BHN: A Brain-like Heterogeneous Network

Liu, Tao

Abstract

The human brain works in an unsupervised way, and more than one brain region is essential for lighting up intelligence. Inspired by this, we propose a brain-like heterogeneous network (BHN), which can cooperatively learn a lot of distributed representations and one global attention representation. By optimizing distributed, self-supervised, and gradient-isolated objective functions in a minimax fashion, our model improves its representations, which are generated from patches of pictures or frames of videos in experiments.

1 Introduction

It is a mystery how different brain regions to be optimized jointly. In this article, we propose a brain-like heterogeneous network(BHN) simulating the multi-module structure of the brain.

We use three hypothesises in this article:

1. The brain is a machine to maximize information of its inner representations. This hypothesis was known as Efficient Coding[Barlow et al., 1961] or Efficient Information Representation[Linsker, 1990, Atick, 1992].
2. The brain learns by optimizing certain objective functions, and different brain regions optimize different objective functions[Lake et al., 2017].
3. The brain works by fusing top-down predictions with bottom-up perceptions. This hypothesis actually enables the brain to process information recursively.

We view hypothesis 1 as the first principle to understand the brain and obtain desired objective functions by formalizing it. The objective is a sum of many objective functions, each applied on an individual module, and all the modules make up BHN. We also seek to understand the brain's information processing scheme, which we name as Recursive Modeling in this article.

Following in this article, firstly, the section 2 will give the objective functions derived from the first hypothesis. Next, the section 3 and the section 5 will elaborate on BHN and Recursive Modeling respectively. And then the section 4 and the section 6 will provide some demonstration experiments for the former two sections respectively.

2 Efficient Information Representation

The brain collects information from the environment(x) and then generates internal representations(z). It is inferred that an important function of the brain is to maximize the information entropy of its representations. It is generally believed that these representations are distributed on the cerebral cortex, and so it is essential to ensure the independence of information they represent. Previous solutions include sparse-coding[Olshausen and Field, 1996], independent component analysis[Hyvärinen and Oja, 2000], and end-to-end deep learning. In this article we propose our solution as follows.

Preprint. Under review.

We use$\{z^1, z^2, \cdots, z^n\}$ to denote the representations distributed on the cerebral cortex, and use $H(z^1 z^2 \cdots z^n)$ to denote the information entropy of them. We then formalize the objective function as $\max H(z^1 z^2 \cdots z^n)$.

Considering

$$H(z^1 z^2 \cdots z^n) = \sum_i H(z^i) + [H(z^1 z^2 \cdots z^n) - \sum_i H(z^i)] \tag{1}$$

the objective function can be roughly decomposed into two sub-objectives[Atick, 1992], as

$$\begin{cases} \max\limits_z \ H(z^i) \\ \min\limits_z \ I(z^i; z^j), \quad \text{if } i \neq j \end{cases} \tag{2}$$

Noting that the second sub-objective is intractable because of the $\Omega(n^2)$ computational complexity, so we introduce a **global** attention[Graves et al., 2014, Vaswani et al., 2017] representation(a) into (2) by reforming the expression in a minimax fashion, as

$$\begin{cases} \max\limits_z \ \sum_i H(z^i) \\ \min\limits_z \max\limits_a \ \sum_i I(z^i; a) \end{cases} \tag{3}$$

Then, by re-composing the two expressions above into a single one, we obtain the objective function:

$$\min_z \max_a \sum_i [-H(z^i) + I(a; z^i)] \tag{4}$$

We use contrastive losses[Hadsell et al., 2006] to formalize $H(z^i)$. Contrastive losses measure the similarities of sample pairs in a representation space. A form of a contrastive loss function, which is called InfoNCE[Oord et al., 2018], is considered in this article:

$$H(z^i) \propto \log \frac{\exp(f(z^i, z^i_+))}{\sum_{z^i_- \in Z^i} \exp(f(z^i, z^i_-))} \tag{5}$$

where f is a density ratio, which preserves the mutual information between a positive or negative pair of samples.

The next step is to formalize $I(a; z^i)$. To stabilize minimaxing on $I(a; z^i)$, we do not formalize it directly. Instead, we use a to produce a probability distribution, i.e. $P(z^i)$, as the prediction of z^i. The a is called as an attention representation because it is used to generate shared Query/Key vectors a^i, each of which is paired with a representation z^i, and these Q/K vectors will be used to calculated each sample's probability/weight. The details are as follows:

We provide a memory pool having N paired samples, as

$$X^i = (A^i, Z^i) = \{(a^i_1, z^i_1), (a^i_2, z^i_2), \cdots, \cdots, (a^i_N, z^i_N)\} \tag{6}$$

where Z^i is the sample space of $P(z^i)$. The probability $P\big|z^i = z^i_j$ is equal to the attention weight w^i_j calculated by

$$P\big|_{z^i = z^i_j} = w^i_j = \text{softmax}\left(similarity(a^i, a^i_j)\right)\Big|_{a^i_j \in A^i} \tag{7}$$

Now we can formalize $I(a; z^i)$ as

$$I(a; z^i) \propto \log \frac{\sum_{z^i_j \in Z^i} w^i_j \exp(f(z^i, z^i_j))}{\sum_{z^i_- \in Z^i} \exp(f(z^i, z^i_-))} \tag{8}$$

Eventually, by bringing (5) and (8) into (4), we formalize the objective function as

$$\min_z \max_a \sum_i [-\log \frac{\exp(f(z^i, z^i_+))}{\sum_{z^i_j \in Z^i} w^i_j \exp(f(z^i, z^i_j))}] \tag{9}$$

Notable, this objective function suggests a probabilistic inference machines[von Helmholtz, 1925] and it is the corollary of our hypothesises. This is biologically plausible and we can say that **the attention representation makes predictions by activating selective replays of representations in the cerebral cortex**.

3 Brain-like Heterogeneous Network

In the section we propose the architecture of BHN to apply the objective function in Equation 9. It has a cortex-network composed of basic units, with unit i generating the corticocerebral representation z^i, and an attention-network generating the global attention representation a. As the name suggests, we use artificial neural network(ANN) to implement these two components. Different from popular approach using end-to-end back-propagation with a global loss function, in our model, there is gradient isolation between the units and between the two networks.

Cortex-network In each unit i, there is an encoder g^i_{enc} to encode the input x into a latent representation z^i. In the following image task, there only is a g^i_{enc} inside the unit. While in the video tasks, in each unit, another network, which is called aggregator g^i_{ar}, is used to output a unit context c^i to act as the positive partner of the z^i. Actually, we are applying Contrastive Predictive Coding[Oord et al., 2018] in each unit, as shown in Figure 1.

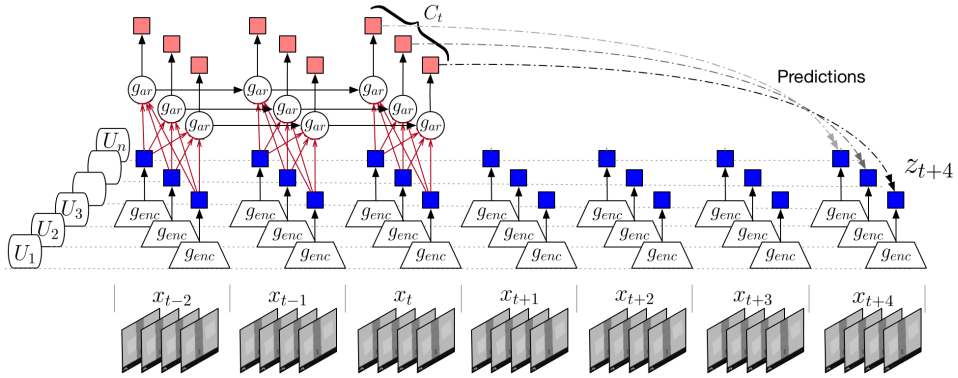
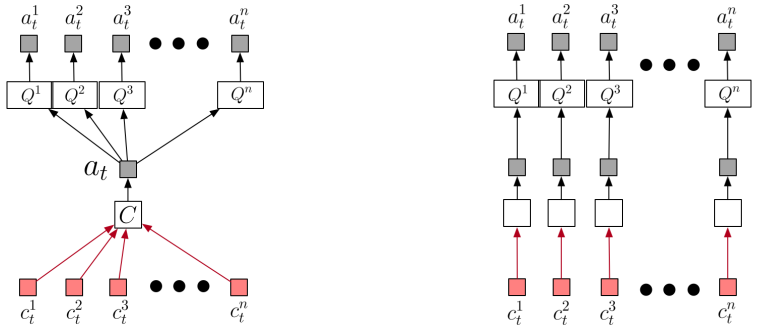


Figure 1: Architecture of the cortex-network in video tasks

Attention-network The attention-network generates the global attention representation a, like the medial temporal lobe in the mammalian brain. Its architecture is like a traditional encoder-decoder network, where the encoder generates a and the the decoder generates a^i, as shown in Figure 2a.



(a) (b) control group 2

Figure 2: Architecture of the attention-network in video tasks

The input of the attention-network is from the output of the cortex-network. In our image task, it is natural to take all z^i as input because they are the only outputs of the cortex-network. In our video tasks, the attention-network takes all c^i as input, because we want to get a in advance of z^i.

The attention representation a should not retain all the information inputted into it, but only needs to capture the information shared by multiple units. To achieve this goal, one option is to make a act as an information bottleneck, which means that a is lower dimensional than the input vector. The other option is to arbitrarily drop out some units' outputs. In our image task, we adopt the second option, and in our video tasks, we adopt the first one.

Neural Interface Neural interface is not an essential component of BHN, but we want to mention it here in advance because it is important for the Recursive Modeling. Unlike the cortex-network and the attention-network, Neural Interfaces have no biological counterparts. Actually, this name comes from Brain-Computer Interfaces(BCIs) [Wolpaw et al., 2000]. By processing information from the cortex-network, neural interfaces perform various functions, such as controlling attention, controlling actions, and whatever as you need.

4 Experiment(1)

4.1 Image Task

We download ten landscape pictures from the internet and crop them into 8000 patches of 16×16 pixels, and then each patch is gray-scaled and normalized. We then design a BHN model to learn on this dataset. The model has 64 units in its cortex-network. The encoder in each unit contains 128 hidden units with leaky-relu activation, and the attention-network contains 256 hidden units, which is also the dimension of a, with leaky-relu activation. The dimensions of z^i and a^i are both set to 1. The batch size, which is also the size of X^i, is 512.

The density ratio f is formalized as

$$f(z, \mathfrak{z}) = -\text{clamp\_max\_5}(|z - \mathfrak{z}|) \tag{10}$$

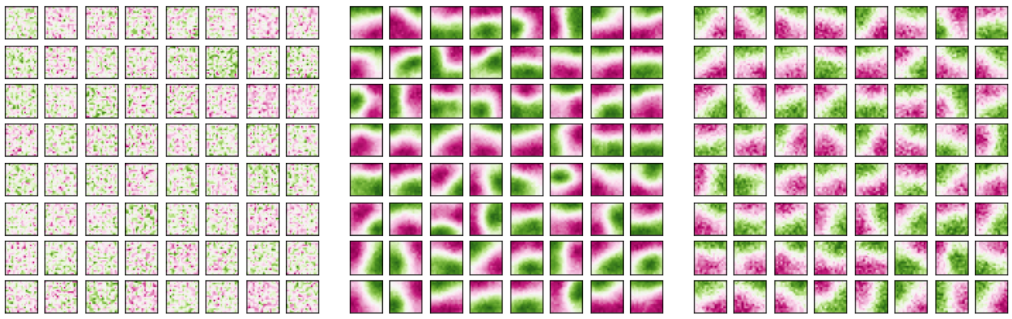
The similarity between a^i is formalized as

$$similarity(a^i, a^i_j) = -|a^i - a^i_j|/\tau \tag{11}$$

where τ is the temperature optimized together with the attention-network.

The inputs are added with Gaussian white noise with $mean = 0$ and $std = 0.1$ for image enhancement, and also in this way to produce positive sample pairs in constrastive loss function. The dropout ratio is 0.2 in the attention-network. We use SGD optimizer with the $lr = 0.1$, $momentum = 0.9$, and $weight\_decay = 0.001$. The model is light and the training runs fast even in a laptop without GPU acceleration.

In addition to the normal experiment, we also establish a control experiment where the objective function is to $\max \sum_i H(z^i)$ only. After 40 epochs of training, we visualize all 64 units by maximizing their outputs. The results are shown in Figure 3.



(a) untrained (b) normal (c) control

Figure 3: Visualized features of units

In order to show clearly, we use red and green to indicate light and shade. As can be seen from the figure, the visualized features are noisy if the model is not trained. In both normal and control experiments, the units have intensified responses to certain image modes after training. The result images in the control experiment are fuzzy. In contrast, the visualized features in the normal experiment are more sharp and diverse.

4.2 Video Task

We build a video set containing 64 episodes recording the play of CarRacing game in OpenAI gym. Each episode lasts for 512 frames and each frame has a size of (96, 96) pixels. The frames are

converted to gray scale and rescaled to (-1,1). At each time step, 4 consecutive frames with additional noises are fed to the input.

A linear layer, shared by all g^i_{enc} for the consideration of reducing the number of parameters, will first reduce the dimensions of inputs from $4 \times (96 \times 96)$ to 512. The encoder architecture g^i_{enc} contains 32 hidden units with leaky-relu activations. We then use a GRU-RNN [Cho et al., 2014] for the autoregressive part of the unit, g^i_{ar}, with 32 dimensional hidden state. The cortex-network has 16 units, and the attention-network is a simple unbiased linear network with a hidden layer. Dimensions of z^i_t, c^i_t , a^i_t and a_t are all set to 2. The batch size, which is also the size of X^i, is 256.

In our experiment, z^i_{t+4} and c^i_t are used as the positive pair for the contrastive loss function. The delay of 4 is, somewhat arbitrary, to quantify the directional information between z and c.

The density ratio f is formalized as

$$f(z^i_{t+4}, c^i_t) = -\cos\langle z^i_{t+4}, c^i_t\rangle / T \tag{12}$$

where T is the temperature optimized together with the cortex-network.

The similarity between a^i is formalized as

$$similarity(a^i, a^i_j) = -\cos\langle a^i, a^i_j\rangle / \tau \tag{13}$$

where τ is the temperature optimized together with the attention-network.

We choose Adam optimizer with $lr = 1e-4$. We use data enhancement in which each episode is folded into 16 segments of 256 frames long. We train each model for 20 epochs. However, according to our experience, a much longer training would not lead to over-fitting.

We use deconvolutional networks[Zeiler et al., 2011] to reconstruct images from representations z_t, c_t and a_t respectively. Mean square errors(mse) of the reconstructed images will be used to evaluate the quality of source representations. Given that a trivial solution could achieve a loss of 0.0225 if none information was provided, in the following, we use the score, calculated by $(0.0225 - mse) \times 255$, to indicate the quality.

We also establish two control groups to demonstrate the performance of adversarial training.

control groups 1 We abandon the attention-network to only perform optimizations on $H(z^i)$, just in the same way as the control group established in section 4.2.

control groups 2 We design a restricted attention-network architecture by cutting off the links via a between units, as shown in Figure 2b.

Table 1 gives the scores of z_t, c_t and a_t before and after training. The scores of the experimental group surpass those of its competitors.

Table 1: Scores of representations

	z_t	c_t	a_t
Before Training	2.04 ± 0.06	2.17 ± 0.04	0.29 ± 0.23
Experimental Group	3.13 ± 0.04	2.33 ± 0.07	0.80 ± 0.23
Control Group 1	2.93 ± 0.07	1.81 ± 0.23	
Control Group 2	2.93 ± 0.07	1.92 ± 0.21	

5 Recursive Modeling

Model building, arguably, is the approach to general intelligence[Lake et al., 2017]. Additionally, we think recursion is essential in the design of strong artificial intelligence, just as it is for many Turing complete machines[Turing, 1936]. So we propose the approach of Recursive Modeling, which means that the agent should not only build causal models for the environment, but also recursively build causal models on the early-built ones. The environment is where negentropy[Schrodinger, 1944] flows in.

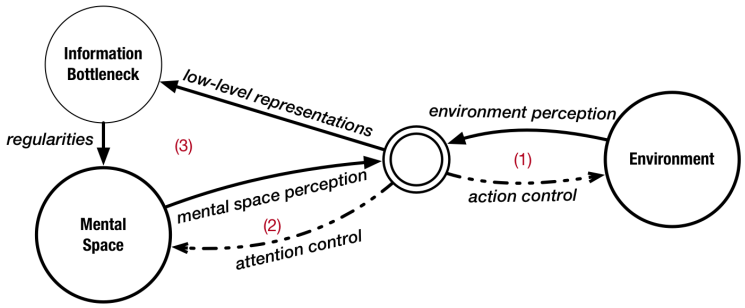


Figure 4: Schematic diagram of Recursive Modeling

As shown in the schematic diagram(Figure 4), the Recursive Modeling approach has two requirements. The first requirement is to build a mental space where models run. If we think of a model as a collection of regularities (or schemas[Piaget, 1929, Bartlett and Bartlett, 1932]), then the mental space is the collection of all regularities. Regularities are usually obtained from information bottlenecks, like the linguistic regularities found in the word vector space[Mnih and Kavukcuoglu, 2013], and the disentangled representations generated by generative models[Bengio et al., 2013, Larsen et al., 2015]. Existing low-level representations should be recursively distilled by the information bottleneck.

The second requirement of Recursive Modeling is to allow the agent to perceive and intervene the mental space, just as it does with the environment in the physical world. Perception and intervention are two necessities to build causal models at any time.

Among the models that have been built, the early built models are to simulate the relations between real entities in the environment, while the later ones are responsible for abstract thinking tasks, such as calculus in a symbolic system. We do not mean that there is a clear hierarchy between models. In fact, the notion of "model" is only a fictitious concept describing a set of closely related regularities, and many of those regularities are actually intertwined and shared, and reappear at different levels. Units in the cortex-network can also cluster into function regions, and regions can be organized in a hierarchy-like pattern. Different models can correspond to different regions in the cortex. However, this may be a future work and the article dose not involve this too much.

5.1 BHN and Recursive Modeling

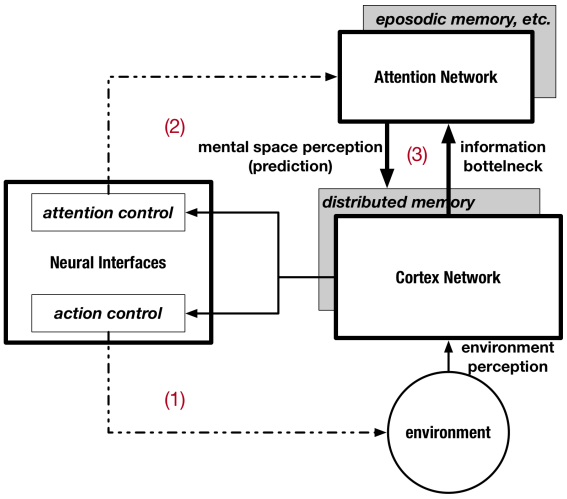


Figure 5: BHN adapted for Recursive Modeling

Figure 5 gives a schematic diagram of BHN adapted for Recursive Modeling, in which the three Loops marked in Figure 4 are also marked roughly at the corresponding positions.

BHN meets the two requirements of Recursive Modeling. Firstly, the attention-network can serve as an information/attention bottleneck[Felleman and Van, 1991], and the global attention(a) can be regarded as representations in the mental space. Secondly, it is possible for the agent to perceive the mental space by fusing bottom-up perceptions with top-down predictions, which will be detailed in the section 5.2.

We think that much of the intelligence of the human brain resides in its sophisticated architecture, and now our BHN model is oversimplified and lacks many essential functions, such as dopaminergic neurons for reward and prediction error learning[Hollerman and Schultz, 1998], a realization of the attention control interface, the hippocampus forming mental maps and episodic memories, etc. There is no doubt that we need more inspiration from the human brain to proceed with this work[Lake et al., 2017].

5.2 Working Memory

We think that the human brain works by continuously mixing real perceptions with imaginary predictions, and in extreme cases it is like "hearing one's thoughts spoken out aloud"[Schneider, 1939]. If z_t^i represents what is heard, then the expectation $e_t^i = \sum_{z_j^i \in Z^i} w_{tj}^i z_j^i$ can represent what the brain predicts to hear. By replacing z_t^i with e_t^i at some times, the agent can somewhat perceive the mental space just as it perceives the external environment.

z_t and e_t are homologous, and they can both be used as the output of a unit, so that the information flow within the net is actually a mixture of perceptions(z_t) and predictions(e_t). z_t is involuntary and volatile, but e_t is processed recurrently and remains somewhat locked inside the Loop (3)(marked both in Figure 4 and Figure 5), and so in this way, e_t can provide gain for z_t in a sense. We speculate that this mechanism corresponds to the brain's working memory, and its gain level determines whether the representations in the cortex will be suppressed or enhanced[Miller et al., 1991].

6 Experiment(2)

We follow the same basic setup of the simple model in section 4.2 to test the hypothesis of working memory by mixing z_t^i with e_{t-4}^i.

First, in the training phase, we feed z_t to $c_t^i = g_{ar}^i(*)$ for even time steps and feed e_{t-4} for odd time steps. A deconvolutional network reconstructing images from e to give a score is also trained in this phase.

Next, in the testing phase, taking a certain time step as the boundary, z_t is used before and e_{t-4} is used after. We judge the performance by how long the score of e keeps positive in the testing phase.

Figure 6 gives the result and it shows that the working memory effectively lasts for about 30 frames, much longer than the one frame which is what we adapt the system to in the earlier training phase.

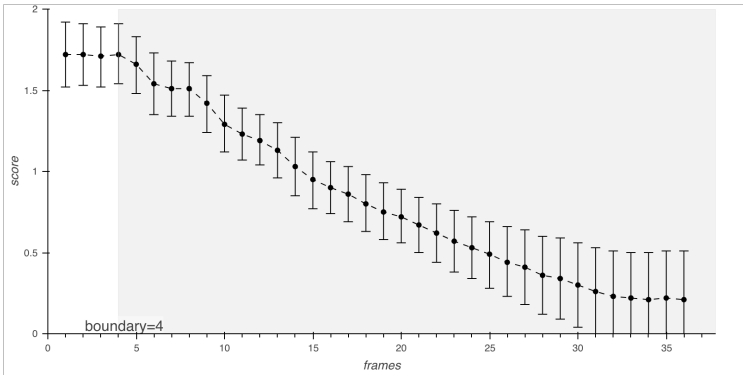


Figure 6: The score of e over time in the testing phase

7 Conclusions

In this article, we propose three hypothesises on the learning and working mechanism of the human brain. By formalizing these hypothesises, we get a computable objective, which is a sum of many objective functions. After that, we build and test a model(BHN), which couples several artificial neural networks together, to optimize the objective functions obtained. Finally, we propose the approach of Recursive Modeling and test a hypothesis on working memory.

Broader Impact

Our work has no direct ethical or societal implications.

References

[Atick, 1992] Atick, J. J. (1992). Could information theory provide an ecological theory of sensory processing? *Network: Computation in neural systems*, 3(2):213–251.

[Barlow et al., 1961] Barlow, H. B. et al. (1961). Possible principles underlying the transformation of sensory messages. *Sensory communication*, 1:217–234.

[Bartlett and Bartlett, 1932] Bartlett, F. C. and Bartlett, F. C. (1932). *Remembering: A study in experimental and social psychology*. Cambridge University Press.

[Bengio et al., 2013] Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828.

[Cho et al., 2014] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

[Felleman and Van, 1991] Felleman, D. J. and Van, D. E. (1991). Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, 1(1):1–47.

[Graves et al., 2014] Graves, A., Wayne, G., and Danihelka, I. (2014). Neural turing machines. *arXiv preprint arXiv:1410.5401*.

[Hadsell et al., 2006] Hadsell, R., Chopra, S., and LeCun, Y. (2006). Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE.

[Hollerman and Schultz, 1998] Hollerman, J. R. and Schultz, W. (1998). Dopamine neurons report an error in the temporal prediction of reward during learning. *Nature neuroscience*, 1(4):304–309.

[Hyvärinen and Oja, 2000] Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430.

[Lake et al., 2017] Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and brain sciences*, 40.

[Larsen et al., 2015] Larsen, A. B. L., Sønderby, S. K., Larochelle, H., and Winther, O. (2015). Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*.

[Linsker, 1990] Linsker, R. (1990). Perceptual neural organization: some approaches based on network models and information theory. *Annual review of Neuroscience*, 13(1):257–281.

[Miller et al., 1991] Miller, E. K., Li, L., and Desimone, R. (1991). A neural mechanism for working and recognition memory in inferior temporal cortex. *Science*, 254(5036):1377–1379.

[Mnih and Kavukcuoglu, 2013] Mnih, A. and Kavukcuoglu, K. (2013). Learning word embeddings efficiently with noise-contrastive estimation. In *Advances in neural information processing systems*, pages 2265–2273.

[Olshausen and Field, 1996] Olshausen, B. A. and Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609.

[Oord et al., 2018] Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

[Piaget, 1929] Piaget, J. (1929). The child's conception of the world (j. & a. tomlinson, trans.). *Savage, Maryland: Littlefield Adams*.

[Schneider, 1939] Schneider, K. (1939). *Psychischer befund und psychiatrische diagnose*. Thieme.

[Schrodinger, 1944] Schrodinger, E. (1944). What is life.

[Turing, 1936] Turing, A. M. (1936). On computable numbers, with an application to the entscheidungsproblem. *J. of Math*, 58(345-363):5.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

[von Helmholtz, 1925] von Helmholtz, H. (1925). *Physiological Optics*, volume 3. Optical Society of America.

[Wolpaw et al., 2000] Wolpaw, J. R., Birbaumer, N., Heetderks, W. J., McFarland, D. J., Peckham, P. H., Schalk, G., Donchin, E., Quatrano, L. A., Robinson, C. J., and Vaughan, T. M. (2000). Brain-computer interface technology: a review of the first international meeting. *IEEE transactions on rehabilitation engineering*, 8(2):164–173.

[Zeiler et al., 2011] Zeiler, M. D., Taylor, G. W., and Fergus, R. (2011). Adaptive deconvolutional networks for mid and high level feature learning. In *2011 International Conference on Computer Vision*, pages 2018–2025. IEEE.