DISENTANGLING MIXED SELECTIVITY NEURONS IN RESNET-34 BY IDENTIFYING RELEVANT NEURONS IN EARLY LAYERS

Yota Kawashima
Bernstein Center for Computational Neuroscience Berlin
yotakawashima@gmail.com

Supervisor: Lorenz Linhardt
Technical University of Berlin

Lab PI: Klaus-Robert Müller
Technical University of Berlin

November 17, 2024

ABSTRACT

Interpreting Convolutional Neural Networks (CNNs) is important for enhancing their reliability in practical applications. When neurons in the model encode multiple features, interpretation of the decision-making process becomes challenging. Dreyer et al. [2024] demonstrated that disentangling mixed features is possible by clustering relevant neurons in the immediate previous layer. However, this approach is limited due to its focus on the immediate previous layer that encodes high-level features, potentially overlooking useful low-level features encoded in earlier layers. Here, we investigated the potential of disentangling neurons by clustering relevant neurons in earlier layers. We disentangled neurons in ResNet-34 and compared the effectiveness of disentanglement across different layers used to identify relevant neurons. Our results indicate that examining the immediate previous layer disentangles neurons better than earlier layers do.

1 Introduction

As Deep Neural Networks have been used across various fields, there is an increasing demand for methods to interpret the decision-making process. Specifically, Convolutional Neural Networks (CNNs) can learn to perform complex visual tasks, some of which require understanding how the model reaches its conclusions. However, interpreting the process is challenging due to their reliance on inductive learning and the vast number of parameters. Several studies seek to bridge this gap by translating learned parameters to human-understandable concepts, giving rise to the field of mechanistic interpretability [Saphra and Wiegreffe, 2024].

One key aspect of mechanistic interpretability is identifying what each neuron in the model encodes. While some neurons exclusively activate for single input features, others similarly activate for multiple input features [Erhan et al., 2009, Nguyen et al., 2016] (e.g. neuron k in layer L in Figure 1a). These mixed selectivity neurons, also known as *polysemantic* neurons, reduce the interpretability of CNNs.

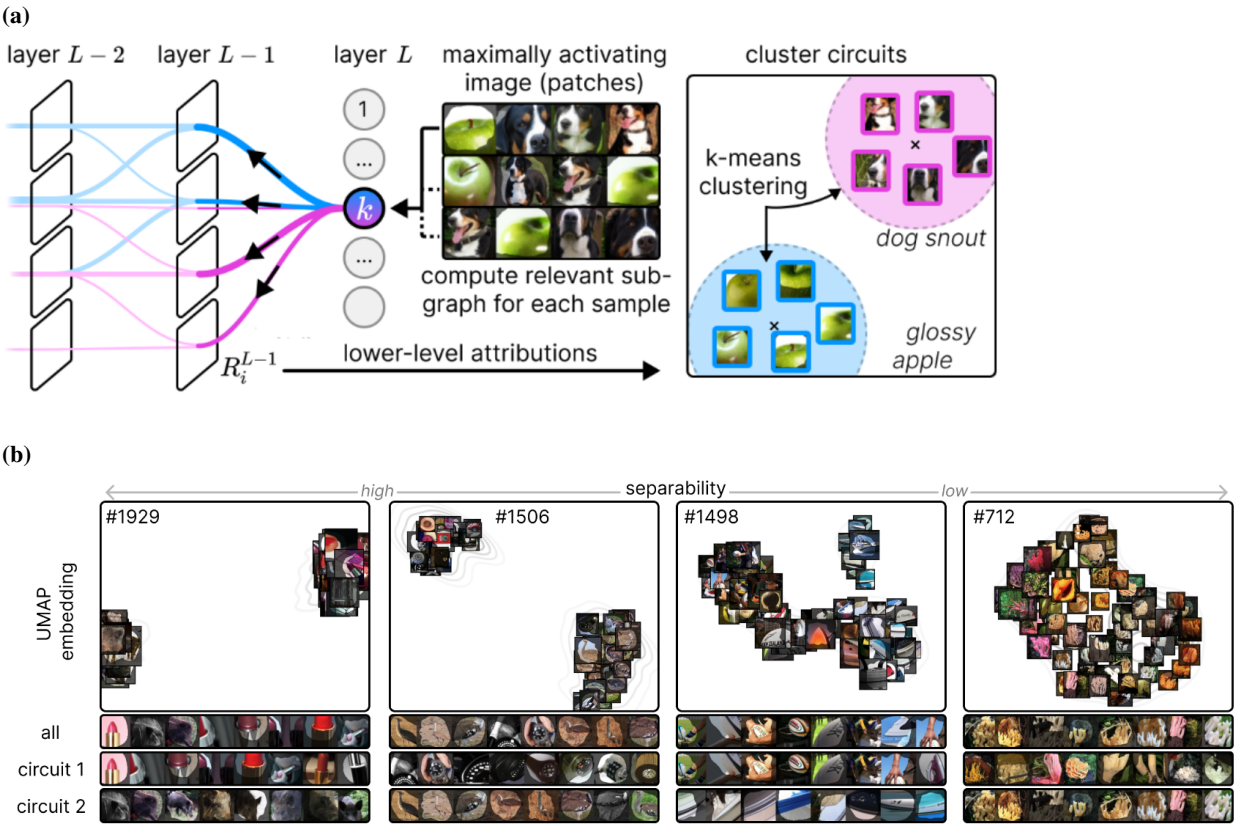


Figure 1: A schematic of PURE and its applications (Modified from Figure 1 and 3 in Dreyer et al. [2024]). **(a)** Disentangling a mixed selectivity neuron. Here, we disentangle a mixed selectivity neuron k in layer L that mixes two input features: dog snout and glossy apple. PURE computes a vector of relevance scores $\boldsymbol{R}^{L-1}$ in layer $L-1$ for each image among the top 50 maximally activating images. Then, it finds clusters of relevance score vectors. Finally, each cluster is linked to input features by visual interpretation. **(b)** Applying PURE to four mixed selectivity neurons in ResNet-50 model. Relevance score vectors for the top 50 maximally activating images are shown in the UMAP embedding space [McInnes et al., 2018]. PURE separates the set of images (all) into two clusters (circuits 1 and 2).

Dreyer et al. [2024] introduced a method called Purifying Representations (PURE) that disentangles mixed selectivity neurons through cluster analysis. To disentangle a mixed selectivity neuron k in layer L, they measured the relevance of each neuron in the immediate previous layer $L-1$ to the neuron k during the CNN's processing of a given input image (Figure 1a). This measure is known as relevance scores [Bach et al., 2015]. They applied a cluster analysis to relevance scores and showed that resulting clusters matched the features of mixed selectivity neurons (e.g. lipstick-boar neuron #1929 in Figure 1b).

However, PURE struggled to disentangle some mixed selectivity neurons. For example, PURE seemed less effective when neurons mixed object categories with similar low-level features (e.g. rugby ball and clothes iron for neuron #1498 in Figure 1b). Additionally, PURE did not disentangle neurons that encode the same object category with varying low-level features (e.g. corals with different colours and textures for neuron #712 in Figure 1b). Although these neurons encode single high-level concepts, achieving finer disentanglement within the same category could be beneficial in certain cases.

We hypothesized that clustering the relevance scores of neurons in early layers could improve the feature disentanglement. The original PURE examined relevance scores in the immediate previous layer, where neurons encode relatively high-level features. Low- and mid-level features in earlier layers may be useful for improving the disentanglement. Furthermore, examining relevance scores across different layers may allow for more precise control over the levels of disentanglement within the same category.

In this project, we aimed to test whether examining relevance scores in early layers improves the feature disentanglement for some neurons. We identified the selectivity of neurons in ResNet-34 [He et al., 2016] by using a subset of the ImageNet dataset [Russakovsky et al., 2015] and disentangled them. We evaluated the effectiveness of disentanglement as measured by the consistency of relevance scores with CLIP embeddings [Radford et al., 2021]. We found that examining the immediate previous layer disentangled the neurons better than earlier layers did.

2 Method

We investigated neurons in the penultimate layer of ResNet-34 (the last layer of block_3 in Table 1) pre-trained on ImageNet [Russakovsky et al., 2015]. We identified the selectivity of each neuron by identifying the top 50 maximally activating images from a subset of the ImageNet test dataset containing 10 image classes (Imagenette [Fastai]).

Table 1: ResNet-34 Architecture (Modified from Table 1 in [He et al., 2016]

Layer Name	Output Size	[width × height, channel]
conv1	112×112	7×7, 64, stride 2
block_0	56×56	3×3 max pool, stride 2
		$\begin{bmatrix} 3\times 3, 64 \\ 3\times 3, 64 \end{bmatrix} \times 3$
block_1	28×28	$\begin{bmatrix} 3\times 3, 128 \\ 3\times 3, 128 \end{bmatrix} \times 4$
block_2	14×14	$\begin{bmatrix} 3\times 3, 256 \\ 3\times 3, 256 \end{bmatrix} \times 6$
block_3	7×7	$\begin{bmatrix} 3\times 3, 512 \\ 3\times 3, 512 \end{bmatrix} \times 3$
fully connected	1×1	average pool, 1000-d fc, softmax

To disentangle mixed selectivity neurons, we applied k-means clustering to relevance vectors. We used Gradient×Activation as relevance scores. When we disentangle neuron k in layer L for a given input image, the relevance score of the neuron i in layer $L-1$, R_i^{L-1}, is:

$$R_i^{L-1} = A_i^{L-1} \frac{\partial A_k^L}{\partial A_i^{L-1}}$$

where A_i^{L-1} and A_k^L are neurons' activatation for the input image. We averaged the relevance scores over neurons within the same channel. We treated the averaged scores from channels as the scores from "neurons" (e.g. 256 "neurons" in the last layer of block_2). We refer to channels as neurons hereafter. Relevance vector $\boldsymbol{R}^{L-1}$ of layer $L-1$ is a set of relevance scores from all neurons in the layer $L-1$. We computed a relevance vector for each of the neuron k's top 50 maximally activating images and clustered them via k-means clustering ($k=2$) (Figure 1a). We performed the clustering analysis on relevance vectors from each of the following three layers: the last layer of block_0, block_1, and block_2. Note that the original PURE analysed the last layer of block_2.

We evaluated the disentanglement by measuring the consistency of relevance vectors with CLIP embeddings. The CLIP neural network maps input images into the CLIP embedding space, where images are clustered according to visual concepts [Radford et al., 2021]. We treated the CLIP embedding space as a reference. We mapped the top 50 maximally activating images to the CLIP embedding space and calculated the Euclidian distance between images. We also calculated the distance in the relevance vector space. Finally, we computed the Pearson correlation between the distance matrices (Figure 2). A higher correlation indicates a better disentanglement. For reference, we computed the correlation in the distance matrix between the CLIP embedding space and another DINOv2 embedding space. The DINOv2 neural network works similarly to CLIP [Oquab et al., 2023].

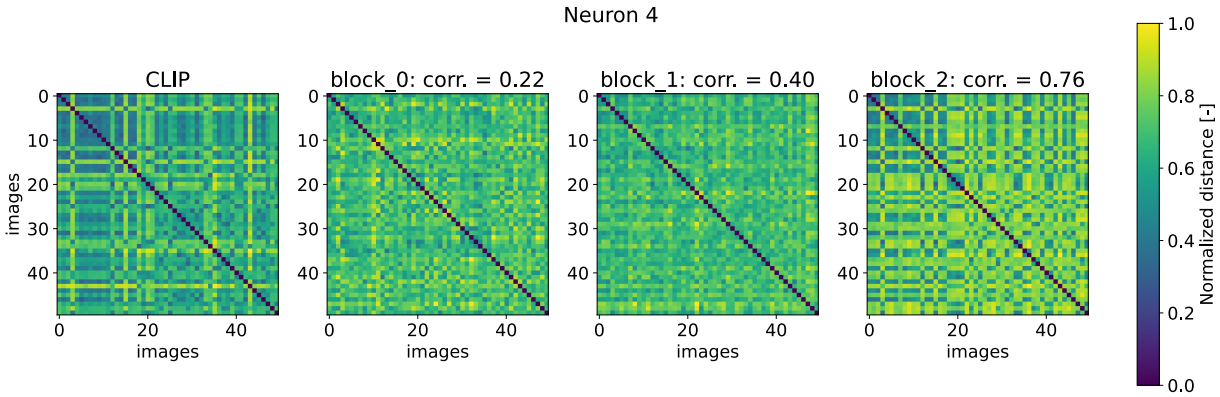
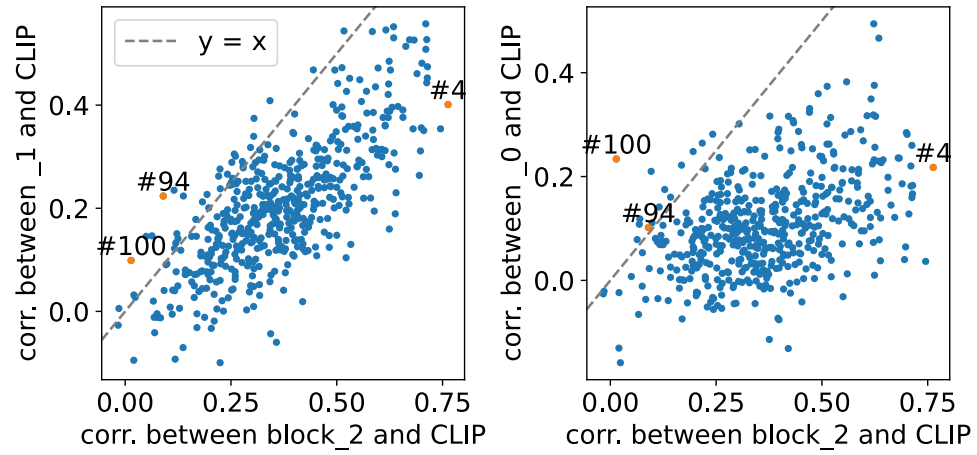


Figure 2: Distance matrix in the CLIP embedding space and the relevance vector space from different layers. Here, we disentangled neuron 4 in block_3. We computed the Euclidian distance between the top 50 maximally activating images in the CLIP embedding space and the distance between the relevance vectors for block_0, block_1, and block_2. If a layer better disentangles the neuron, its distance matrix is more highly correlated to the CLIP.

3 Result

We disentangled neurons in the block_3 of ResNet-34 by examining relevance vectors from block_0, block_1, and block_2. We measured the effectiveness of disentanglement by correlation in the distance matrix between the CLIP embedding space and the relevance vector space. Figure 3a shows the correlation from each neuron in block_3. Some neurons showed a higher correlation for block_0 or block_1 than block_2 (e.g. neuron #94 and #100). Overall, block_2 showed a higher correlation than the earlier layers (Figure 3b).

(a)



(b)

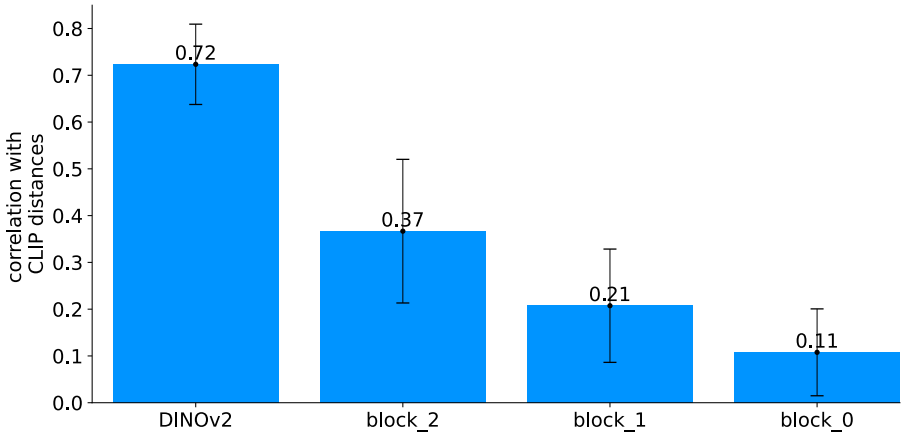
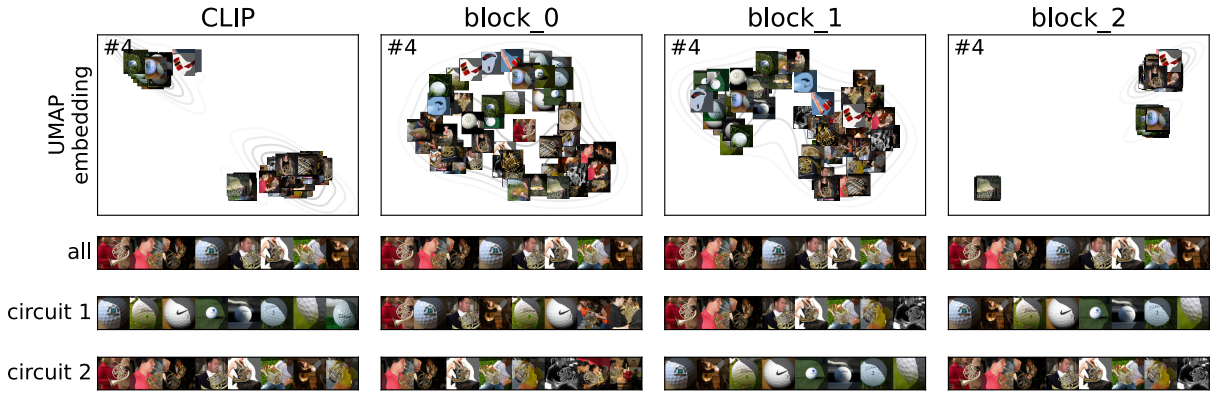


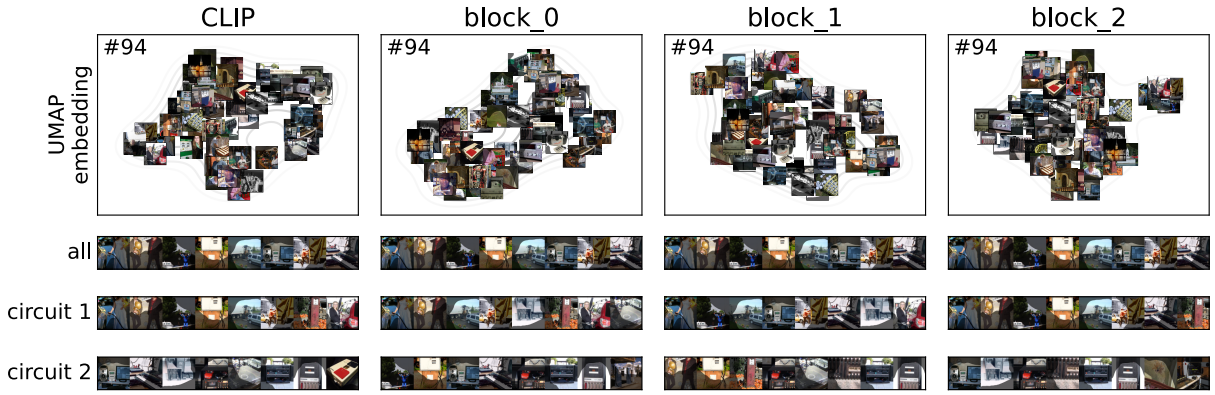
Figure 3: Overall better disentanglement by the immediate previous layer than the earlier layers. **(a)** Correlation between CLIP embeddings and relevance vectors from different layers. Dots correspond to single neurons in the penultimate layer (block_3). Dots above the dashed line are neurons with a higher correlation for the earlier layers (block_0 or block_1) than the immediate previous layer (block_2). **(b)** Mean and standard deviation across 512 "neurons" in block_3. The DINOv2 result is for reference.

We visually inspected the disentanglement for some neurons. We showed relevance vectors in the UMAP embedding space and the top 8 maximally activating images within each cluster. We inspected the neuron with the highest correlation in block_2 (neuron #4), with the largest difference in correlation between block_1 and block_2 (neuron #94), and between block_0 and block_2 (neuron #100) (Figure 4a - 4c). The visual inspection of these neurons revealed no clear qualitative improvements when the earlier layers were used.

(a)



(b)



(c)

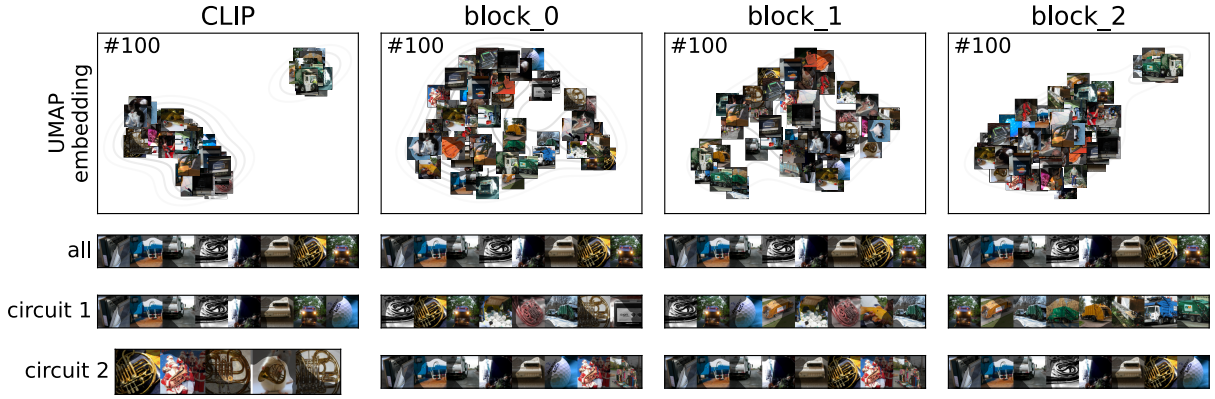


Figure 4: Clusters of maximally activating images for example neurons marked in Figure 3a. **(a)-(c)** CLIP embeddings and relevance vectors with the top 8 maximally activating images within each cluster.

4 Discussion

In this project, we tried to disentangle mixed selectivity neurons in ResNet-34 by examining the relevance of neurons in different layers to the mixed selectivity neuron. We measured the relevance by Gradient×Activation and clustered the scores. We found that the immediate previous layer of mixed selectivity neurons disentangled the neurons better than earlier layers did.

The poor feature disentanglement by earlier layers could be due to misidentifying the selectivity of neurons. We identified maximally activating images from a subset of ImageNet containing only 10 image categories. Neurons are highly likely to encode features present in the other image categories and the selectivity was likely to be misidentified. Using the full ImageNet dataset would improve identifying the selectivity, which might lead to better disentanglement by early layers.

Alternative evaluation measures for the effectiveness of disentanglement might be useful. We used one of the evaluation measures in the Dreyer et al. [2024] to select neurons for the visual inspection in Figure 4a-4c. Other evaluation measures might be helpful in finding neurons that were better disentangled.

5 Code Availability

Code is available on the following GitHub repository: `https://github.com/yotaKawashima/modified_PURE`.

References

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On Pixel-Wise explanations for Non-Linear classifier decisions by Layer-Wise relevance propagation. *PLoS One*, 10(7):e0130140, July 2015.

Maximilian Dreyer, Erblina Purelku, Johanna Vielhaben, Wojciech Samek, and Sebastian Lapuschkin. PURE: Turning polysemantic neurons into pure features by identifying relevant circuits. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, April 2024.

D. Erhan, Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. 2009. URL `https://api.semanticscholar.org/CorpusID:15127402`.

Fastai. Fastai/imagenette: A smaller subset of 10 easily classified classes from imagenet, and a little more french. URL `https://github.com/fastai/imagenette`.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE, June 2016.

Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, February 2018.

Anh Totti Nguyen, J Yosinski, and J Clune. Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks. *arXiv.org*, 2016.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *arXiv [cs.CV]*, April 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, December 2015.

Naomi Saphra and Sarah Wiegreffe. Mechanistic? *arXiv [cs.AI]*, October 2024.