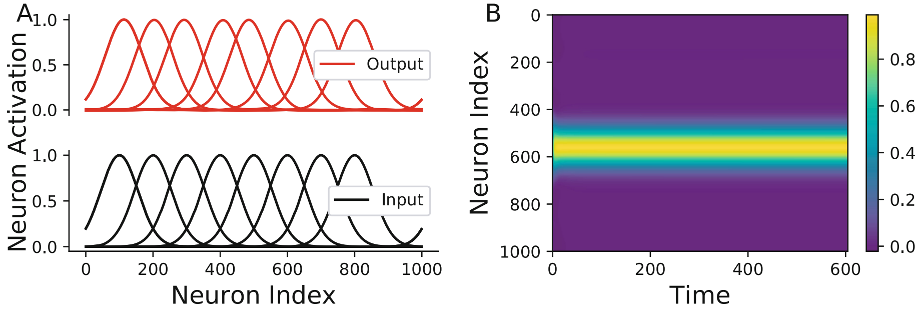

Fig. 4. Learning continuously morphed patterns to form a CANN. Totally 1000 morphed gaussian-bump-like patterns are constructed. (A) Lower panel: examples of input patterns; upper panel: examples of the learned network output. There are one-to-one correspondence between the inputs and outputs. (B) The activity map of neurons in the output layer. The input is removed at $T = 3$. The network state is sustained after the input is removed, indicating the existence of an attractor. Parameters are: $N_{in} = N_{out} = 1000$. Other parameters are the same as in Fig. 2.

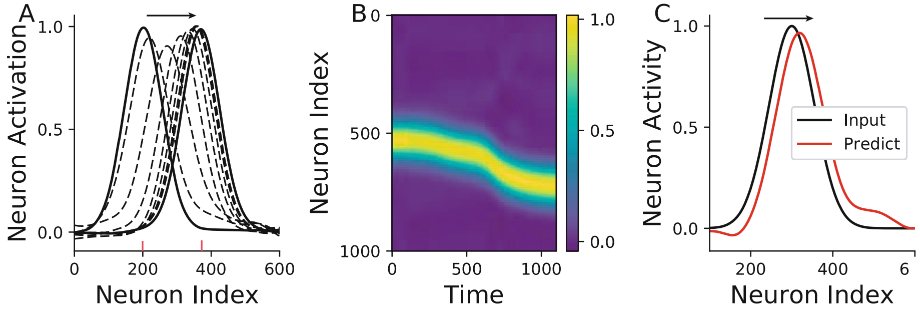

Fig. 5. Properties of the learned network. (A) Mental rotation. The network state is initially at pattern index 200. Under the drive of an external input at pattern index 360, the network state smoothly rotates from the initial to the target positions. (B) Travelling wave. The activity map of the output layer in the travelling state. $m = 0.4$. (C) Anticipate tracking. The black curve is the external moving input, and the red curve the network state which leads the moving input. $m = 0.4$. Parameters are: $\tau_v = 60, N_{in} = N_{out} = 1000$. Other parameters are the same as in Fig. 2. (Color figure online)

4 Conclusion

In the present study, we have investigated a supervised strategy to learn correlated patterns in neural networks, which are different from the unsupervised ones proposed in the literature. The key idea of our method is that we enforce the network to learn the memory patterns as its attractors. To achieve this goal, we require that the network not only learns to reconstruct the given input pattern, but also holds the pattern as persistent activity long after the input cue is removed. Using both synthetic and real data, we show that after training, the network is able to store highly correlated patterns and can also acquire the structure of a CANN if continuously morphed patterns are presented. Moreover, we induce SFA in the neural dynamics after training, and demonstrating that the network holds interesting computational properties, including anticipative tracking and the capacity of disentangling superposed patterns. We hope this study, as to a complement to other unsupervised approaches, enrich our knowledge of how neural systems process correlated representations for objects.

Acknowledgments. This work was supported by National Key Basic Research Program of China (2014CB846101), BMSTC (Beijing municipal science and technology commission) under grant No: Z161100000216143 (SW), Z171100000117007 (DHW & YYM); the National Natural Science Foundation of China (No: 31771146, 11734004, YYM), Beijing Nova Program (No: Z181100006218118, YYM).

References

1. Huth, A.G., Nishimoto, S., Vu, A.T., et al.: A continuous semantic space describes the representation of thousands of object and action categories across the human brain. Neuron **76**(6), 1210–1224 (2012)
2. Chang, L., Tsao, D.Y.: The code for facial identity in the primate brain. Cell **169**(6), 1013–1028 (2017)
3. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013)
4. Yosinski, J., Clune, J., Nguyen, A., et al.: Understanding neural networks through deep visualization. Computer Science (2015)
5. Hertz, J., Krogh, A., Palmer, R.G.: Introduction to the Theory of Neural Computation. The Advanced Book Program (1991)
6. Blumenfeld, B., Preminger, S., Sagi, D., et al.: Dynamics of memory representations in networks with novelty-facilitated synaptic plasticity. Neuron **52**(2), 383–394 (2006)
7. Kropff, E., Treves, A.: Uninformative memories will prevail: the storage of correlated representations and its consequences. HFSP J. **1**(4), 249–262 (2007)
8. Zou, X., Ji, Z., Liu, X., Mi, Y., Wong, K.Y.M., Wu, S.: Learning a continuous attractor neural network from real images. In: Liu, D., Xie, S., Li, Y., Zhao, D., El-Alfy, E.S. (eds.) Neural Information Processing. ICONIP 2017. LNCS, vol. 10637. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70093-9_66
9. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: International Conference on Machine Learning. JMLR.org, III-1310 (2013)
10. Wu, S., Wong, K.Y.M., Fung, C.C.A., et al.: Continuous attractor neural networks: candidate of a canonical model for neural information representation. F1000Research, 5 (2016)
11. Gutkin, B., Zeldenrust, F.: Spike frequency adaptation. Scholarpedia **9**(2), 30643 (2014)
12. Curtis, C.E., D'Esposito, M., Curtis, C.E.: Persistent activity in the prefrontal cortex during working memory. Trends Cognit. Sci. **7**(9), 415–423 (2003)
13. Shepard, R.N., Metzler, J.: Mental rotation of three-dimensional objects. Science **171**(3972), 701–703(1971)

14. Fung, C.C.A., Wong, K.Y.M., Wu, S.: A moving bump in a continuous manifold: a comprehensive study of the tracking dynamics of continuous attractor neural networks. Neural Comput. **22**(3), 752 (2010)
15. Mi, Y., Fung, C.C.A., Wong, K.Y.M., et al.: Spike frequency adaptation implements anticipative tracking in continuous attractor neural networks. In: Advances in Neural Information Processing Systems, vol. 1, no. 3, pp. 505–513 (2014)