Improved prediction of behavioral and neural similarity spaces using pruned DNNs

Homa Priya Tarigopula
Center for Mind/Brain Sciences - CIMeC
University of Trento, Italy
homapriya.tarigopula@studenti.unitn.it

Scott Laurence Fairhall
Center for Mind/Brain Sciences - CIMeC
University of Trento, Italy
scott.fairhall@unitn.it

Anna Bavaresco
Data Science Program
University of Trento, Italy
anna.bavaresco@studenti.unitn.it

Nhut Truong
Center for Mind/Brain Sciences - CIMeC
University of Trento, Italy
leminhnhut.truong@unitn.it

Uri Hasson
Center for Mind/Brain Sciences - CIMeC
University of Trento, Italy
uri.hasson@unitn.it
[*]

Abstract

Deep Neural Networks (DNNs) have become an important tool for modeling brain and behaviour. One key area of interest has been to apply these networks to model human similarity judgements. Several previous works have used the embeddings from the penultimate layer of vision DNNs and showed that a reweighting of these features improves the fit between human similarity judgments and DNNs. These studies underline the idea that these embeddings form a good basis set but lack the correct level of salience. Here we re-examined the grounds for this idea and on the contrary, we hypothesized that these embeddings, beyond forming a good basis set, also have the correct level of salience to account for similarity judgments. It is just that the huge dimensional embedding needs to be pruned to select those features relevant for the considered domain for which a similarity space is modeled. In Study 1 we supervised DNN pruning based on a subset of human similarity judgments. We found that pruning: *i*) improved out-of-sample prediction of human similarity judgments from DNN embeddings, *ii*) produced better alignment with WordNet hierarchy, and *iii*) retained much higher classification accuracy than reweighting. Study 2 showed that pruning by neurobiological data is highly effective in improving out-of-sample prediction of brain-derived representational dissimilarity matrices from DNN embeddings, at times fleshing out isomorphisms not otherwise observable. Using pruned DNNs, image-level heatmaps can be produced to identify image sections whose features load on dimensions coded by a brain area. Pruning supervised by human brain/behavior therefore effectively identifies alignable dimensions of knowledge between DNNs and humans and constitutes an effective method for understanding the organization of knowledge in neural networks.

[*] Correspondence: uri.hasson@unitn.it

Preprint. Under review.

1 Introduction

1.1 Deep networks for vision as models of human similarity spaces

Deep Neural Networks for computer vision are now routinely used as predictive models of human brain and behavior (e.g., Cichy and Kaiser, 2019). A key question is whether these networks develop knowledge that is organized according to latent dimensions similar to those that structure human knowledge. For a given set of images this issue can be addressed by *i*) soliciting pairwise similarity judgments from humans; *ii*) computing cosine similarity for the same set from DNN embeddings, and *iii*) evaluating the correspondence between the two similarity matrices. The magnitude of this correspondence is reported as an R^2 value and frequently referred to as second-order-isomorphism (*2OI*, Kriegeskorte et al., 2008). Practically, it reflects the network's capacity to predict human similarity judgments. There is no need to use explicit similarity judgements; these can be replaced by any procedure that outputs a similarity space from human behavior or neural activity.

There is substantial heterogeneity in 2OI R^2 values reported when relating DNN embeddings to human similarity judgments. Peterson et al. (2018) analyzed embeddings from the penultimate layer of VGG-19 (Simonyan and Zisserman, 2014) and reported 2OI for six image-sets, each set drawn from a different semantic category such as ANIMALS or FRUITS (a VARIOUS category combined images across such categories). Reported values were in the range of $R^2 = 0.2 - 0.6$. King et al. (2019) reported similar values for object and scene categories with a maximal Spearman's Rho value of 0.56 (R^2 not reported). Groen et al. (2018), studying more complex scenes, reported 2OI values of $R^2 = 0.07,\ (r = 0.26)$. A large-scale analysis based on estimating psychological embeddings from 50,000 stimuli (Roads and Love, 2021) reported Spearman Rho values between 0.02 and 0.36, across 12 DNN architectures.

However, Peterson et al. were able to increase 2OI's R^2 by reweighting the activation (output) of each node in a DNN's penultimate layer. This is not an architectural change *per se*, as no network parameter is changed. Instead, this process can be considered as modifying each feature's salience. The authors implemented reweighting by maximizing the fit between the human and DNN's similarity matrices through linear regression. Practically, the fit between human similarity judgment (s) and DNN-pairwise-similarity for any two images k, j was defined as $s_{kj} = \sum_{i=1}^{4096} w_i f_{ki} f_{ji}$. Here, f_{k_i} and f_{j_i} are the values of feature i (of $n = 4096$) for image k and image j. The estimated pair-wise similarity s_{kj} is just the sum of these products across all $n = 4096$ features. Regression therefore corresponds to learning a 4096-weight-set w, that alters the importance of each feature so that the sum of the products becomes a better estimator of human similarity judgments.

Several subsequent studies have extended this approach. Other forms of linear-transforms of the embedding matrix from images have been studied (Attarian et al., 2020). In computational linguistics, reweighting of word-embedding vectors improved prediction of human similarity judgments (Richie and Bhatia, 2020). Non-linear reweighting was also shown to be effective in modeling similarity judgments for images (Sanders and Nosofsky, 2020).

The concept of reweighting shares similarities with attention modulation in DNNs. For instance, in Lindsay and Miller (2018), units' activity in feature maps was adjusted using weights proportional to their preference for an object category, boosting performance in detecting that category in challenging scenarios. Similarly, Luo et al. (2021) presented a learnable attention layer to reweight incoming activations. They found that a strong attention led to heightened hit rates and false alarms, whereas moderate increases yielded improved hit rates with only slight rises in false alarms.

The effectiveness of reweighting for prediction of human similarity speaks to the strengths and limitations of vision-DNNs. It suggests that a DNN's penultimate layer already forms a useful basis-set for predicting human similarity in that it captures visual features that structure human knowledge and are used when humans compare objects. Nonetheless, when modeling human similarity judgments, recalibration is required in order to assign features their correct levels of salience (see Richie and Bhatia, 2020, for similar argument in context of word embeddings).

1.2 Framework and Aims

The departure point of the current work is that the argument for reweighting may reflect an under-appreciation of the capacity of DNNs to predict human similarity judgments. Reweighting operationalizes the assumption, summarized above, that DNNs learn relevant features but assign them different levels of salience with respect to humans. A different possibility, which we probe here, is that DNNs do in fact acquire the relevant features at appropriate levels of salience. It is just that in any particular evaluation context where human similarity-space is predicted, the contribution of relevant features is diluted by irrelevant ones. Specifically, taking the *entire* penultimate layer of a DNN as the relevant basis set effectively combines two representational sub-spaces: those relevant for human similarity judgments and those less relevant. While re-weighting can be considered as a way to de-mix these spaces, it comes with two costs. First, reweighting is applied via linear or non-linear transforms of node activation values in the DNN's penultimate layer. This does not directly translate to a change in network architecture as no network weights are changed but instead a positive or negative multiplier is applied to the dot product of a feature's values for two images. Explainability is further reduced by the fact that interpreting regression coefficients for reweighting is non-trivial even in the case of linear regression, and to our knowledge has not been attempted. Second, reweighting strongly reduces a network's ability to classify, which significantly diminishes the ability to understand the relation between those features important for classification and those important for predicting similarity. To confirm that reweighting abolishes classification, we successfully reproduced the analyses and results of Peterson et al. (2018), and then evaluated VGG-19's accuracy after the penultimate layer was reweighted. Top-1/top-5 accuracy dropped from $\{72.7; 91.0\}$ to $\{9.4; 23.92\}$ respectively.

The alternative position that we propose and evaluate is that when modeling similarity spaces produced by human similarity judgments (or brain activity) there exists a non-reweighted subset of features that is most informative. The idea then is to produce a pruned DNN network that better models human similarity than the original non-pruned network. The intuition behind this idea borrows directly from neurobiology. Neuroscientists often quantify the 2OI between multivariate brain-activity and human similarity judgments, and this has strongly advanced knowledge of brain areas sensitive to particular categories. When doing so, an initial step involves feature selection: selecting a limited set of brain-features (e.g., fMRI voxels or EEG sensors) that are expected, a-priori, to contain the relevant information. For example, similarity judgments for scenes, objects, or faces would naturally be predicted using multivariate activation patterns sampled from different brain areas. That is, relatively limited brain areas will be selected, depending on the semantic categories studied, their breadth and depth. In some cases when the entire brain is of interest, an exhaustive "searchlight" search is conducted successively within small volumetric parcels, to again focus sensitivity on relatively limited brain areas, rather than the entire brain's features. In any case, multivariate analyses of brain activity do not consider all brain voxels/sensors jointly as features, and for this reason a preliminary selection of brain regions is mandatory for obtaining a neurobiologically-meaningful estimation. We put that the same holds when studying DNNs: it does not necessarily make sense to use all features (embeddings) for purposes of modeling a particular set of similarity judgments. These observations motivate the two aims of the current work.

Aim 1: Learn pruned DNN configurations that improve out-of-sample prediction of human similarity judgments from a DNN's penultimate layer, without activation reweighting. Pruning is implemented via feature selection that is supervised by human similarity structure. To the extent this aim was accomplished, we planned three derivative aims: 1) contrast the performance of pruning and reweighting-based approaches with respect to similarity prediction; 2) determine if pruned networks provide a better match to taxonomic structure (WordNet), 3) and evaluate whether classification errors for pruned networks indicate biases (increased attention) towards the category for which similarity judgments were obtained.

Aim 2: Use neurobiological data to supervise DNN-pruning so that pruning improves out-of-sample prediction of representational spaces manifest in multivariate patterns of human brain activity.

2 Study 1: Predicting human similarity judgments

2.1 Method: Data set

The image set and similarity ratings data we use were curated by Peterson et al. (2018) and kindly provided to us by the authors. The image set consists of images from six datasets, each with 120 images. The dataset represented categories that varied in perceptual and semantic heterogeneity. For consistency we refer to them using the labels introduced by Peterson et al.

1. ANIMALS: includes birds, reptiles and mammals of different types.
2. AUTOMOBILES: includes various transportation devices including sleds, horses, rafts, trucks, trains, wheel barrels, planes, blimps, and roller-skates.
3. FRUITS: fruits; mainly in the context of original vegetation.
4. VEGETABLES: vegetables; mainly in the context of original vegetation.
5. FURNITURE: mainly household furniture captured in indoor settings.
6. VARIOUS: a mix of images from the above categories but also including faces of people and outdoor scenery.

As evident, the datasets varied in taxonomic breadth, with VARIOUS being the broadest. Within each dataset, the authors obtained pairwise similarity ratings from humans for all combinations of 120 images; no similarity ratings were obtained for cross-dataset pairs, though the VARIOUS dataset can be treated as similarity mapping across superordinate-level categories.

2.2 Method: Network pruning via supervised feature selection

We perform separate analyses for each of the 6 different categories. Each category contains 120 images. The human similarity-space is the upper-triangle of the 120×120 Similarity-judgment Matrix, from here on referred to as SM_{HM}, and the DNN similarity space is the upper triangle of 120×120 pairwise cosine distances between images (SM_{DNN}). For the DNN analysis, activation values were extracted from the penultimate layer of VGG-19, which contains 4096 nodes. The 2OI between SM_{HM} and SM_{DNN} is quantified as their coefficient of determination, $R^2(SM_{HM}, SM_{DNN})$.

2.2.1 Main pruning algorithm

Algorithm 1 Pruning: Main algorithm

```
Inputs:
  • $SM_{HM}$: similarity Matrix of human similarity judgments
  • $SM_{DNN}$: similarity Matrix of similarity estimations derived from the DNN by computing the Pearson's correlation between the embeddings of two images

1. Compute baseline $R^2(SM_{HM}, SM_{DNN})$, using the full set of features.
2. Rank features
     • For each feature:
         – Remove the feature from original embeddings, compute reduced similarity matrix $SM_{DNNRED}$.
         – Calculate difference $D = R^2(SM_{HM}, SM_{DNN}) - R^2(SM_{HM}, SM_{DNNRED})$.
     • Rank features based on $D$, with higher values indicating great importance.
3. Construct pruned embeddings
     • Initialize an empty set of features.
     • Iterate over ranked features in descending order of importance according to $D$
         – Reinsert one feature at a time.
         – Calculate $R^2$ after each feature reinsertion, store values in array $a$.
     • Determine the maximum value in array $a$.
     • The index of the maximum value delimits the set of features to be included in the pruned embeddings.
```

To improve prediction of human similarity ratings from DNN activity, we implement the structural pruning of entire nodes (rather than single weights) using a variant of a sequential feature selection (SFS) algorithm that is supervised by human similarity judgments. The process, implemented separately for each dataset, consisted of *i*) determining feature contribution, *ii*) selection-to-criterion, and *iii*) out of sample testing. The process was repeated for 5 folds in a cross-validation framework. We present the algorithm and provide details in the subsequent sections.

Determining feature contribution: In each cross-validation iteration, we designated 20% ($n = 24$) of the images as a test set and 80% ($n = 96$) as the training set. $Baseline\, R^2$ is defined as the training-set's R^2 between the DNN similarity matrix and the human similarity matrix for those 96 images[2]. We quantify each feature's contribution to $Baseline\ R^2$ by removing only that feature and recomputing train-set-R^2. The feature is then reinserted and the next removed until the process is completed for all 4096 features. Consequently, 'relevant' features are those whose removal produces an R^2 value below $Baseline\ R^2$ and 'irrelevant' features are those whose removal produces a 2OI value above $Baseline\ R^2$. This produces a rank order of each feature's independent contribution to $Baseline\ R^2$.

Selection-to-criterion: After ranking, we consecutively insert features, according to their importance rank, into a candidate feature set. Each time a feature is added to the set, we recompute 2OI against the train-set human similarity judgments. We add all features exhaustively and then we identify the set of features associated with the maximal value reached. The set thus identified constitutes the pruned network associated with a specific fold. All the steps described so far are summarized in algorithm 1.

Out-of-sample generalization: Once a pruned node-set is determined, we apply it to the left-out test set. The test-set images are passed through the DNN, and coded as activation values for the retained nodes in the pruned layer. We then construct a pair-wise similarity matrix as described above and report pruned-net-R^2. We evaluate this value in relation to the *test-set's* $BaselineR^2$ which is the R^2 produced when considering all 4096 nodes rather than the retained subset. The overall out-of sample performance for a given dataset is the mean R^2 value across the five left-out test-set folds.

2.2.2 Alternative pruning algorithms

The sequential feature selection algorithm we use has the advantage of a rapid compute time when selecting from a large number of features, as the feature ranking stage scales linearly with the number of features. Other selection algorithms perform an iteration over the entire remaining feature set after selection of each feature, making them less practical. To evaluate if those algorithms produced better results we implemented the entire pruning pipeline using three other forward and backward selection algorithms and compared their performance to the main algorithm. The descriptions of these alternative algorithms can be found in *Appendix*.

2.3 Method: Regression-based reweighting

As an alternative to pruning we evaluated two regression-based approaches: one using ridge-regression, as introduced by Peterson et al. (2018), and another using LASSO-based regularization which we implemented here with an additional positivity constraint on the weights. The regression models were fit separately for each dataset, in the same way that pruning was optimized separately for each dataset.

Assuming Z is the total number of objects whose features are described, we can write the regression as follows.

For each pair of objects i and j, where $1 \leq i < j \leq Z$:

$S_{ij} = w_0 + w_1(A1_i \cdot A1_j) + w_2(A2_i \cdot A2_j) + \ldots + w_n(An_i \cdot An_j) + \varepsilon_{ij}$
where: S_{ij} represents the similarity value between Object i and Object j; $w_0, w_1, w_2, \ldots w_n$ are the weights or regression coefficients to be estimated for each feature interaction; $A1_i, A2_i, \ldots, An_i$ represent the feature values of Object i; $A1_j, A2_j, \ldots, An_j$ represent the feature values of Object j.

When using Ridge Regression to regularize the solution, the usual loss function is used:
$\text{RSS} + \lambda\left(\beta_1^2 + \beta_2^2 + \ldots + \beta_n^2\right)$

where RSS is the residual sum of squares, and λ is the regularization parameter that controls the amount of shrinkage applied to the regression coefficients and is non-negative.

When using LASSO for regularization, the following loss function is used:
$\text{RSS} + \lambda\left(|\beta_1| + |\beta_2| + \ldots + |\beta_m|\right)$

[2] Whenever we refer to similarity matrices for the purpose of R^2 computation, we only consider their flattened upper triangle

	Animals	Automobiles	Fruits	Furniture	Various	Vegetables
Baseline	0.61 (0.07)	0.51 (0.07)	0.33 (0.08)	0.29 (0.05)	0.43 (0.10)	0.32 (0.07)
PAG18	0.71 (0.09)	0.50 (0.05)	0.25 (0.15)	0.34 (0.08)	0.50 (0.13)	0.27 (0.07)
LASSO	0.64 (0.12)	0.51 (0.08)	0.38 (0.13)	0.37 (0.11)	0.47 (0.12)	0.31 (0.08)
Sim-DR	0.64	**0.57**	0.30	0.33	0.50	0.30
Pruned	**0.75** (0.05)	0.55 (0.08)	**0.39** (0.08)	**0.38** (0.07)	**0.56** (0.1)	**0.41** (0.05)
# nodes	807 (63)	647 (45)	563 (76)	557 (101)	830 (44)	605 (190)

Table 1: Pruning outperforms other methods in prediction of out-of-sample similarity judgments. R^2 for out-of-sample prediction of human similarity from pruned and original penultimate layer of VGG19 (baseline). For the pruned layer we also report the average number of nodes selected (± SD across folds).

where RSS is the residual sum of squares; λ is the regularization parameter that controls the amount of shrinkage applied to the regression coefficients and is non-negative; $|\beta_1|, |\beta_2|, \ldots, |\beta_m|$ represent the absolute values of the regression coefficients. Finally, we introduced our own non-negativity constraint on LASSO that requires: $\beta_1 \geq 0, \beta_2 \geq 0, \ldots, \beta_n \geq 0$

Following Peterson et al. (2018) the features were not (column) normalized prior to training the regression model. Instead, we followed those authors in applying normalization of values by object (i.e., by row). This operationalizes the assumption that, when comparing two objects, those object features that are more important in the context of a given comparison are those that are relatively more salient relative to each object's other features. To investigate whether this object-based normalization reduces sensitivity of the regression-based approach, we also implemented the regression model after applying column-normalization via Z-scaling or via min-max $(0 - 1)$ range normalization. We found that both these procedures produced poorer predictions than object-normalization for all datasets and so do not further discuss that analysis.

2.4 Result: Pruning outperforms competitive methods in prediction of human similarity

Pruning markedly improved the prediction of human similarity judgments for out of sample image-sets as compared to the non-pruned test-set Baseline. Results for each of the six datasets are reported in Table 1. Out-of-sample prediction improved for all datasets including VARIOUS. This first result is important in showing that pruning supervised by human behavior is a viable approach for extracting relevant information from DNNs.

Pruning also outperformed other reweighting/alignment methods used to date. For reference, Table 1 also includes the 2OI values produced by other alignment methods when tested on the same cross-validation folds. It can be seen that pruning outperformed the other three methods in 17 of the total 18 comparisons. The methods we compared are explained below:

1. Baseline refers to the match between the DNN and human similarity space prior to any modification, averaged across the five out-of-sample data for the test folds.
2. PAG18 refers to application of ridge regression as implemented by Peterson et al. (2018), but applied to the five out-of-sample folds used in our data.
3. Sim-DR is a reweighting approach developed by Jha et al. (2020) which optimizes a projection of DNN embeddings to a lower-dimensional space that matches human similarity judgments (we include values reported by the authors on the same dataset, as we have not implemented this learning model).
4. LASSO is our own variation of the reweighting implemented by PAG18 but which uses LASSO-regularized regression (Tibshirani, 1996) that is further constrained to only positive weights. As noted by others (Attarian et al., 2020) PAG18's use of ridge regression produced negative weights which is difficult to reconcile with intuition about psychological processes, as it means there are features for which a larger product-term results in reduced similarity.
5. Pruned refers to the node-pruning method we introduce here.

The number of nodes retained differed across datasets, and there was no strong indication of a relationship between number of nodes maintained and the R^2 achieved. For example, the ANIMALS and VARIOUS datasets were the ones with most nodes maintained, but were associated with markedly different R^2 values.

As indicated in *Methods*, an advantage of the sequential feature selection (SFS) we used is that it scales linearly with n, the number of features, whereas iterative forward- or backward-selection algorithms scale approximately with $n^2/2$. However, we also evaluated three other SFS algorithms to guide pruning; one forward-selection algorithm, and two backward-selection algorithms (see *Appendix*). The main findings of this analysis were as follows. All three under-performed compared to the main selection algorithm we used when tested on out-of-sample prediction of human similarity judgments. In addition, two of them, Backward selection by maximum 2OI (BWD1) and Forward selection (FWD) outperformed feature-reweighting for all six datasets. The most competitive of these three SFS algorithms was BWD1, which produced R^2 values that were only very slightly below our reference algorithm. However, BWD1 appeared to prune more effectively, producing smaller sets of features, e.g, achieving the same prediction accuracy for ANIMALS with only 400 features instead of the 800 features identified by our reference algorithm. These findings suggest that it may be useful to evaluate several different pruning algorithms for a given experiment, as the distribution of information across features may make some feature-selection algorithms more effective than others.

2.5 Result: Reweighting has no additive effect when applied to a pruned network

Pruning and reweighting embody different perspectives on the information contained in DNNs. That said, they may be conceptually and technically mutually compatible and synergistic. This would be supported if it were shown that applying reweighting to an optimally pruned network produces a further improvement in predictive strength.

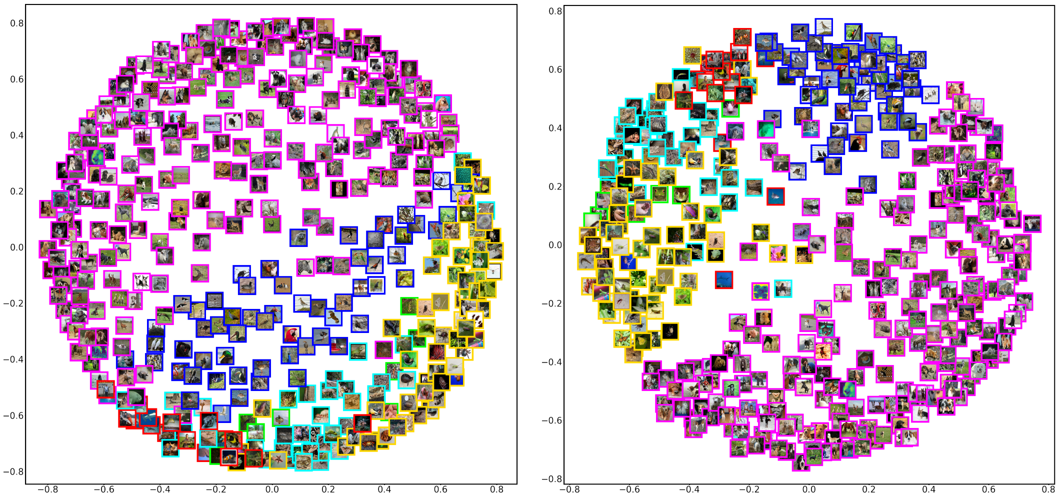
To determine whether reweighting offers an additional predictive benefit, we stacked a reweighting step on top of the pruned network in a context of cross-validation. Specifically, the cross-validation procedure consisted of two steps. The first identified the best-fitting pruned feature-set exactly as described in Section 2.2.1. Then, in a second step, we applied a reweighting procedure to those features selected within the fold, which constituted an opportunity to learn a better fit. We then applied both solutions (from step 1 [pruning alone] and from step 2 [regression after pruning]) to the validation set and stored the two R^2 values for comparison. For completeness, we implemented reweighting via three regression approaches: Ridge regression, LASSO, and Elasticnet.

We found that applying reweighting to a pre-pruned solution did not improve predictive capacity, but instead consistently reduced it for all six datasets (see Table 2). This suggests that the feature set identified by pruning already reflects relevant features at adequately tuned levels of salience, and that further tuning is not beneficial.

	Animals	Automobiles	Fruits	Furniture	Various	Vegetables
Pruned baseline	0.75 (0.06)	0.55 (0.09)	0.39 (0.08)	0.38 (0.07)	0.56 (0.11)	0.41 (0.06)
Ridge	0.72 (0.09)	0.49 (0.07)	0.31 (0.13)	0.35 (0.08)	0.51 (0.12)	0.30 (0.05)
Lasso	0.68 (0.12)	0.45 (0.1)	0.37(0.1)	0.33 (0.11)	0.40 (0.09)	0.28 (0.07)
Elasticnet	0.69 (0.1)	0.46 (0.1)	0.37 (0.1)	0.35 (0.1)	0.42 (0.09)	0.3 (0.06)

Table 2: Applying reweighting to pruned networks does not increase prediction accuracy. R^2 for out-of-sample prediction of human similarity from pruned networks (Pruned baseline) and from the same pruned networks subsequently reweighted using regression.

As a control, we evaluated whether the failure of reweighting to improve on performance of a pre-pruned network owes to low effectiveness of reweighting when applied to vectors with lower dimensionality (low node numbers) as opposed to the full 4096-node vector. We repeated this analysis, but selected a random set of nodes per dataset with the number of nodes matching those in Table 1, and applied reweighting to those nodes. In this case we found that all regression methods improved out of sample prediction for ANIMALS. In addition, for the other categories (apart from AUTOMOBILES), at least one regression method produced a fit above test-baseline. This means that reweighting can improve the fit when applied to a randomly selected small subset of features, but failed to do so when this set was determined via supervised pruning.



(a) Original VGG-19

(b) Pruned VGG-19

Figure 1: Multidimensional Scaling Plots of the embeddings corresponding to the 398 animal classes of ImageNet with Original VGG-19 and the same network pruned for Animals. Magenta-mammals, Yellow-invertebrates, Cyan-reptiles, Green-amphibian, Blue-bird, Red-fish.

2.6 Results: pruning reorganizes semantic space and improves alignment with WordNet's hierarchical taxonomy

2.6.1 Pruning produces tighter clustering in multidimensional space

To obtain a better insight into the similarity spaces, we produced Multidimensional Scaling (MDS) solutions for an independent dataset consisting of the entire set of Animal categories in Imagenet (Deng et al., 2012) (398 categories in all). The embeddings we used were extracted either from the original VGG-19 network, or from the VGG-19 network that was pruned using ANIMAL similarity judgments from the stimuli of Peterson et al. (2018). Note that these are two independent datasets.

For each of Imagenet's 398 animal categories we used the best exemplar image embeddings to construct the two dimensional MDS plot using sci-kit learn Python library with maximum iteration limit of 10,000 and converge tolerance of 1e-100. We chose the best fitting solution among four independent initializations. The images in the MDS plot were subsequently color coded to represent the six broad groupings of Animals within the WordNet (Fellbaum, 1998) taxonomy. Color-coding was performed after the MDS routine, and the groupings did not constrain the MDS solution in any way. The results, presented in Figure 1, clearly show that the pruned embeddings produce better-defined clusters, clustering fish, reptiles, amphibians and invertebrates more tightly.

To quantitatively determine this issue, for each image in these six superordinate categories, we computed the similarity to all images within the same superordinate category (Similarity Within, Sim_W), as well as its similarity to all images not sharing the same superordinate category (Similarity Between, Sim_B). These two statistics were then averaged over all images within each of the six superordinate categories. The results are shown in Table 3. We see that already in the baseline embeddings (Full Feature Set, column *Full*) there is a meaningful differentiation between Sim_W and Sim_B meaning that images are more similar to other images within their superordinate category than to images in other superordinate categories. Importantly, pruning consistently increased Sim_W and reduced Sim_B. That is, it created tighter clustering within a category and stronger separation between categories. The low Sim_W value for MAMMALS can be explained by the fact that this category is the largest (N=218 of 398) and most heterogeneous.

2.6.2 Pruning produces a hierarchical structure that better approximates WordNet

To evaluate whether pruning improves approximation of taxonomic structure, we analyze the same set of 398 animal images taken from each of the animal species in ImageNet. The fact that pruning improves 2OI suggests that it maintains features that are important for separation between animal

	Similarity Between		**Similarity Within**	
Category	Full	Pruned	Full	Pruned
Amphibians	0.02	-0.00	0.19	0.23
Birds	0.01	-0.01	0.13	0.18
Fishes	0.01	-0.01	0.17	0.19
Invertebrates	0.00	-0.01	0.10	0.11
Mammals	0.00	-0.02	0.04	0.06
Reptiles	0.01	-0.01	0.14	0.18

Table 3: Image-similarity to other images within and across the same superordinate category. Computed separately from image embeddings in the full and pruned VGG-19 network. Images were 398 images, independent of those used for pruning.

	$N = 6$	$N = 7$	$N = 8$	$N = 9$	$N = 10$	$N = 11$	$N = 12$
Orig. Vgg-19	0.20	0.20	0.20	0.18	0.18	0.22	0.22
Pruned Vgg-19	0.30	0.26	0.26	0.25	0.26	0.26	0.26

Table 4: Jaccard-Index concordance between a category's WordNet taxonomic neighborhood and its neighbors in DNN clusters. Higher values indicate greater agreement. Values shown for solutions across $N = 6 : 12$ DNN clusters. All comparisons statistically significant at $p < .01$ Bonferroni corrected for 7 comparisons.

categories, but because 2OI reflects ranking of pair-wise similarities, it does not directly address hierarchical structure. The analysis was carried out in two steps. We first produced a hierarchical taxonomic representation from the similarity spaces we had used to construct the MDS solutions (i.e., one solution for the original VGG-19, and one for a pruned variant supervised by ANIMALS). These reflected the hierarchical structure latent in the DNN similarity space of ImageNet's 398 animal species. We then compared those hierarchical structures to that of WordNet (Fellbaum, 1998), which is a lexical database that also includes IS-A relations between animal types. In this context, WordNet is taken to be the reference. We capitalize on the fact that ImageNet's labels are derived from WordNet. The detailed methods (see Appendix) report how we produced hierarchical structure from DNN embeddings using Hierarchical Clustering Analysis (HCA) and how we operationalized hierarchical structure from Wordnet. Once the two hierarchical structures were constructed we could compute the relative match between them. We used the Jaccard Index to quantify to what extent cluster-members in the DNN HCA were also nearby leaf nodes in WordNet (see Appendix). We repeated the DNN HCA analysis to produce solutions with $N = 6..12$ clusters and evaluated the results for each of these solutions.

As evident in Table 4, in all cases the pruned network's hierarchical structure produced a better match to WordNet than the original, non-pruned network. This was seen in that Jaccard Index values were higher for the pruned network, and this held independent of the number of clusters set as a parameter. To compute statistical significance we conducted an item-level paired analysis comparing the Jaccard Index value for each category for the original and pruned cases. In all cases, these values were higher for the pruned networks (Wilcoxon tests, Bonferroni corrected for multiple comparisons).

2.7 Pruning retains classification performance as well as strengthens categorization

The fully connected layers of DNNs contain substantial redundancy (e.g., Cheng et al., 2015), which suggests that pruned networks could retain adequate classification performance. We computed top1/top5 accuracy for the pruned networks that we derived (with each optimized for each of the six datasets in Table 1) and report the mean values across folds (Figure 2). Accuracy was computed for 50K independent images from ImageNet's validation dataset. The activations passed from the penultimate layer were modified in the following way. For pruning, only penultimate layer activations corresponding to the nodes retained after pruning were passed to the final layer of the network to classify. For Ridge regression as presented in PAG18, the ridge weights, optimized for each of the six datasets, were unconstrained and hence took both positive and negative values. For this reason, we chose to multiply the penultimate layer activations by the ridge weights with no further transformations before passing them to the final layer for classification. for Lasso regression, since we constrained the weights to be positive, we multiplied the penultimate activations by the square root of the Lasso weights. For the regression models, this analysis determines if the features that are picked out by regression as crucial for comparison are also ones that are important for classification. For this reason we did not evaluate further fine-tuning to regain classification after applying these weights.

As can be seen in (Figure 2), for pruning, top-5 accuracy never dropped below 79% and top-1 accuracy was between $56\% - 66\%$ (VGG-19's top-1 is 74.5). Pruning provided much better classification performance than non-modified Ridge regression (Peterson et al., 2018), where respective values never exceed top-1, 9%; top-5, 24%. Finally, lasso-based regularization produced categorization accuracy slightly lower than found for pruning.

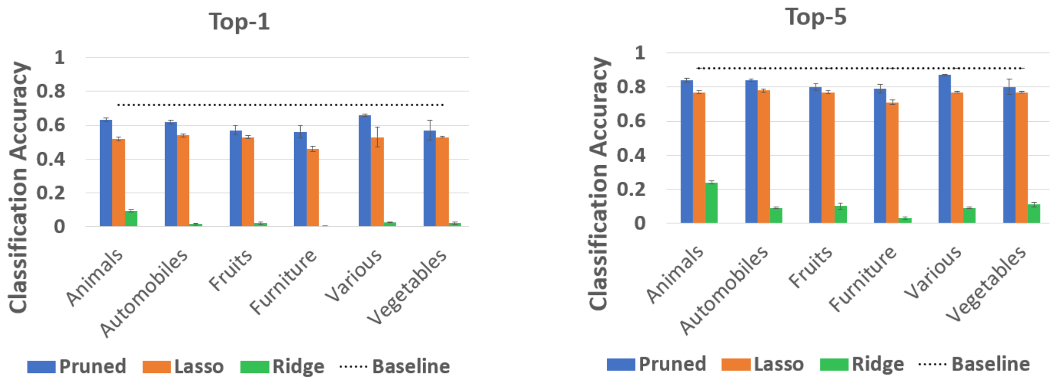
As indicated by the small error bars in Figure 2, categorization performance across folds was quite similar independently of the configuration of the training folds used for pruning. In fact we found that supervised pruning, as implemented across the different training folds (i.e, supervised via different similarity judgments) produced very similar distributions of activity patterns in the network's ultimate layer (1000-node post-softmax), when assessed for ImageNet's validation set. We computed for each image the correlation of its embeddings across folds. We found that the average correlation of the post-softmax distribution computed per image across folds often exceeded $r = 0.95$ (see Appendix). This suggests that different prunings, supervised by different sets of similarity judgments within a dataset, contain similar categorization-related information.

In addition, we see that pruning reduced top-1 accuracy, at minimum by around 5% as compared to VGG-19's performance. To understand if this was due to the fact that pruning emphasizes features that are relevant to the pruned category, we analyzed the error patterns exhibited by the original VGG-19 and by VGG-19 pruned to approximate human similarity judgments for the ANIMAL dataset. The images we passed through these networks were ImageNet's 50K validation set. In this analysis, we focused on 'substantial errors' made by these two networks, defined as images where the true label was not among the top-5 post-softmax activations for a given input image, for either the original or pruned network. This was repeated for each of the 5 folds/evaluations. When analyzing confusions within these substantial errors, we found that pruned networks more frequently classified images as animals. The magnitude of the bias, relative to the original VGG-19, appeared independent of whether the correct label belonged to the animal category or not. When the correct label was a type of Animal, the non-pruned VGG-19 labeled the image as (the wrong) animal 69% of the time whereas the pruned-net did so 91% of the time. When the correct label was not a type of animal, the values were respectively 5% and 30%. The fact that the bias was independent of whether the correct label was an animal or not suggests the pruned network did not develop enhanced (useful) sensitivity to animal features, but expressed a linear (constant) bias to classify any image as an animal. This supports the idea that pruning by ANIMALS maintains features that account for variance within this category, and that these features contribute strongly (i.e., produce strong activation) to animal categories in the output layer. The fact that pruning produced a higher rate of animal decisions, raising both hits and false alarms is a pattern very similar to that documented in Luo et al. (2021). In their study, they learned an attention layer to improve categorization of target category. Thus, optimizing for a category similarity judgments through pruning produces an effect that is similar to increasing weights via attention in that study.

2.8 Discussion of Study 1

The results of Study 1 are straightforward. Performance-wise, pruning outperformed all other methods on the out-of-sample prediction of human similarity judgments across five different datasets, and was competitive with state of the art on the sixth. The strongest benefit was seen for VEGETABLES, where pruning achieved an R^2 of 0.41 whereas no other method exceeded baseline (R^2 = 0.32). We also found that attempting to reweight an already pruned network offers no additional benefits, indicating that once an optimal feature set is identified, there is no further gain in trying to improve the saliency of the identified features via reweighting. This result held for three different regression methods (Table 2). As opposed to reweighting, pruning maintained classification accuracy.

In relation to the organization of the network's representational space, we find that pruning produces beneficial changes to the latent knowledge dimensions in the DNN with respect to the target set



(a)

(b)

Figure 2: Classification performance for Imagenet's validation set when passed through networks pruned or reweighted by human similarity judgments. Accuracy computed across the five folds. (a) Top-1 performance (b) Top-5 performance

of similarity judgments. Specifically, pruning that improved prediction of similarity judgments for Animals also produced a better structured MDS solution of animal categories, stronger clustering of animals within their superordinate categories (Table 3), and a better alignment with WordNet's hierarchical taxonomy (all found for an independent dataset of animals). Consistent with the idea that pruning is effective in identifying features that are important for the set of images supervising the pruning, we found that pruning from animal similarity judgments strongly increased the network's tendency to mis-classify novel images as animals.

3 Study 2: Predicting neural similarity spaces and probing representations

Study 1 demonstrated the effectiveness of pruning as a Machine Learning tool for prediction of human similarity judgments. However, pruning can be supervised by any $N \times N$ pair-wise similarity matrix, and in the context of neuroscience it offers a novel method to gain insights into the representational structure employed in different neural systems. In Study 2 we evaluated if the supervised pruning of DNNs using neurobiological data improves the ability to predict out of sample brain responses. We implemented Sequential Feature Selection on DNN embeddings as applied in Study 1. The main difference was that instead of using human similarity judgments we used pairwise similarity values derived from multivariate neurobiological data collected using fMRI.

3.1 Methods

3.1.1 Dataset and construction of representational dissimilarity matrices

The fMRI data and the image-materials were obtained from a public dataset made available by King et al. (2019) who examined, in part, the second order isomorphism (2OI) between DNN, human, and brain-derived similarity matrices. To maintain consistency with the neurobiology literature and the study by King et al., we use the term Representational Dissimilarity Matrix (RDM) to refer to a distance matrix computed by subtracting a pairwise similarity matrix from 1. The data set consisted of Human, DNN and Brain RDMs computed for two independent image-sets. Each image-set consisted of 144 images: 3 images from each of 48 categories. Different participants made similarity judgements for each set, making this dataset viable for testing cross-participant generalization. That is, it was possible to evaluate whether optimizing pruning for one set, based on data from one group of participants, improved the ability to predict behavioral and neural responses for a separate group of participants.

Human RDMs were derived using an item-arrangement method that produces pairwise distances between images. Brain-derived RDMs were computed using typical multivariate analyses for different regions of interest (ROIs) associated with object and scene perception (see King et al. for details). This made it possible to use each ROI as an independent supervisor for pruning.

For the DNN, the authors report RDMs constructed from embeddings extracted from the VGG-S architecture but we re-implemented the analysis using the VGG-19 architecture. This was to maintain consistency with our analyses in Experiment 1, and because pretrained networks for VGG-19 are more easily available than ones for VGG-S. We note that RDMs produced by VGG-19 (penultimate layer) and VGG-S (final layer used by King et al.) were quite strongly correlated: for the two sets, values were 0.76, 0.78. Correlation between the DNN RDMs and the behavioral RDMs was almost identical for VGG-S and VGG-19 (in both cases, Pearson's $R \sim 0.6$).

3.1.2 Tests of pruning

We evaluated eight brain areas for which data were provided: ventral temporal cortex (vTC), lateral occipitotemporal cortex (lOTC), fusiform face area (FFA), occipital face area (OFA), parahippocampal place area (PPA), occipital place area (OPA), and ventral and dorsal early visual cortex (vEVC and dEVC).

We implemented four tests of pruning. In the first, the DNNs were pruned based on behavioral RDMs and we evaluated the impact on predicting brain RDMs. In contrast, analyses 2, 3 and 4 used the brain RDMs themselves to supervise pruning of the DNN. To maintain compatibility with prior literature (e.g, King et al., 2019), in these latter analyses we considered both the penultimate and final ($n = 1000$ nodes) layers of the network (before softmax normalization).

In the first analysis, we pruned DNN nodes using the behavioral RDMs (one RDM per image set) and evaluated if this improves the 2OI between the DNN and any of the brain RDMs. In this case, for each image set a single pruned network configuration was determined by the best fit between a pruned DNN RDM and behavioral RDM. This pruned configuration was then used to derive (pruned) DNN RDMs that were compared with the brain regions' neural RDMs.

In the second analysis, separately within each image set, we pruned either VGG-19's penultimate layer or VGG-19's final layer (pre-softmax) to determine if pruning improves prediction of brain RDMs for out of sample images (within-set cross validation). For each brain ROI, we employed 5-fold cross-validation where we supervised pruning based on the group-level neural RDMs (average of all single-participant RDMs within each set separately). In each fold, 80% ($n = 115$) of the 144 images composed the train set, and the remaining composed the validation set. There was no overlap between the images used for training and validation. To evaluate the impact of pruning, 2OI was computed for the validation subset in each fold, both for the pruned and non-pruned embeddings.

In the third analysis, we used the two-fold approach that we had applied for the behavioral data where we pruned DNNs based on brain activations from one image set, and then evaluated the performance of the pruned network on the other image set (cross-set prediction).

In the fourth analysis, we implemented pruning outside a cross-validation context, where the complete brain RDM (from all 144 images) supervised pruning of a complete DNN RDM constructed from the same images. While this necessarily produces over-fitting, it can also be seen as an upper-bound indicator of the 'signal' contained in a given DNN layer with respect to its ability to predict a brain RDM, making it an important quantity.

In all these analyses we followed Experiment 1 in pruning nodes in the penultimate layer of VGG-19, also extending the analysis to the final layer as in King et al. However, the concept of pruning is not limited to removal of single nodes and can be effectively applied to earlier layers in a DNN. To demonstrate the feasibility of pruning earlier convolutional layers, we adapted the algorithm as follows: Instead of removing single-nodes, we applied pruning to single feature maps so that entire feature maps were removed one at a time to determine their relative importance. These feature maps were then inserted sequentially (most important first) to determine the set of feature maps that offered the best prediction of a brain RDM. To demonstrate the feasibility of this approach we applied it to each of four deepest convolutional layers in VGG-19 for purpose of predicting activity patterns for vTC. We also applied it to last convolutional layer when predicting activity patterns for FFA; in this case, a single layer was chosen for analysis as earlier layers very poorly approximated FFA activity patterns.

Table 5: Pruning supervised by human similarity-judgments improves 2OI R between DNN RDMs and Brain-region RDMs.

	vTC	lOTC	FFA	OFA	PPA	OPA	vEVC	dEVC
Set 1	0.130	0.099	-0.021	0.078	0.184	0.098	0.022	0.041
Set 1 pruned	**0.152**	0.097	**-0.018**	**0.089**	**0.207**	0.089	**0.054**	**0.060**
Set 2	0.150	0.073	0.009	0.066	0.187	0.139	0.023	0.017
Set 2 pruned	0.146	**0.079**	**0.027**	**0.071**	0.187	0.134	**0.033**	**0.019**

3.2 Results

3.2.1 Replication of study 1: Pruning by human similarity judgments improves prediction of out of sample human similarity judgments

Before conducting the main four analyses described above, we evaluated whether we could replicate a key result in Study 1, in showing that pruning supervised by human similarity judgments improves a DNN's ability to predict out-of-sample judgements. For each set separately we used all pairwise similarity judgments between the 144 images to prune the DNN's penultimate layer ($n = 4096$ nodes). Throughout our analysis, for each set, we used the mean behavioral RDM across all subjects within the set to supervise pruning. (We opted to prune using a mean behavioral RDM reflecting all 144 images rather than a mean behavioral RDM reflecting 48 categories in order to learn similarity relations within category.) We first applied cross-validation to each set separately, so that all data were sampled from the same participants. As in Study 1, we split the similarity judgments into train and test sets. Pruning, implemented via 5-fold cross-validation within each image set improved prediction of human RDMs for both Set1 and Set 2 (means across validation folds: Set1: $R^2 = 0.28 \rightarrow 0.36$; Set2: $R^2 = 0.22 \rightarrow 0.30$. In addition, because the two sets consisted of (different) images that belonged to the same categories, we could determine whether pruning the DNN based on the behavioral RDM of one image set improves 2OI for the other image set (a simple two-fold cross-validation). For both sets we found improved 2OI. Set1: $R^2 = 0.25 \rightarrow 0.29$; Set2: $R^2 = 0.22 \rightarrow 0.25$. To conclude, pruning improved out of sample prediction when applied to data collected within or across participants.

3.2.2 Pruning from human similarity judgments improves prediction of neural RDMs

We find that pruning a DNN based on a behavioral RDM improved the isomorphism between the DNN RDM and RDMs of the 8 regions of interest. Table 5 presents the 2OI R values for the raw and pruned DNNs. A paired t-test applied to the Fisher-Z transformed R values (considering the 16 comparisons; 8 regions per two sets) confirmed these impressions statistically, $t(15) = 2.88, p = 0.005$ (one tailed for $pruned > raw$ directional test). This shows that even a highly generic pruning based on human similarity judgements can improve 2OI with brain ROIs. As indicated above, the remaining analyses used each brain region's RDM separately to supervise DNN pruning.

3.2.3 Pruning from neural RDMs improves out of sample prediction of neural RDMs

Figure 3 shows results for within-set cross validation. We find that supervising DNN pruning from brain RDMs was highly effective in increasing the ability to predict brain RDMs for out of sample images. In evaluating the brain ROIs we found improved prediction for 31 of the 32 analyses (pruning evaluated for 8 regions, for two image sets, for two layers). To evaluate the data statistically, we analyzed results for each layer separately. For the penultimate layer, we contrasted the 16 raw correlation values with the 16 values obtained from the pruned network (after Fisher-Z transform). The mean values differed markedly ($M_{pruned} = 0.13$ vs. $M_{raw} = 0.08$) and the difference was statistically robust, accompanied by a large effect size, $t(15) = 7.02, p < .001, d = 1.75$. A similar result held for the final layer, $M_{pruned} = 0.16$ vs. $M_{raw} = 0.09$), $t(15) = 6.06, p < .001, d = 1.51$.

As in King et al. (2019), we found relatively weak results for the FFA when using the full embeddings extracted from the original DNN model (for both the penultimate and final layer). However, supervised pruning could still slightly improve the prediction of this region's RDM, particularly when applied to VGG-19's final layer (see Figure 3B). Qualitatively, for vTC pruning of the penultimate layer was more effective than pruning the final layer, but for FFA, vEVC and dEVC a converse pattern

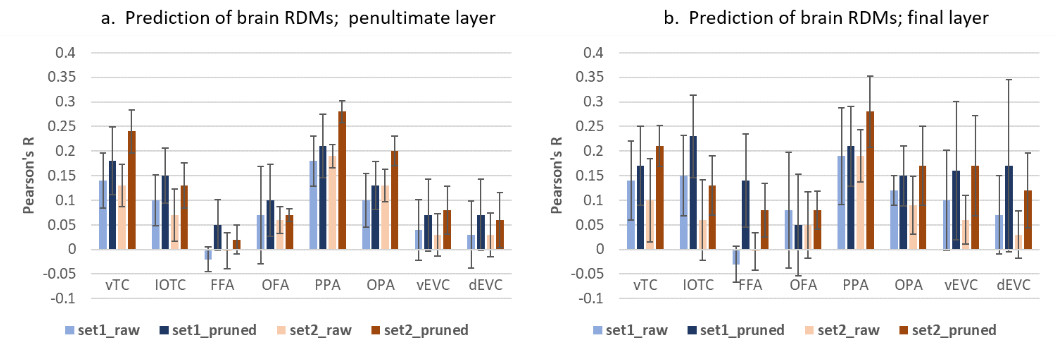


Figure 3: Learning pruning within each image set for separate cortical ROIs. Pruning was tested using 5-fold cross validation, separately for image set 1 and image set 2. (a) Prediction from embeddings from VGG-19 penultimate layer (b) Predictions from embeddings from VGG-19 final (1000-node) layer

held suggesting these regions benefit from pruning information coded at the category level of the final layer. As we discuss below, this could occur whenever a distribution of values in the final layer constitutes an effective lower-dimensional space.

Figure 4 reports results of the cross-set RDM prediction. Here, a DNN pruned by an RDM produced from one image set was used to predict brain activity recorded while participants observed a different image-set. This analysis therefore generalizes over both between-participant and between-image-set variance. Here too pruning improved prediction of brain RDMs almost without exception (30/32 of cases examined). And as in the within-set analysis, pruning improved the ability to predict FFA RDMs, and more strongly so when using the final layer. The mean values differed for both the penultimate layer ($M_{pruned} = 0.13$ vs. $M_{raw} = 0.08$), $t(15) = 9.37, p < .001, d = 2.34$, and the final layer, ($M_{pruned} = 0.14$ vs. $M_{raw} = 0.09$), $t(15) = 4.30, p < .001, d = 1.07$.

Extending this analysis, we also implemented the cross-set RDM predictions by applying pruning to feature maps in the convolutional layers themselves (see *Methods*). To demonstrate feasibility we analyzed the vTC area against information stored in convolutional layers $13 - 16$ in VGG-19. A first observation is that the convolutional layers presented higher baseline match to the brain RDM. For the penultimate layer, Pearson's R was below 0.15 at baseline (for both datasets), and improved to around 0.2 when pruned feature-sets were used. In contrast, baseline values were higher in conv13, with values of $R = 0.24/0.17$ (for the two datasets we studied), and importantly, improved to $R = 0.34/0.24$ when using pruned feature maps. A similar pattern was found for conv14; $R = 0.22/0.17$ at baseline, improving to $R = 0.32/0.25$. The same pattern, though with lower values and more modest improvement held for conv15 and conv16 where the results approximated those found for the penultimate layer.

We were also curious to know whether pruning a convolutional layer would provide stronger improvement in prediction of FFA activity, for which we could not surpass $R = 0.05$ when pruning the penultimate layer. We selected conv16 for this analysis because it offered the highest baseline second-order isomorphism values for that region. However, for FFA did not find a strong improvement, with pruning only modestly improving prediction from $R = 0.03/-0.03$ to $R = 0.08/0.04$. These results show that pruning is easily extended to entire feature maps, and in some cases may reveal more sensitive results than when pruning the penultimate layer. However, this is not necessarily the rule and will vary by the target domain supervising the pruning.

Finally, we evaluated the impact of directly pruning DNN embeddings based on brain RDMs outside a validation context (Figure 5). This shows to what extent pruning improves the match between DNN and brain RDMs when applied to embeddings produced from the same image set. As seen in the Figure, pruning improved isomorphism between DNN and brain-region RDMs across the board, often by substantial multipliers.

Importantly, this last analysis also allowed us to determine how many features were retained when supervising the pruning using each RDM. As shown in Table 6, when applied to the final layer of VGG-19 ($n = 1000$ nodes), pruning generally produced a sparse configuration with a very low

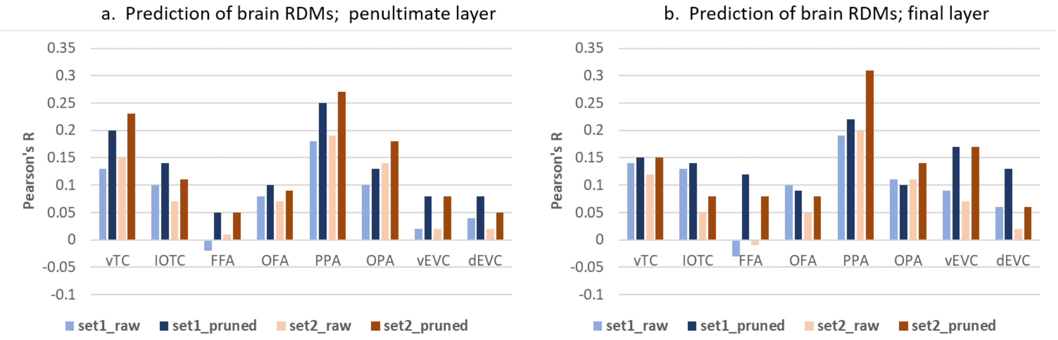


Figure 4: Learning pruning across image sets. Pruning was learned for DNN embeddings for one image set and applied to predict brain activity patterns associated with a different image set. (a) Prediction from embeddings from VGG-19 penultimate layer (b) Predictions from embeddings from VGG-19 final (1000-node) layer

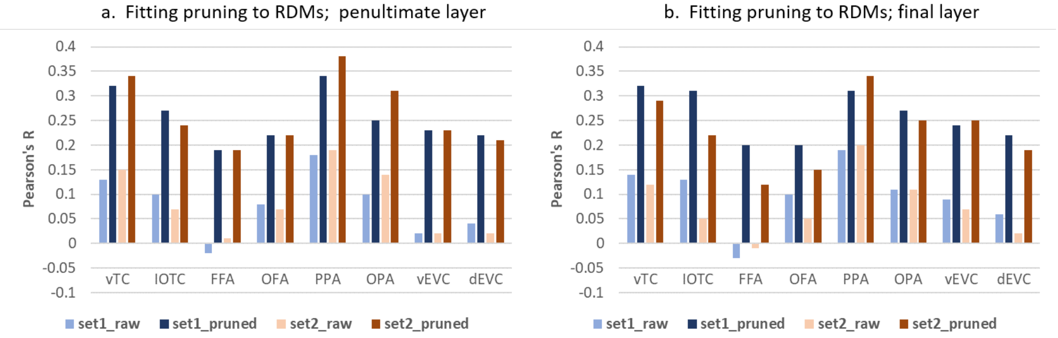


Figure 5: Direct pruning of DNN RDMs from brain-region RDMs without generalization. (a) Fitting VGG-19 penultimate layer (b) Fitting VGG-19 final (1000-node) layer

number of nodes. What immediately stands out in the Table is that pruning DNN embeddings from FFA RDMs was optimized by selecting as few as ten nodes, for both image sets (out of 1000 nodes in the layer). In contrast, PPA required a substantially larger number of nodes and was associated with the largest number of retained nodes for both image sets. There was a moderate agreement between the number of nodes retained across sets ($n = 8$ regions, Pearson's $R = 0.52$) indicating that the number of nodes retained through pruning, per brain ROI, is systematically linked to the information coded for in that brain area for different image sets and by different individuals.

It is important to keep in mind that pruning embeddings from the final layer does not mean that a given brain region is necessarily sensitive to defining features of the retained category labels. It only means that the multivariate activity pattern in those nodes across images, as expressed in a DNN RDM, tracks the brain-region's RDM. This multivariate activity pattern would reflect any meaningful covariance between the penultimate layer and final layer. For example, for Set 2, the brain RDM for FFA was optimized by selecting only 7 of the 1000 nodes, and these 7 nodes had the following labels: *geyser, volcano, killer-whale, steel-arch-bridge, steam-locomotive, electric-locomotive, strainer*. This just means those 7 nodes constitute a useful lower-dimensional space for tracking regional-FFA response, and the reason for this needs to be explored using methods suitable for studying lower dimensional spaces in DNNs. The next section introduces such a method for studying which image sections are relevant to the match between a brain ROI and the DNN whose pruning it supervised.

3.2.4 The impact of brain-supervised pruning on representational space

Because pruning fleshes out shared dimensions between a brain ROI and a pruned DNN, it is possible to identify, for a given image, the contribution of each image section to those shared dimensions. The principle is based on evaluating the impact of masking a part of a single image on the 2OI between

Table 6: Number of nodes retained from VGG-19 final layer for pruning based on different ROIs.

	vTC	lOTC	FFA	OFA	PPA	OPA	vEVC	dEVC
Set 1	27	29	10	60	75	40	41	71
Set 2	40	79	7	128	138	84	12	18

the DNN and Brain RDMs. In brief (see *Appendix* for methods details), we consider as input a set of N images presented for viewing in an fMRI scanner. One image is selected for analysis and for this target image we compute an RDM capturing the correlation between the target image and each other image. Correlations not involving the target image are not considered. One RDM is computed from Brain data and another from DNN embeddings. The second-order-isomorphism value for these two RDMs is taken as *baseline* 2OI, $2OI_{base}$. A portion of the target image is then masked, and the DNN RDM is recomputed, whereas the brain RDM remains unaltered. This produces a modified second-order isomorphism $2OI_{mask}$.

If the masked area changes the DNN RDM in a way that reduces its 2OI with the brain RDM, i.e., $2OI_{mask} < 2OI_{base}$, this means that the masked area contains information that loads on a latent dimension that contributes to 2OI. In contrast, if masking does not reduce $2OI_{base}$, or even improves on it, the information within it is less relevant to shared dimensions. The contribution of an image patch is therefore simply $Contrib = 2OI_{base} - 2OI_{mask}$ with higher values indicating greater importance of the masked area.

To make this concrete, consider a set of ten images where images 1-5 include a face and images 6-10 do not. Assume that a certain brain area only codes for the presence of a face. This brain area's RDM will separately cluster images 1-5 and images 6-10. Assume also that a DNN has been pruned by this brain area, and therefore produces a similar RDM. The relation between the two RDMs is quantified via $2OI_{base}$. Image 1 is chosen as the target image, and the face depicted in that image is masked. The DNN RDM for correlations with Image 1 changes: now, images [2-5] are strongly clustered but image 1 clusters with images [6-10]. Because the brain RDM remains unaltered. the result is a reduction in $2OI_{base}$ because the masked region was related to a dimension that organized both RDMs. Contrarily, if a non-important part of Image 1 were masked, the DNN RDM would not change, and so $2OI_{base}$ would remain unaltered. Implementation details can be found in the Appendix.

To apply this method we used brain RDMs from vTC and PPA and the two DNNs pruned by these RDMs. For any given target image, the image was masked by sweeping a mask over the entire image, and assigning a Contribution score to a 4×4 pixel area in the center of the mask. Following prior work Palazzo et al. (2020) masks at different scales were applied, and we selected the $Contrib$ value that departed most strongly from zero as the value assigned to the center of the mask. As an internal control, the analysis was also repeated by computing DNN RDMs from a non-pruned version of VGG-19. This control identifies shared dimensions between the 'vanilla' non-pruned network and a given brain RDM. The code for this can be found on Github[3].

A sample result is shown in Figure 6 (see *Appendix* for all results). As shown, the method is highly useful for identifying types of information that may be important for a given brain area. In the outdoors image, for vTC, masking of sky-areas strongly perturbed $2OI_{base}$, but this was found for the pruned DNN only. For PPA, in contrast, the unpruned DNN identified the face as important, but the pruned DNN notably excluded face information. We note that these effects were mediated by the global rather than local structure of the image: applying the method to target images rotated by 180-degrees (e.g., sky is below) produced substantially different heatmaps.

3.3 Discussion of Study 2

Study 2 replicated and extended the finding of Study 1 in showing that pruning supervised by human similarity judgments improves out of sample prediction of human similarity judgments, here found to generalize across participants. DNN-pruning supervised by human similarity judgments was sufficient to improve prediction of neural RDMs.

[3] https://github.com/tlmnhut/Visualize_PrunedDNN_by_HumanSim

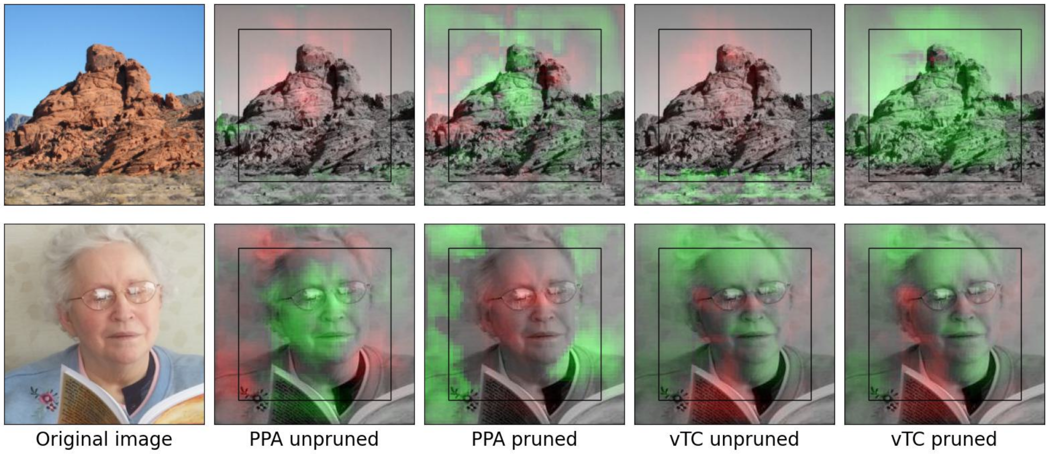


Figure 6: Heatmap showing the contribution of each image section to second order isomorphism between a DNN RDM and a Brain RDM. 'pruned' and 'unpruned' refer to whether or not the brain region supervised the pruning of the DNN. Green colors indicate image areas whose features contribute to shared DNN/Brain dimensions. The area within the inner black square was captured by masks at all scale-sizes; areas outside the black square also included padded data.

Most importantly, we found that supervising the pruning of DNNs directly from neural RDMs strongly improves the ability of predicting out of sample neural RDMs. This shows that pruning increases the sensitivity of these analyses, which are becoming an increasingly common approach in Neuroscience. In this context, we found that pruning supervised by neural RDMs produced for image-set1 improved prediction of neural RDMs produced for image-set2, obtained from a different group of participants.

Pruning improves understanding of brain function. Here we found that pruning from neural RDMs associated with different brain regions produces different levels of sparsity at the final layer of VGG-19. Specifically, pruning by PPA RDMs was associated with the least sparse configuration, for both image sets, whereas pruning by FFA RDMs was associated with the most sparse configuration. Finally, by using a brain RDM to supervise pruning, it is possible to gain insights into the filters instantiated in different brain areas by identifying the relative importance of each image section to the shared dimensions between the pruned DNNs and the brain area.

4 General Discussion

4.1 Core technical results

The advantages of pruning as a supervised learning method were detailed in sections 2.8 and 3.3 and we only briefly summarize them here. Pruning outperformed state-of-the-art regression-based methods in predicting human similarity judgments, and furthermore, stacking a regression-based model on top of the pruning solution produced no additional predictive power. In a study of animal images, passing an independent dataset through a DNN pruned by a different animal dataset produced better clustering of animal types into superordinate categories and a better approximation of WordNet animal hierarchy. All this indicates that performance-wise, pruning can be applied in the context of human-oriented AI, with a potential contribution to applications based on similarity (e.g. recommendation systems).

Pruning also generalizes well. Beyond prediction based on train-test folds in which data were obtained from the same cohort of participants, it also generalized across both image-sets and participant-cohorts in behavioral and neuroimaging data (Study 2). This suggests that pruning identifies meaningful dimensions that are shared between representational spaces.

4.2 Why pruning works and directions for future

Our theoretical motivation for studying the potential of supervised pruning originated from the intuition that DNNs trained for categorization may contain a wealth of information that describes each image, only part of which is pertinent for modeling the representational geometry *of any specific domain*. Pruning effectively supervised the learning of neural and behavioral representational spaces, but the reasons for its success in these two cases is likely quite different.

For neural representational spaces, the effectiveness of pruning probably owes to the fact that different brain areas code (or filter) for different information, and this filter is approximated by pruning. In future work, pruning may help clarify hierarchical processing in the brain by identifying brain areas whose feature space approximates a combination of feature-spaces identified for lower-level regions.

For human judgments, the success of pruning is probably related to the fact that for a given domain (e.g., fruits) the relative distance between objects, as operationalized by human similarity judgments, is mainly determined by features that differentiate objects within the domain. In contrast, less-relevant features will include those that categorically differentiate that domain from others (on which all domain members may have similar values), or features that are unique to the domain but less important for the comparison judgment. Both sorts of features are good candidates for pruning.

Given that behavior-supervised pruning has not been implemented to date, the current effort has some limitations that point to specific research directions in the future. From an implementation perspective, we implemented pruning as an additional machine learning step rather than merging it into the DNN training itself. Future work could integrate supervised pruning with the classification training as in Piggyback (Mallya et al., 2018). It would also be important to evaluate the efficacy of pruning for other DNN architectures.

Another potential improvement of the current method would be performing feature-selection over DNN features prior to application of supervised pruning. Several approaches are viable here. For example, Wang et al. (2021) present a combined feature selection and feature extraction approach where the original features are projected into a projection matrix that approximates the information in the original data, but considers only information contained in a subset of the original features. This maintains features that are more meaningful, while ablating noisy/meaningless ones. Because it optimizes both dimensionality reduction (extraction) and feature-selection in a single step, it can provide a better starting point for applying the supervised pruning we present in the paper. A related approach involves supervising DNN pruning, but using the similarity matrix produced when using the DNN's full feature set. This method aims to identify a smaller subset of features that reproduces that similarity matrix. Using this approach, we have observed that for the datasets used in Experiment 1, less than 50% of the features are required Truong and Hasson (2022). In summary, it is important to consider methods that narrow the search space by eliminating irrelevant features, which can be used as a starting point for pruning supervised by human data.

The method we presented can be used in tandem with architectures that combine inputs from a vision-oriented DNN and inputs from a DNN processing human behavior to create a shared latent code (e.g., Palazzo et al., 2020; Liu et al., 2023). Pruning can complement such procedures to constrain the learned feature space (either pre-learned or during training) so that object-distances satisfy a target similarity matrix. More generally, constraining feature spaces using human knowledge offers several advantages. Given that DNNs achieve categorization by relying on information distinct from that considered by humans, they manifest low explainability and heightened vulnerability to specific attacks. which has prompted researchers to try and incorporate ground-truth human knowledge into classifiers. For instance, in Li et al. (2023), semantic knowledge pertaining to relationships between animal body parts is explicitly encoded to influence DNN-based decisions. While further work is needed, we posit that supervised pruning, which selects for features important for human object-comparison, inherently emphasizes shape-based features and other features meaningful for humans.

4.3 Conclusions

These practical advantages and the future potential of supervised pruning are secondary to the theoretical implications of the current study. Our findings indicate that DNNs already capture features relevant to human similarity spaces (quantified behaviorally or via brain recordings) at an adequate level of salience. For this reason, node activations do not need to be reweighted. One just needs to filter out those features/nodes that are less relevant to modeling the similarity space of the domain at hand. Supervised pruning therefore improves the sensitivity of quantifying isomorphism between DNNs and humans, and opens the door to new studies of semantic knowledge in artificial and biological systems, with greater precision and potential for explainability.

5 Appendix

5.1 Alternative feature selection algorithms

In addition to the Main Algorithm described in the text, we evaluate three other sequential feature selection approaches. Because compute time for these scales exponentially with number of features, the following steps were taken to reduce compute time. We note these steps were only applied to the BWD1, BWD2, and FWD algorithm in order to reduce compute time, and were not applied to the Main algorithm used throughout the text. First, we removed any feature that was uninformative because it coded '0' for all images in the dataset, or was highly correlated with another feature at a level exceeding Pearson's $R \geq 0.95$. In addition, for BWD1 and BWD2 we began the selection from a reduced set of $n = 1500$ initial features, which we determined using the feature-ranking step of the main algorithm. Moreover, early stopping was implemented for BWD1, BWD2 and FWD, so that feature ranking halted once 2OI achieved its maximum 2OI on the training set.

Average performance on prediction of human similarity judgments for held out data are presented in Table 7. The number of features maintained when using each algorithm is shown in Table 8.

Table 7: Average out of sample prediction accuracy (R^2; 2OI) obtained using different feature-selection approaches computed on the test set, with the standard deviation in brackets. Main Algorithm refers to the feature selection algorithm used throughout the study

	Animals	Automobiles	Fruits	Furniture	Various	Vegetables
Baseline	0.60(0.03)	0.50(0.10)	0.27(0.05)	0.31(0.04)	0.45(0.07)	0.34(0.03)
Main Algorithm	0.75(0.06)	0.55(0.08)	0.39(0.09)	0.37(0.08)	0.56(0.11)	0.41(0.06)
BWD1	0.75(0.06)	0.53(0.05)	0.39(0.1)	0.38(0.11)	0.54(0.12)	0.38(0.08)
BWD2	0.62(0.12)	0.49(0.08)	0.34(0.06)	0.40(0.06)	0.19(0.08)	0.37(0.05)
FWD	0.73(0.06)	0.53(0.06)	0.36(0.11)	0.37(0.09)	0.51(0.12)	0.37(0.09)

Table 8: Average number of retained features obtained with the different feature-selection approaches, with the standard deviation in brackets.

	Animals	Automobiles	Fruits	Furniture	Various	Vegetables
Main Algorithm	806(78)	654(60)	572(92)	582(108)	831(45)	559(212)
BWD1	408(26)	458(39)	518(58)	411(22)	489(41)	432(31)
BWD2	70(15)	574(233)	505(336)	304(106)	116(42)	667(178)
FWD	464(16)	615(65)	516(48)	478(44)	517(28)	590(27)

Algorithm 2 Backward selection by maximum 2OI (BWD1)

```
Inputs:
    • $SM_{HM}$: similarity Matrix of human similarity judgments
    • $SM_{DNN}$: similarity Matrix of similarity estimations derived from the DNN by computing
      the Pearson's correlation between the embeddings of two images

    1. Temporarily remove one of the $n$ features
    2. Compute the reduced DNN similarity matrix $SM_{DNNr}$ from the embeddings with $n - 1$
       features
    3. Compute $R^2(SM_{DNNr}, SM_{HM})$
    4. Repeat the $R^2$ computation for every subset of $n - 1$ features
    5. Permanently remove the feature leading to the highest $R^2$
    6. Repeat the whole process starting from $n - 1$ features
    7. Rank the features based on the order of removal, with the first removed feature being the
       least important
```

Algorithm 3 Backward selection by minimum 2OI (BWD2)

```
Inputs:
    • $SM_{HM}$: similarity Matrix of human similarity judgments
    • $SM_{DNN}$: similarity Matrix of similarity estimations derived from the DNN by computing
      the Pearson's correlation between the embeddings of two images

1. Temporarily remove one of the $n$ features
2. Compute the reduced DNN similarity matrix $SM_{DNNr}$ from the embeddings with $n - \mathit{1}$
   features
3. Compute $R^{\mathit{2}}(SM_{DNNr}, SM_{HM})$
4. Repeat the $R^{\mathit{2}}$ computation for every subset of $n - 1$ features
5. Permanently remove the feature leading to the lowest $R^{\mathit{2}}$
6. Repeat the whole process starting from $n - 1$ features
7. Rank the features based on the order of removal, with the first removed feature being the
   most important
```

Algorithm 4 Forward selection (FWD)

```
Inputs:
    • $SM_{HM}$: similarity Matrix of human similarity judgments
    • $SM_{DNN}$: similarity Matrix of similarity estimations derived from the DNN by computing
      the Pearson's correlation between the embeddings of two images

1. Select one starting feature as that whose removal causes the greatest decrease in
   $R^{\mathit{2}}(SM_{HM}, SM_{DNN})$
2. Compute the partial DNN similarity matrix $SM_DNNp$ from every pair of DNN features
   consisting of the initial one and one of the remaining
3. Compute the $R^{\mathit{2}}$ with human similarity judgements for all the similarity matrices obtained
   from the previous step
4. Keep as second feature the one leading to the highest $R^{\mathit{2}}$ value
5. Repeat the entire process starting from 2 features and adding one more at each iteration
6. Rank the features based on the insertion order, with the first inserted being the most important
```

5.2 Detailed methods: Hierarchical analysis of WordNet vs. Pruned and non-pruned DNNs

To determine hierarchical information latent in the DNN similarity spaces we evaluated the relative fit between WordNet's hierarchical structure and that of the pruned and non-pruned similarity spaces. From the DNN's similarity spaces we derived hierarchical clustering analysis solutions based on a distance matrix computed from pair-wise cosine distances. We used the scipy Python library, dendrogram with complete linkage function and the leaves within each cluster of the dendrogram were distance sorted in ascending order. To form the desired number of clusters from the dendrograms, we used *fcluster* from Scipy Python library with our criterion specified as maxclust.

For the DNNs we computed HCA solutions from the embeddings in the penultimate layer associated with the best exemplar of each category. The exemplar was the image that produced the correct decision with highest confidence of all category members. From these embeddings we computed similarity matrices and HCA solutions with $N = 6..12$ clusters. To define the neighborhood-set of each category in the DNN's HCA result, we extracted for each category member the set of all categories in the same HCA cluster.

To define the neighborhood-set of each category in WordNet we looked up the category in the WordNet graph, and extracted all leaf nodes subsumed by the category's grandparent (two links above). This increased the granularity of WordNet as in many cases a category node had no siblings or very few ones, which made it non-feasible to use direct siblings as a neighborhood. This effective smoothing also usefully countered some ontological sub-divisions in WordNet that are not likely to have a counterpart in human similarity space. Specifically, WordNet contains multiple graph sections that increase in depth of IS-A links but without splitting (i.e, chains of parents that have only a single child; e.g., scorpaneoid $\rightarrow$ scorpaenid $\rightarrow$ lionfish). This is a knowledge-structure that would not appear to have a direct psychological analogue and can reduce the psychological validity of directly using Wordnet distances as a proxy for conceptual distances (e.g., Huang et al., 2021). For the 398 categories we used, 100 were ones for which the target-node's grandparent only subsumed a single leaf node (i.e., the target). For this reason we excluded these 100 categories from the analysis (as they did not even have "cousins"). We further only analyzed categories with neighborhood-set sizes between 10 and 160, which resulted in using 245 categories of the total 398.

As a final step, we computed the set-match for each of these 245 categories by determining the Jaccard Index between the category's DNN neighborhood-set and WordNet set (intersection of sets divided by union of sets). A grand mean was then computed over all categories.

5.3 Different instantiations of pruning produce similar post-softmax activation values

We examined if two pruned networks, both pruned to optimize prediction of human similarity judgments of the same semantic category, develop similar representations. Because we established performance of pruned networks across test-folds (5 folds per category) we could determine to what extent different instantiations of the pruned layer (one per fold), for a given category, produced similar activation patterns at the final (categorization) layer. Finding similar activations would suggest that different prunings, supervised by different sets of similarity judgments, reflect similar categorization-related information. For each category separately, we examined post-softmax final-layer patterns for pruned configurations produced in the 5 folds. Specifically, we used ImageNet's 50K validation set. Per fold, we saved the 1000-valued vector of softmax outputs per image. Then, for each pair of two folds among the possible combinations of (5 choose 2), we computed the correlation between the softmax values for the same image across the two folds. We finally took the mean of those cross-fold correlations as a measure of correspondence at the final layer.

We found substantial consistency in post-softmax activation vectors. For ANIMALS, Min/Max values for cross-fold correspondence were $R = 0.92 - 0.96$. For the other categories these values were, Vehicles: $0.90 - 0.95$, Fruits: $0.85 - 0.94$, Furniture: $0.86 - 0.96$, Various: $0.94 - 0.96$, Vegetables: $0.68 - 0.93$.

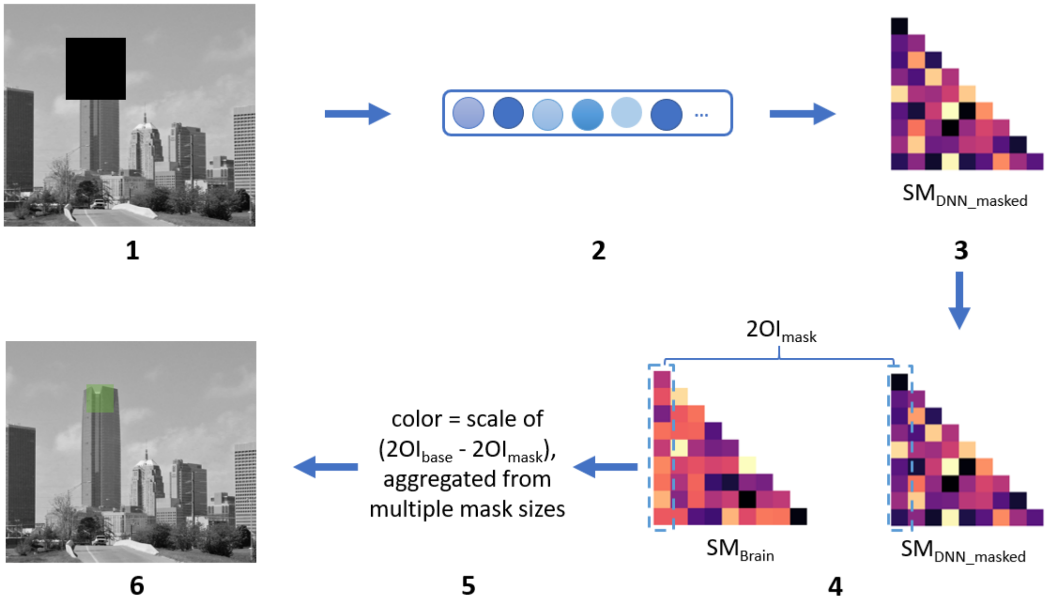
5.4 Production of second-order-isomorphism image-specific heatmaps

An image was masked using a sliding mask to evaluate how the masking of each image section impacted the 2OI between the DNN RDM and Brain RDM.

Figure 7 describes the main steps in the analysis. All 144 images in Set2 of King et al. (2019) were passed through a pruned DNN to extract embeddings. From these we constructed a baseline Similarity Matrix, (SM_{DNN_base}). The correlation between SM_{DNN_base} and SM_{Brain} constituted ($2OI_{base}$). The masking procedure was applied to a target image and applied as follows. Masks were square 0-filters, and their sizes were set the range 24-56 pixels in intervals of 4 pixels (9 mask sizes in all). We used variable sizes to be sensitive to features of different granularity. The stride step size was set to 4 pixels for all filter sizes. We added zero padding to the edges of images as required depending on the size of each masking filter. As described in the main text, the perturbation to $2OI_{base}$ induced by each mask (computed as $2OI_{mask}$) was assigned to a 4×4 area at the center of the mask. This produced 8 perturbation values for the center of each set of 8 masks, of which we selected the value associated with the maximum absolute value (i.e., the negative or positive value that departed maximally from zero).

This entire procedure was applied to DNN embeddings extracted from VGG-19 DNNs whose embeddings were pruned as supervised by brain RDMs, or to embeddings derived from a non-pruned version of VGG-19 (internal control). In all cases we used a VGG-19 pretrained model as provided in Pytorch.

To visualize the perturbation scores we colored the 4×4 area in the center of each mask to avoid overlapping colors. Green colors denote a positive score, meaning the masking the given area produced a drop in 2OI, whereas red denotes the converse. Figure 8 presents more sample results.



color = scale of ($2OI_{base}$ - $2OI_{mask}$), aggregated from multiple mask sizes

Figure 7: **Main steps in producing 2OI-perturbation heatmap**. 1. A section of the target image is masked. 2. The masked image is passed through the DNN and the image embeddings are extracted. 3. A Similarity Matrix is constructed to reflect the distance between the masked image and all other images (SM_{DNN_masked}); only correlations involving the target image are considered from this point on. 4. A 2OI value is computed by relating this set of correlation values to the set computed from brain data (SM_{Brain}). Those correlations involving the target image (here, e.g., Image 1) are delineated in the Figure by a dashed blue square. The two sets of correlation values are related via R^2 coefficient of determination. 5. The difference between $2OI_{mask}$ and $2OI_{base}$ is stored as the impact of the mask. 6. The magnitude of the difference is mapped onto a color scale.

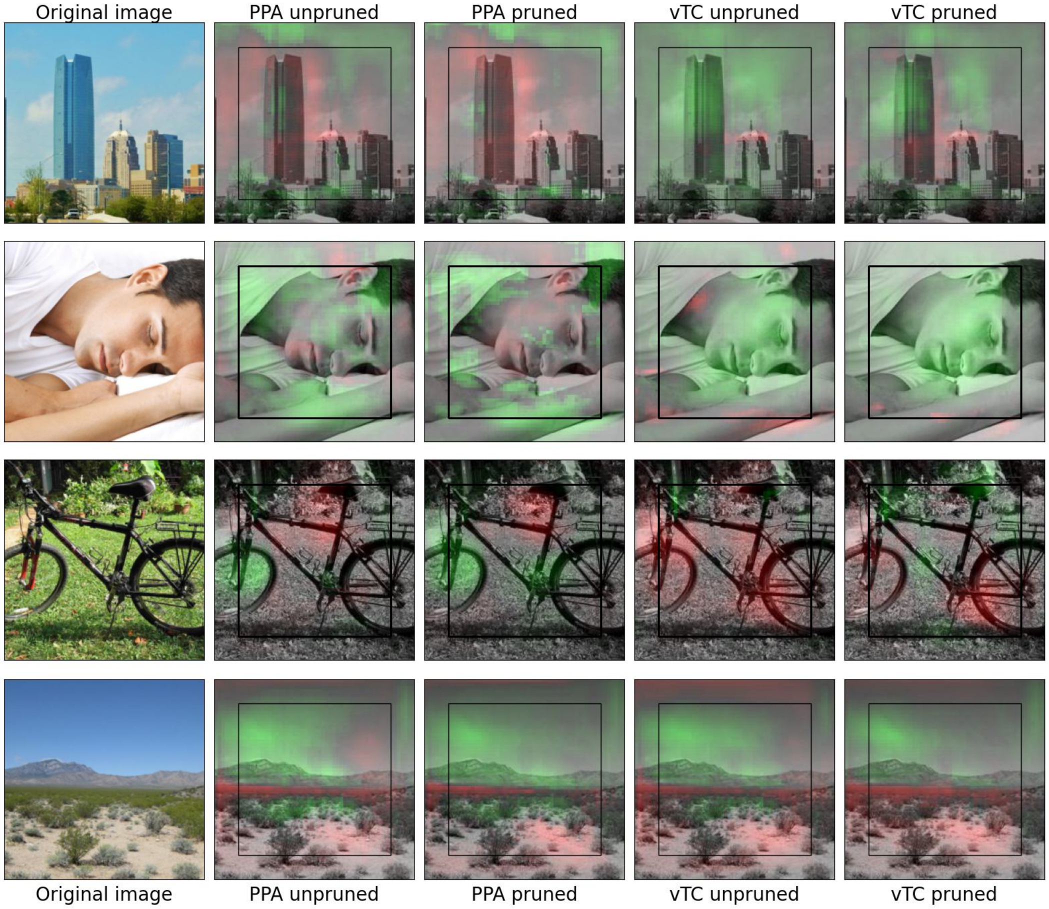


Figure 8: Additional sample heatmaps showing the contribution of each image section to second order isomorphism between a DNN RDM and a Brain RDM. More results can be downloaded from Github: *https://github.com/tlmnhut/Visualize_PrunedDNN_by_HumanSim/tree/main/results/grid*

Acknowledgments and Disclosure of Funding

Scott Fairhall is funded by the European Research Council (ERC) Starting Grant, ‘CRASK’, awarded under the European Union’s Horizon 2020 research and innovation program (Grant Agreement No. 640594).

References

Attarian, M., Roads, B. D., and Mozer, M. C. (2020). Transforming neural network visual representations to predict human judgments of similarity. *arXiv preprint arXiv:2010.06512*.

Cheng, Y., Yu, F. X., Feris, R. S., Kumar, S., Choudhary, A., and Chang, S.-F. (2015). An exploration of parameter redundancy in deep networks with circulant projections. In *Proceedings of the IEEE international conference on computer vision*, pages 2857–2865.

Cichy, R. M. and Kaiser, D. (2019). Deep neural networks as scientific models. *Trends in cognitive sciences*, 23(4):305–317.

Deng, J., Berg, A., Satheesh, S., Su, H., Khosla, A., and Fei-Fei, L. (2012). Ilsvrc-2012, 2012. *URL http://www. image-net. org/challenges/LSVRC*, 3.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books.

Groen, I. I., Greene, M. R., Baldassano, C., Fei-Fei, L., Beck, D. M., and Baker, C. I. (2018). Distinct contributions of functional and deep neural network features to representational similarity of scenes in human brain and behavior. *Elife*, 7:e32962.

Huang, T., Zhen, Z., and Liu, J. (2021). Semantic relatedness emerges in deep convolutional neural networks designed for object recognition. *Frontiers in computational neuroscience*, 15:16.

Jha, A., Peterson, J., and Griffiths, T. L. (2020). Extracting low-dimensional psychological representations from convolutional neural networks. *arXiv preprint arXiv:2005.14363*.

King, M. L., Groen, I. I., Steel, A., Kravitz, D. J., and Baker, C. I. (2019). Similarity judgments and cortical visual responses reflect different properties of object and scene categories in naturalistic images. *NeuroImage*, 197:368–382.

Kriegeskorte, N., Mur, M., and Bandettini, P. A. (2008). Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:4.

Li, X., Wang, Z., Zhang, B., Sun, F., and Hu, X. (2023). Recognizing object by components with human prior knowledge enhances adversarial robustness of deep neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Lindsay, G. W. and Miller, K. D. (2018). How biological attention mechanisms improve task performance in a large-scale visual system model. *ELife*, 7:e38105.

Liu, D., Dai, W., Zhang, H., Jin, X., Cao, J., and Kong, W. (2023). Brain-machine coupled learning method for facial emotion recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Luo, X., Roads, B. D., and Love, B. C. (2021). The costs and benefits of goal-directed attention in deep convolutional neural networks. *Computational Brain & Behavior*, 4(2):213–230.

Mallya, A., Davis, D., and Lazebnik, S. (2018). Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 67–82.

Palazzo, S., Spampinato, C., Kavasidis, I., Giordano, D., Schmidt, J., and Shah, M. (2020). Decoding brain representations by multimodal learning of neural activity and visual features. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):3833–3849.

Peterson, J. C., Abbott, J. T., and Griffiths, T. L. (2018). Evaluating (and improving) the correspondence between deep neural networks and human representations. *Cognitive science*, 42(8):2648–2669.

Richie, R. and Bhatia, S. (2020). Similarity judgment within and across categories: A comprehensive model comparison.

Roads, B. D. and Love, B. C. (2021). Enriching imagenet with human similarity judgments and psychological embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3547–3557.

Sanders, C. A. and Nosofsky, R. M. (2020). Training deep networks to construct a psychological feature space for a natural-object category domain. *Computational Brain & Behavior*, pages 1–23.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Truong, N. and Hasson, U. (2022). Using deep neural networks for modeling representational spaces: the prevalence and impact of rarely-firing nodes. page 249–250.

Wang, J., Wang, L., Nie, F., and Li, X. (2021). Joint feature selection and extraction with sparse unsupervised projection. *IEEE Transactions on Neural Networks and Learning Systems*.