A machine learning approach to predict brain response dissimilarity to pairs of visual stimuli (GitHub)

Lindsay Hexter (A12871874), Mariona Puente Quera (PAO054327), Idan Suliman (212229132)

Course: Theories on Nerve Networks and Machine Learning (852750401)

Bar Ilan University

Introduction

Learning Problem

Existing research has focused on predicting a stimulus from brain activity (decoding) and predicting brain activity from a stimulus (encoding) (Chen et al., 2014). In the last ten years, deep neural networks (DNNs) have become popular because of their increasing performance classifying objects and predicting brain activity in the visual system (Muttenthaler & Hebart, 2021). Notably, obtaining image feature vectors from pretrained Machine Learning (ML) models and calculating dissimilarity between the resulting pair of embeddings is moderately correlated with that calculated between fMRI responses to that same image pair, suggesting similar representations in those models and the brain (Muttenthaler & Hebart, 2021).

However, to the best of our knowledge, the field has not yet approached the task of directly predicting how differently our brain encodes two visual stimuli. Therefore, here we address the novel task of predicting dissimilarity between fMRI responses to pairs of natural scene images.

We used the Natural Scenes Dataset, a high-resolution fMRI dataset collected from 8 healthy adult subjects as they viewed 10,000 natural scenes during a continuous recognition task (NSD, 2025). We selected 1 - Pearson correlation to compute dissimilarity between fMRI recordings, calculating over each pair of fMRI responses to construct a Representational Dissimilarity Matrix (RDM).

We explored a variety of ML architectures, with the ultimate goal being to input two images and output the dissimilarity value of their corresponding fMRI responses as found in the RDM.

Previous Solutions

Our first attempts yielded poor results. Firstly, we trained a Siamese Network with two Convolutional Neural Network layers per branch. We used 500 images that were first converted to grayscale, resized to 128×128, and normalized. The fMRI data consisted only of the V1 region of subject 1. The model took two images as input and processed them through two separate Siamese branches, each consisting of two convolutional layers, both followed by max-pooling layers. These produced two feature vectors that were concatenated and passed through a dense layer before outputting the result. We used Mean Squared Error (MSE) as the loss function and a linear activation function. Our resulting R^2 value between the predicted and true values was -0.018, with a mean of 1.023 and a standard deviation of 0.000 (i.e. nearly predicting the mean).

In our second approach, we used a pretrained model (details in next section) to obtain embeddings for 500 images, rather than training from scratch. We retained all Region of Interest (ROI) data of subject 1 rather than selecting only V1, to input more fMRI information. Each pair of embeddings was concatenated and input into a trainable Multi-Layer Perceptron (MLP) with 0, 1, 2, or 3 hidden layers. Again, we used MSE loss and a linear activation function. The resulting R^2 values ranged from -15 to 0 (i.e. predicting the mean or worse).

As the results were disappointing and adding layers to the MLP did not improve performance, we decided to focus instead on a basic problem using simple ML architecture to better understand the feasibility of our novel task.

Methodology

Data preparation

FMRI data preparation – we used preprocessed, organized data from the Algonauts 2023 challenge (Algonauts Project, 2023; updated for 2025). We applied ROI mappings to extract voxel activity specific to predefined areas associated with visual processing (e.g. V1v, V2v, etc.). The selected ROI data from both hemispheres was concatenated to yield a unified, standardized feature vector.

RDM – pairwise dissimilarities between these fMRI feature vectors were calculated using 1 - Pearson correlation, ranging from 0 (most similar) to 2 (most dissimilar). We then selected a threshold of 1 for binary classification of 'similar' and 'dissimilar' pairs. Figure 1 shows the two RDMs constructed from a subset of 100 images and corresponding fMRI responses (100 choose 2 total pairs).

Figure 1

Representational Dissimilarity Matrices for continuous and binary values
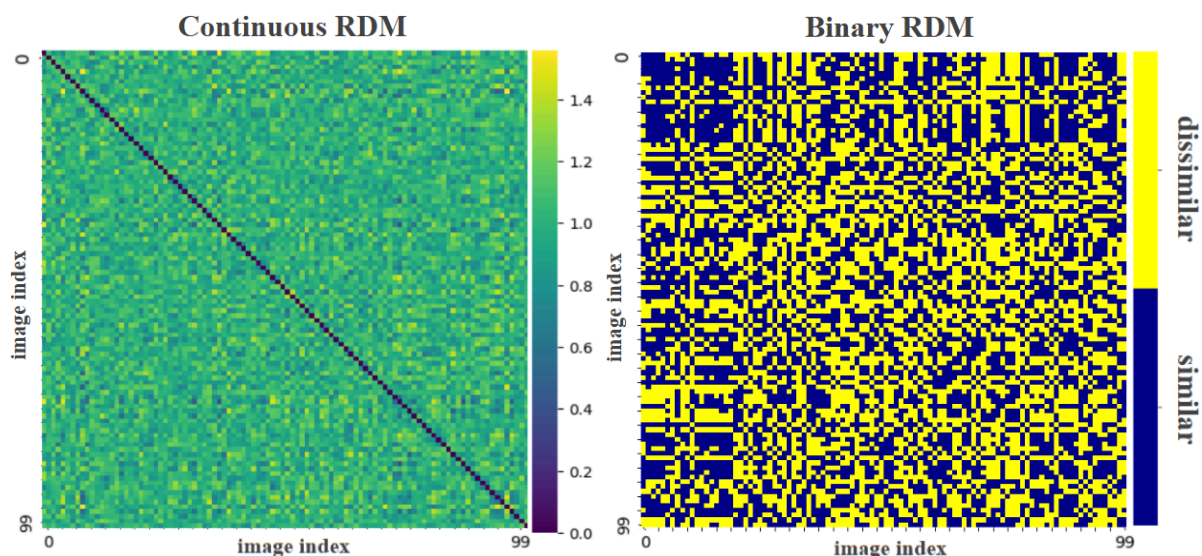
CLIP-ViT (Vision Transformer) (Radford et al., 2021) – as per the second model (MLP), we used this pretrained model to obtain embeddings as model input, rather than preprocessed images. CLIP-ViT is well known and maps images and text to a shared embedding space.

Preparing data subsets

As said, after observing generally poor performance and in many cases improved metrics only on training but not test sets, we shifted to a simpler task and focused on data preselection to better understand our dataset and our problem space. Namely, to ensure training and test datasets shared similar features, we selected the data based on two paradigms:

Extreme RDM values – for each of the 500 images, we identified the 50 most similar and dissimilar image pairs based on dissimilarity metrics. This filtering resulted in a distribution of two clusters with higher variance, yielding a dataset shaped accordingly for binary classification. Figure 2 shows the two most extreme pairs. The lowest RDM values indicate the highest dissimilarity between the fMRI responses to image pairs, i.e. highest similarity. In contrast, the highest RDM values reflect the most negative correlations, suggesting the strongest inverse relationship, i.e. highest dissimilarity.

Figure 2

Lowest and highest RDM values

Lowest RDM value = 0.356	Highest RDM value = 1.524

Red / Blue colors – to control for the effect of color selection on fMRI responses, we classified images based on color hue (red, blue, green), using a threshold of 0.7 to determine dominant color based on pixel count. If no color reached the threshold, we labeled those images as unclassified. Even out of 2000 images, 1195 remained unclassified. To ensure enough data for testing and training, we selected the two colors with the most images: blue (239) and red (498). Red images were clipped to 239 to ensure an even distribution. In Figure 3 we can see examples of all three color classes.

Figure 3

From left to right – red, blue and green examples of images classified as such
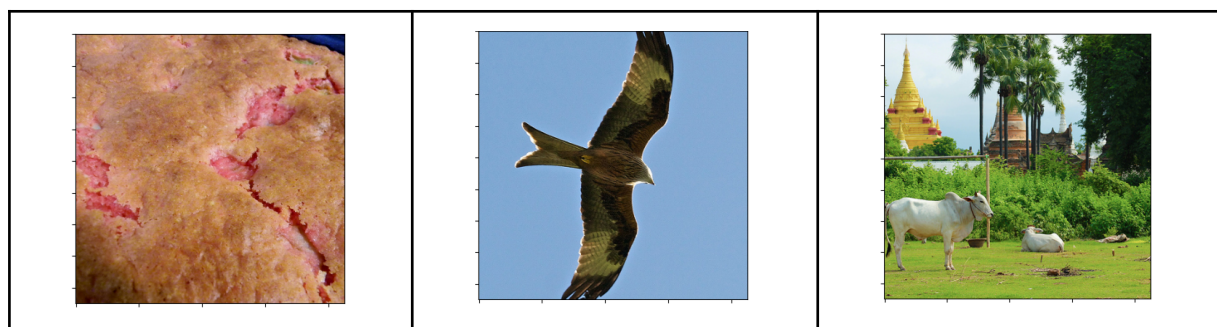

ML models

Support Vector Machine (SVM) – we trained an SVM to classify 'similar' or 'dissimilar' pairs, using subject 1 data (all ROIs) filtered to extremes or colors as discussed. Input to the model was a pair of concatenated embeddings and output was a binary label (0 or 1). We used Support Vector Classifier from the scikit-learn library, which implements regularized hinge loss. After hyperparameter tuning we selected a polynomial kernel with a degree of 2 and default C value of 1. We chose accuracy, precision, recall, and F1 score as evaluation metrics.

Contrastive siamese network – we trained a contrastive learning model using 1500 image embeddings and fMRI data of subject 1 over all ROIs. As shown in Figure 4, the model inputs pairs of image embeddings into two identical branches—MLPs with shared weights—to extract features. These features are eventually concatenated and passed through a similarity head to output the final dissimilarity score. The network was trained using MSE loss between predicted and true dissimilarity values. We used 5-fold cross-validation, a dropout rate of 0.5, 15 epochs, and R^2 for evaluation. To verify understanding of our loss and evaluation metrics, we calculated the chance levels of MSE and

R^2, i.e. between the labels of our set and the same labels shuffled (MSE varied per subset, $R^2 = -1$), as well as the performance of simply predicting the mean (MSE varied per subset, $R^2 = 0$).

Figure 4
Contrastive siamese network
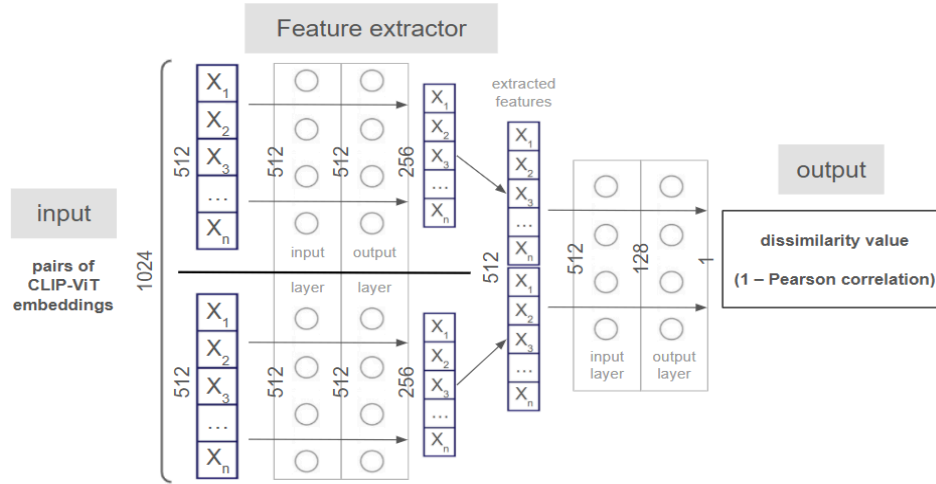
Results

SVM – As shown in Table 1, we saw better results with extreme values over color partitioning. In Figure 5 we can see the corresponding confusion matrices for both experiments.
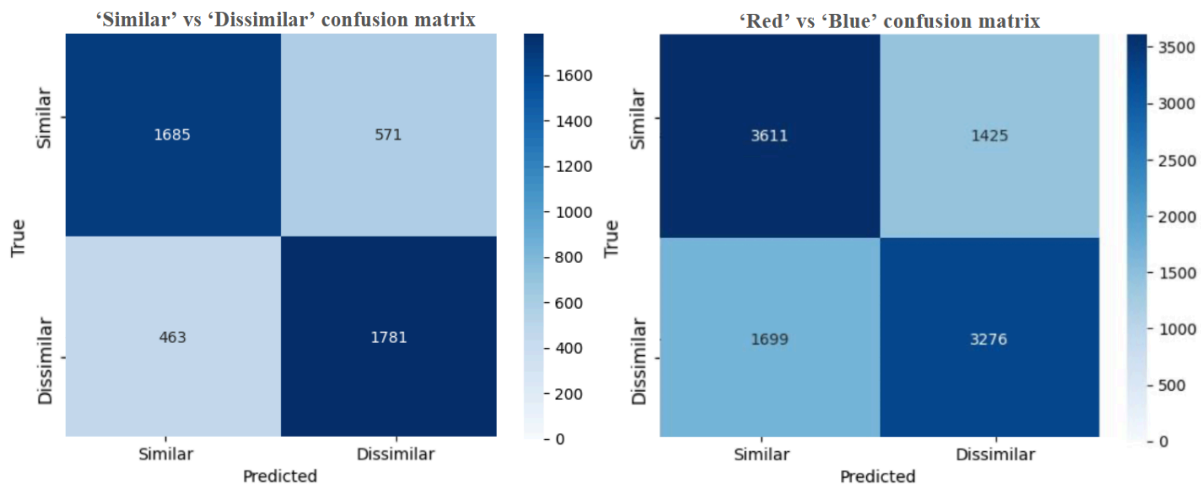
Table 1
SVM Evaluation Metrics

	Accuracy	Precision	Recall	F1
Extreme values	0.770	0.757	0.794	0.775
Colors (R/B)	0.680	0.680	0.717	0.698

Figure 5
SVM resulting confusion matrices for 'similar' vs. 'dissimilar' and 'red' vs. 'blue'

Contrastive Siamese Network – In Table 2, we see the resulting R^2 and MSE loss averaged across folds. We can see metrics across all folds in Figure 6 as well, which also shows reference lines for chance (shuffle) and predicting the mean, for result comparison. Figure 7 shows a scatter plot of

Table 2
Contrastive network - average evaluation metrics across folds

Avg across folds	$R^2 \pm$ **StdDev**	**MSE $\pm$ StdDev**
Testing set	0.279 ± 0.023	0.015 ± 0.000
Training set	0.757 ± 0.010	0.005 ± 0.000

predicted versus true values, where a diagonal line would denote perfect prediction; we can see of course that the training plot is much more aligned with true values compared to the testing set, and generally we reached moderate performance when training over all images compared to other data partitioning schemes (results not shown). Therefore the contrastive model benefitted from as much data input as possible, compared to any specific data preprocessing (extreme values, red/blue colors).

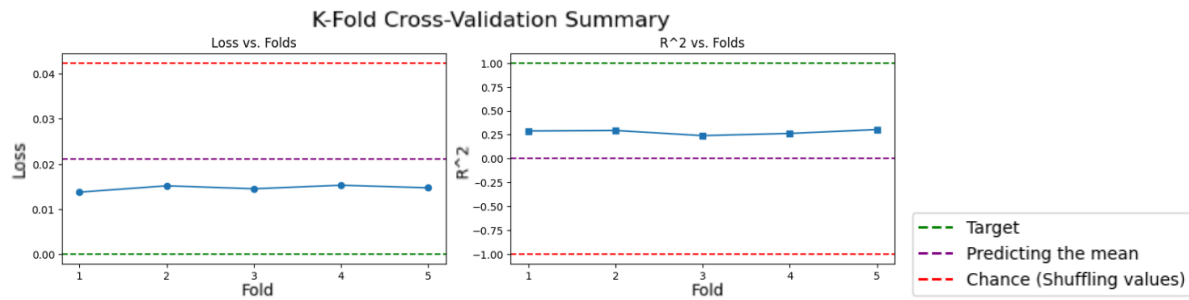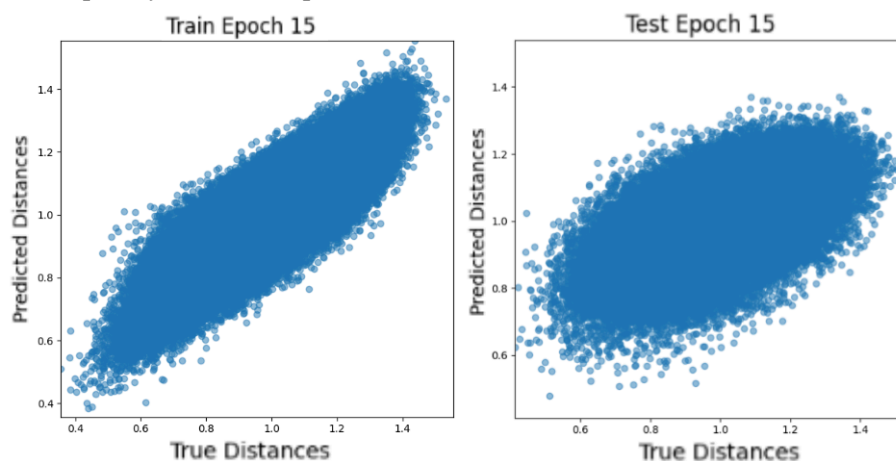Figure 6

R² and MSE loss across folds

Figure 7

Scatter plot of the true vs. predicted similarities in the train and test set

In conclusion, our models yielded better results than chance or than just predicting the mean. Intuitively the SVM performed better on the dissimilar vs similar task compared to red vs blue colors, since in the first case as expected the margin between the dissimilarity metrics was larger.

We also tried a few other iterations not shown, including: replacing 1-Pearson correlation with Euclidean distance, balancing the dissimilarity values to distribute evenly extreme and mid-range samples, and selecting certain ROIs. The models did not perform as well in these experiments – as expected: a more divided binary class yields better SVM performance, and overall, providing more input to the siamese network allows it to learn a more nuanced feature space, since it performed better over all data rather than over data partitions (extreme values or red vs. blue).

These findings demonstrate the feasibility of learning brain-based similarity from image features. If future versions of our model achieve high performance (e.g. using other pretrained models, partitioning the data in new ways, selecting particular ROIs, comparing across subjects), we could help uncover which image features are most aligned with human brain responses, ultimately contributing to a deeper understanding of visual perception.

References

Algonauts Project. (2023). *The Algonauts Project 2023 Challenge*. Retrieved from
https://algonautsproject.com/

Allen, E. J., St-Yves, G., Wu, Y., Breedlove, J. L., Prince, J. S., Dowdle, L. T., Nau, M., Caron, B.,
Pestilli, F., Charest, I., Hutchinson, J. B., & Kay, K. N. (2022). *A massive 7T fMRI dataset to bridge
cognitive neuroscience and artificial intelligence*. Nature Neuroscience, 25(1), 116–126.
https://doi.org/10.1038/s41593-021-00962-6

Chen, M., Han, J., Hu, X. *et al.* Survey of encoding and decoding of visual stimulus via FMRI: an
image analysis perspective. *Brain Imaging and Behavior* 8, 7–23 (2014).
https://doi.org/10.1007/s11682-013-9238-z

Muttenthaler, L., & Hebart, M. N. (2021). THINGSvision: A Python toolbox for streamlining the
extraction of activations from deep neural networks. *Frontiers in Neuroinformatics, 15*, 679838.
https://doi.org/10.3389/fninf.2021.679838

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A.,
Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). *Learning transferable visual models from
natural language supervision*. In *Proceedings of the 38th International Conference on Machine
Learning (ICML)*.