Linking Sparse Coding Dictionaries for Representation Learning

Nicki Barari
Department of Computer Science
Drexel University
Philadelphia, USA
nb895@drexel.edu

Edward Kim
Department of Computer Science
Drexel University
Philadelphia, USA
ek826@drexel.edu

***Abstract*—Sparsity is a desirable property as our natural environment can be described by a small number of structural primitives. Strong evidence demonstrates that the brain's representation is both explicit and sparse, which makes it metabolically efficient by reducing the cost of code transmission. In current standardized machine learning practices, end-to-end classification pipelines are much more prevalent. For the brain, there is no single classification objective function optimized by back-propagation. Instead, the brain is highly modular and learns based on local information and learning rules.**

In our work, we seek to show that an unsupervised, biologically inspired sparse coding algorithm can create a sparse representation that achieves a classification accuracy on par with standard supervised learning algorithms. We leverage the concept of multi-modality to show that we can link the embedding space with multiple, heterogeneous modalities. Furthermore, we demonstrate a sparse coding model which controls the latent space and creates a sparse disentangled representation, while maintaining a high classification accuracy.

Keywords—Classification, machine learning, neuro-inspired artificial intelligence, representation learning.

I. Introduction

Representation learning is the process of encoding raw input data and transforming it into a vector embedding that can be used for subsequent tasks. In machine learning, a useful embedding would make the encoded information explicit, thus supporting less complex decoding downstream. Furthermore, a good representation extracts the underlying explanatory factors for a given input, which can be used as input to a supervised predictor [1].

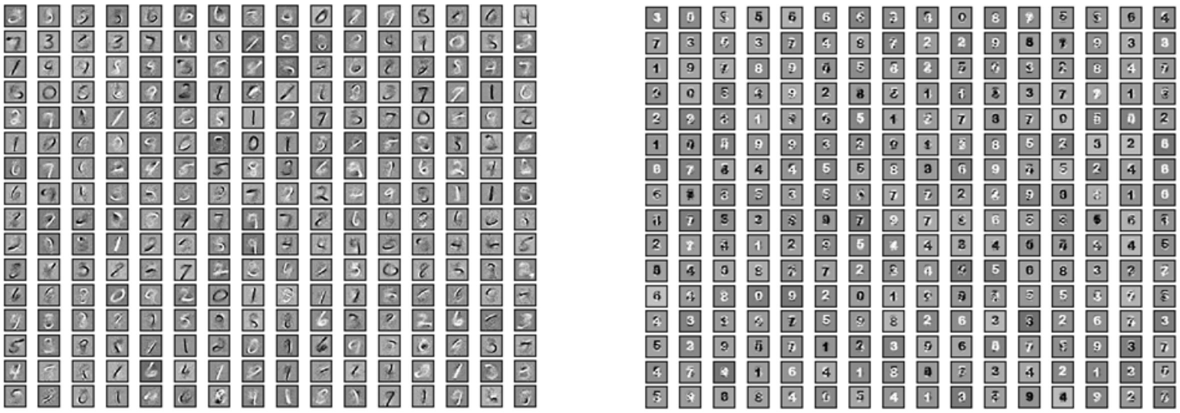
Likewise, in the brain, strong evidence demonstrates that the brain's representation (neural code) is both explicit and sparse [2] where neurons fire selectively to specific stimuli. Olshausen and Field [3] show that sparsity is a desirable property as our natural environment can be described by a small number of structural primitives. Biologically, the sparsity of neural codes are more metabolically efficient and reduce the cost of code transmission [4]. As an example, we can observe explicit information in biology when using two-photon calcium imaging in head-fixed *Drosophila melanogaster* i.e. observing the brain of a fruit fly. Brain activity is monitored as the fly walks on a ball in a virtual reality arena. When monitoring the activity the ellipsoid body, the orientation and angular path can be explicitly observed. More specifically, a neural population encodes the fly's azimuth and a simple population vector average (PVA) suffices to decode the fly's orientation [5]. Thus, we postulate that a neuromorphic representation learning algorithm should find a sparse, explicit encoding of an input stimulus.

However, in current standardized machine learning practices, end-to-end classification pipelines are much more prevalent. In these cases, representation and class discrimination are entangled as shown when opening the black box of deep learning through the lens of information theory [6]. Supervised learning seeks to optimize a narrow objective function, carelessly tuning all the parameters of the model towards maximizing this single objective. Thus, supervised learning creates representations that do well in *one* task, but fail to extend themselves to other tasks. This failure manifests itself in the lack of generalizability and catastrophic forgetting observed in supervised models. For the brain, there is no single classification objective function optimized by back-propagation [7]. Instead, the brain is highly modular and learns based on local information and learning rules [8].

In this paper, we seek to show that an unsupervised, biologically-inspired sparse coding algorithm can create a representation embedding that is explicit and sparse. Even with no supervision, we show that a sparse representation achieves a classification accuracy on par with standard supervised learning algorithms. Second, we show that we can link the embedding space with multiple, heterogeneous modalities without loss of classification accuracy. This linking process enables the representation to encode multiple heterogeneous signals in a single vector. Finally, we show that we can control the latent space using current-induced drivers to specific neurons, yielding a disentangled and interpretable activity response.

II. Background

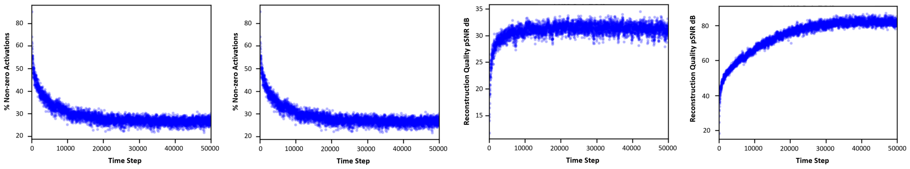
The advantages of sparsity in an artificial neural network are supported by the research and include information disentangling, efficient variable-size representation, efficient computing, and evidence that sparse representations are more linearly separable [9]. Indeed, within deep neural networks, the literature has shown that one can sparsify some networks up to 90% (only 10% non zeros) and still achieve the same



(a) Dictionary 1: MNIST digits

(b) Dictionary 2: Arial digits

Fig. 1: Linked Dictionaries



(a)

(b)

(c)

(d)

Fig. 2: (a) MNIST non-zero activations (b) Arial non-zero activations (c) MNIST reconstruction quality (d) Arial reconstruction quality

level of classification accuracy [10]. Algorithmically, we can demonstrate that we can recover the causes of image data through the use of a brain inspired algorithm called sparse coding.

Sparse coding has primarily been developed for computer vision, with successful applications including denoising, upsampling, compression and object detection. Sparse coding can be considered a self-supervised learning algorithm for signal reconstruction. Moreover, sparse and predictive coding explains many of the response properties of simple cells in the mammalian primary visual cortex, including both classical and non-classical phenomena [2], [11], [12].

Aside from sparsity, another crucial concept is the invariance of neurons to different modalities, i.e. neurons would fire from various sensory information such as image, sound or text [13]. Leveraging this concept, a multi-modal sparse coding model is presented in [14] that learns invariant, joint representations between two modalities, vision and language, by alternating between the optimization of a sparse signal representation and optimization of the dictionary elements. In this work, we leverage the concept of multi-modality to show that we can link the embedding space with multiple, heterogeneous modalities.

Finally, our reconstruction model creates a sparse disentangled representation. Disentangled representation means that independent latent units are being mapped to single data generative factors. In this work, we demonstrate a sparse coding model which controls the latent space and creates a sparse disentangled representation, while maintaining a high classification accuracy.

III. Methodology

Given an overcomplete basis, sparse coding algorithms seek to identify the minimal set of generators that most accurately reconstruct each input image. In neural terms, each dictionary element is a neuron (generator) that adds its associated feature vector to the reconstructed image with an amplitude equal to its activation. For any particular input image, the optimal sparse representation is given by the vector of sparse activation coefficients that minimizes both image reconstruction error and the number of non-zero coefficients. Formally, finding a sparse representation involves finding the minimum of the following cost function:

$$E = \frac{1}{2}||\mathbf{I} - \{\mathbf{\Phi} * \mathbf{a}\}||_2^2 + \lambda||\mathbf{a}||_1 \quad (1)$$

Where I is an input stimulus, which may be spatiotemporal and/or multi-modal and Φ is a dictionary of features, which may span a range of frames, camera views, and/or data modalities, all of which are combined linearly with the corresponding coefficients that constitute a sparse representation of the stimulus. The λ factor is a tradeoff parameter; larger values encourage greater sparsity (fewer non-zero coefficients) at the cost of greater reconstruction error.

Both the sparse coefficients and the dictionary of features can be determined by a variety of methods. Here, we solve for the sparse coefficients using a biologically plausible approach based on the Locally Competitive Algorithm (LCA) [15]. LCA finds a local minimum of the above cost function by introducing the dynamical variables (membrane potentials), such that the output of each neuron is given by a soft-threshold transfer function, with threshold λ, of the membrane potential $\mathbf{u}$:

$$\mathbf{a} = T_\lambda(\mathbf{u}) = H(\mathbf{u} - \lambda)\mathbf{u} \tag{2}$$

where H is the Heaviside (step) function. The cost function defined above is then minimized by taking the gradient of the cost function with respect to a and solving the resulting set of coupled differential equations for the membrane potentials,

$$\dot{\mathbf{u}} \propto -\frac{\partial E}{\partial \mathbf{a}} = -\mathbf{u} + \mathbf{\Phi}^T\{\mathbf{I} - \mathbf{\Phi}T_\lambda(\mathbf{u})\} + T_\lambda(\mathbf{u}). \tag{3}$$

A learning rule can be defined by taking the gradient of the cost function with respect to Φ, which leads to a local Hebbian learning rule that reduces reconstruction error given a sparse representation.

$$\Delta\mathbf{\Phi} \propto -\frac{\partial E}{\partial \mathbf{\Phi}} = \mathbf{a} \otimes \{\mathbf{I} - \mathbf{\Phi a}\} = \mathbf{a} \otimes \mathbf{R} \tag{4}$$

Dictionary learning will be performed via Stochastic Gradient Descent (SGD), using training data. As mathematically defined, sparse coding is a reconstruction algorithm that attempts to find a sparse representation and activation that can best match the input stimulus. However, in the case of our application, the energy function can be modified to accommodate multiple heterogeneous signals in the following way,

$$E = \frac{1}{2}(||(\mathbf{I_1}-\{\mathbf{\Phi_1}*\mathbf{a}\})||_2^2+||(\mathbf{I_2}-\{\mathbf{\Phi_2}*\mathbf{a}\})||_2^2)+\lambda||\mathbf{a}||_1 \tag{5}$$

Conceptually, one can see that there are now two distinct input stimuli, $\mathbf{I_1}$ and $\mathbf{I_2}$. Each input has an associated dictionary, yet share the same activation, $\mathbf{a}$. We define this methodology as “linking” the different dictionaries of heterogeneous inputs. We can generalize this method to cases where there are more than two modalities as follows:

$$E = \frac{1}{2}\sum_{i=1}^{n}||(\mathbf{I_i} - \{\mathbf{\Phi_i} * \mathbf{a}\})||_2^2 + \lambda||\mathbf{a}||_1 \tag{6}$$

IV. Results and Conclusion

In this experiment, we train our dictionaries of 256 neurons on MNIST dataset. In the linked dictionaries model, the input of the first dictionary is MNIST digits, and the input of the second dictionary is the Arial digits corresponding to the MNIST digits. As can be observed in Figure 1, the dictionaries are linked, i.e. they share the same activation vector. The model alternates between the optimization of activity vectors and optimization of the dictionary elements. The dictionaries are linked during the learning process, which means that similar neurons would fire in both dictionaries, given a particular digit. For example, if after the training process, we use just an MNIST digit and the MNIST dictionary to get an activation vector, we can use that activation to reconstruct the corresponding Arial digit (Figure 3).

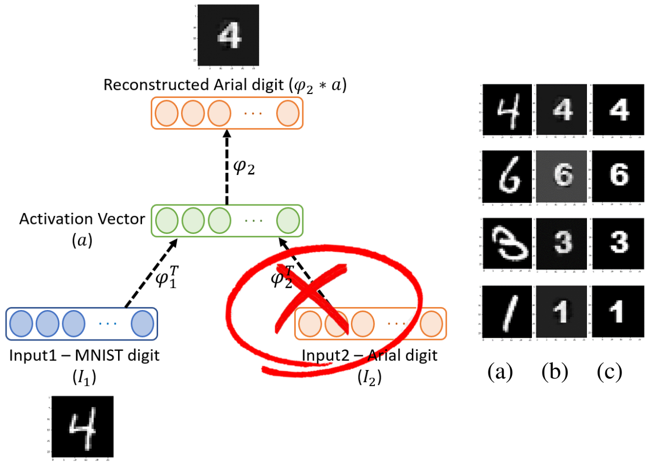


Fig. 3: (a) Input: MNIST digit (b) Reconstruction: $\Phi_2 * a$ (c) Expected Arial digit

To evaluate our model, we perform SVM and logistic regression (LR) on top of the linked-dictionary models, as well as on raw pixels, with accuracy as our metric. Our goal is not to improve performance on MNIST, but rather to demonstrate modifications to the framework that enable linking and controlling of the linked space, without sacrificing accuracy in classification. As can be observed in Table I, we can classify MNIST digits, using their sparse activation vectors and still achieve a high accuracy.

TABLE I: Accuracy results

	LR	SVM (linear)	SVM (non-linear)
Raw pixels	0.9167	0.8522	0.9250
Linked dictionaries	0.9633	0.9619	0.981
Ordinal-linked dictionaries	0.9541	0.9542	0.9738

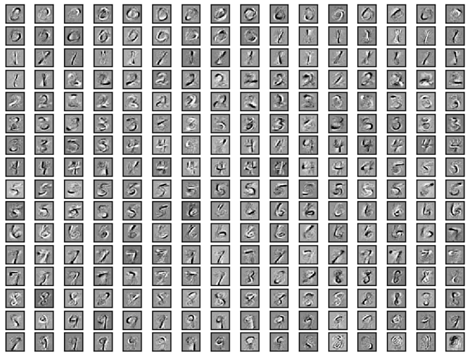
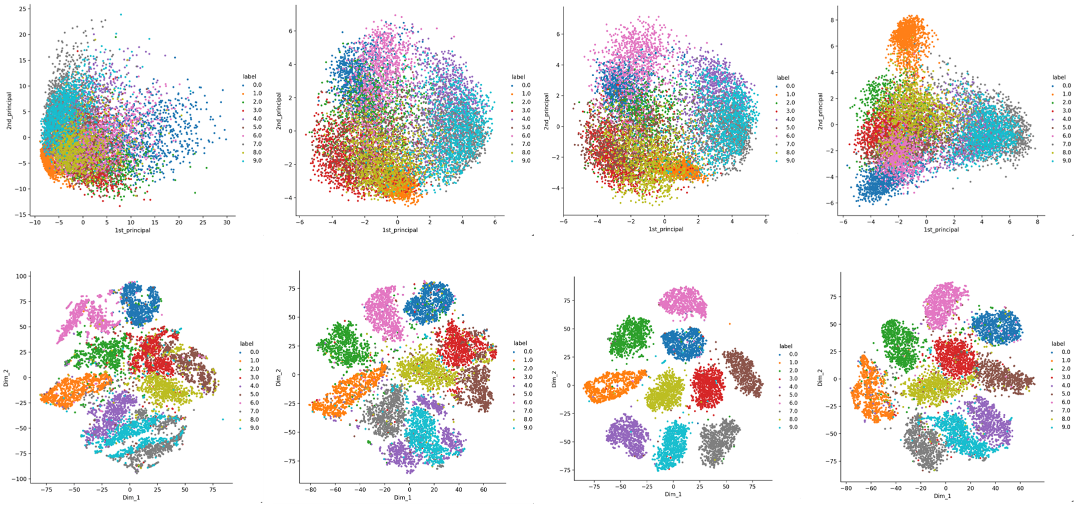


Fig. 4: Ordinal-linked dictionary where the latent space is class controlled through the use of current-induced neuronal drivers.

Aside from maintaining accuracy, another advantage of sparsity in neural networks is information disentanglement and controlling the latent space. As shown in Figure 1, our linked dictionaries are randomly generated, and we cannot find a specific order in dictionary neurons corresponding to digits. However, we can enforce a specific structure on dictionary



(a) (b) (c) (d)

Fig. 5: MNIST digits clusters resulted from performing PCA (first row) and t-SNE (second row) on (a) Raw pixels, (b) Single dictionary, (c) Linked dictionaries and, (d) Ordinal-linked dictionaries.

neurons, using ordinal-linked approach. Our goal here is to control the latent space by forcing certain neurons to be active during the dictionary update. We can choose which neurons control the generation of which digits, using a vector of 0s and 1s as our third input. Assuming our inputs are a 0 digit in MNIST and the corresponding 0 digit in Arial, our third input will be a vector of size 256, with 1s in the first 25 bins and 0s elsewhere. In that way, we are going to artificially activate the first 25 neurons out of 256 neurons. Thus, we can control the latent space and encourage certain blocks of neurons to be active for certain classes. Using this approach, neurons 0 to 24 will control the generation of 0s, neurons 25 to 49 will control the generation of 1s, and so on (Figure 4). Therefore, we are controlling the latent space by enforcing specific order on it, as well as creating a sparse disentangled representation, since the response of the dictionary corresponds directly to the class of the input image.

Finally, we believe that sparse representation encodes critical information, thus we perform PCA and t-SNE on the activation vectors as well as raw pixels (Figure 5). t-SNE (t-Distributed Stochastic Neighbor Embedding) is an unsupervised, non-linear technique used for dimensionality reduction [16]. Both PCA and t-SNE are used for visualizing and exploring high-dimensional data by transforming it into a 2-dimensional space. Performing t-SNE on the activation vectors should improve the result by making stronger clusters than the result of t-SNE on raw pixels. As can be observed in Figure 5, the result of t-SNE on the activation vectors is more separated clusters, and using linked dictionaries further improves the separation.

Acknowledgment

This material is based upon work partially supported by the National Science Foundation under Grant No. 1954364.

References

[1] Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[2] P. Foldiak, “Sparse coding in the primate cortex,” *The handbook of brain theory and neural networks*, 2003.

[3] B. A. Olshausen and D. J. Field, “Sparse coding with an overcomplete basis set: A strategy employed by v1?” *Vision research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[4] R. Baddeley, “Visual-perception-an efficient code in v1,” *Nature*, vol. 381, no. 6583, pp. 560–561, 1996.

[5] J. D. Seelig and V. Jayaraman, “Neural dynamics for landmark orientation and angular path integration,” *Nature*, vol. 521, no. 7551, pp. 186–191, 2015.

[6] R. Shwartz-Ziv and N. Tishby, “Opening the black box of deep neural networks via information,” *arXiv preprint arXiv:1703.00810*, 2017.

[7] S. Löwe, P. O’Connor, and B. S. Veeling, “Putting an end to end-to-end: Gradient-isolated learning of representations,” *arXiv preprint arXiv:1905.11786*, 2019.

[8] N. Caporale and Y. Dan, “Spike timing–dependent plasticity: a hebbian learning rule,” *Annu. Rev. Neurosci.*, vol. 31, pp. 25–46, 2008.

[9] X. Glorot, A. Bordes, and Y. Bengio, “Deep sparse rectifier neural networks,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 315–323.

[10] S. Han, H. Mao, and W. J. Dally, “Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding,” *arXiv preprint arXiv:1510.00149*, 2015.

[11] E. Kim, M. Daniali, J. Rego, and G. T. Kenyon, “The selectivity and competition of the mind’s eye in visual perception,” *arXiv preprint arXiv:2011.11167*, 2020.

[12] D. M. Paiton, C. G. Frye, S. Y. Lundquist, J. D. Bowen, R. Zarcone, and B. A. Olshausen, “Selectivity and robustness of sparse coding networks,” *Journal of vision*, vol. 20, no. 12, pp. 10–10, 2020.

[13] R. Q. Quiroga, L. Reddy, G. Kreiman, C. Koch, and I. Fried, “Invariant visual representation by single neurons in the human brain,” *Nature*, vol. 435, no. 7045, pp. 1102–1107, 2005.

[14] E. Kim, D. Hannan, and G. Kenyon, “Deep sparse coding for invariant multimodal halle berry neurons,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1111–1120.

[15] C. Rozell, D. Johnson, R. Baraniuk, and B. Olshausen, “Locally competitive algorithms for sparse approximation,” in *2007 IEEE International Conference on Image Processing*, vol. 4. IEEE, 2007, pp. IV–169.

[16] L. Van der Maaten and G. Hinton, “Visualizing data using t-sne.” *Journal of machine learning research*, vol. 9, no. 11, 2008.