Abstraction Mechanisms Predict Generalization in Deep Neural Networks

Alex Gain [1] **Hava Siegelmann** [2]

Abstract

A longstanding problem for Deep Neural Networks (DNNs) is understanding their puzzling ability to generalize well. We approach this problem through the unconventional angle of *cognitive abstraction mechanisms*, drawing inspiration from recent neuroscience work, allowing us to define the Cognitive Neural Activation metric (CNA) for DNNs, which is the correlation between information complexity (entropy) of given input and the concentration of higher activation values in deeper layers of the network. The CNA is highly predictive of generalization ability, outperforming norm-and-sharpness-based generalization metrics on an extensive evaluation of close to 200 network instances comprising a breadth of dataset-architecture combinations, especially in cases where additive noise is present and/or training labels are corrupted. These strong empirical results show the usefulness of the CNA as a generalization metric and encourage further research on the connection between information complexity and representations in the deeper layers of networks in order to better understand the generalization capabilities of DNNs. [1]

1. Introduction

Deep neural networks (DNNs) have made big strides in recent years, improving substantially on state-of-the-art results across many benchmarks and showing great generalization abilities (LeCun et al., 2015). This is perhaps surprising given the large number of parameters, flexibility, and relative lack of explicit priors enforced in DNNs. Even more puzzling is their ability to memorize random datasets (Zhang et al., 2016; Yun et al., 2019). In part due to this, there has been a surge of work aimed at understanding the crucial factors to high-performing DNNs. Some studies have approached explaining generalization in DNNs via optimization arguments characterizing critical points (Dauphin et al., 2014; Kawaguchi, 2016; Haeffele and Vidal, 2017), the smoothness of loss surfaces (Nguyen and Hein, 2017; Choromanska et al., 2015; Li et al., 2017), and implicit priors and regularization brought on by DNNs' learning methods (Mianjy et al., 2018), including that overparameterization itself leads to better learned optima and easier optimization (Arpit and Bengio, 2019; Allen-Zhu et al., 2019; Oymak and Soltanolkotabi, 2019; Allen-Zhu et al., 2018). The work from (Morcos et al., 2018) provides insight by relating DNN generalization capability to reliance on single directions. Others have used information theory as a means of explaining the perfomance and inner-mechanisms of DNNs (Tishby and Zaslavsky, 2015; Shwartz-Ziv and Tishby, 2017). In (Lampinen and Ganguli, 2018), task structure of DNNs w.r.t. transfer learning is incorporated for tighter bounds on generalization error.

Complementary to these works, we approach the problem of understanding generalization of DNNs via the unconventional angle of applying the cognitive neuroscience concept of abstraction mechanisms (Calvo and Gomila, 2008; Peters et al., 2017; Taylor et al., 2015; Shivhare and Kumar, 2016; Gilead et al., 2014) – i.e. *how representations are formed that compress information content while retaining only information which is relevant for generalization to unseen examples*. Specifically, we seek to analyze DNNs' representational patterns with comparison to abstraction mechanisms employed by the brain. Arguably, if we can quantify representational similarity (and dissimilarity) of these mechanisms in DNNs in comparison to the brain then it may aid in understanding their inner-mechanisms and could allow us to leverage current and ongoing neuroscience research. Additionally, it could bring new perspectives or understanding to the optimization, regularization, and information theory works cited in the previous paragraph.

While we don't fully realize these aims in this paper, we instead provide practical metrics merely inspired by these neuroscience works that could shed light on the generalization mechanisms of DNNs, and provide a potential starting

[1]Department of Computer Science, The Johns Hopkins University, Baltimore, MD 21218, USA [2]School of Computer and Information Sciences, University of Massachusetts Amherst, Amherst, MA 01003, USA. Correspondence to: Alex Gain <again1@jhu.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

[1]An implementation of this paper can be found on GitHub: `https://github.com/alexgain/cna-icml2020`

point for understanding representational similarity between the brain and DNNs from one particular, albeit limited, angle. We emphasize that much further work is needed to establish rigorous causal connections between DNN representations and those of the brain.

We focus on a study particularly amenable to algorithmic translation into conventional statistical learning settings (in our case, visual classification tasks) (Taylor et al., 2015), and we review similar works in the following related works section.

We translate these results into the form of a measure applicable to DNNs, which we term the Cognitive Neural Activation metric (CNA), which is computationally tractable, can be applied to any network architecture and dataset, and is easy to implement. Additionally, the formulation of the CNA is a straightforward formalization of some known results in DNNs, which we motivate in section 3, grounding and clarifying the connection between CNA and generalization to some extent.

The remaining organization of this paper is as follows:

In section 3, we motivate and introduce the precise definitions of the CNA and its modification, the CNA-Margin, which serve as measures of the test error and generalization gap respectively.

In section 4.1 and 4.2, we empirically relate test error and the CNA. We show three main results:

1. The loss landscape of classification error overlaps nicely with the CNA.
2. High entropy images return up to 16 times larger test error across training epochs.
3. Test error significantly correlates with the CNA across a breadth of image datasets and network architectures.

In section 4.3, we show the effectiveness of the CNA-Margin in predicting the *difference* between training and test error (termed the *generalization gap*), and empirically validate its efficacy on a breadth of architectures and datasets, comprising over 200 network instances and a breadth of dataset-architecture combinations. The architectures include Multi-layer perceptrons (MLPs), VGG-18, ResNet-18, and ResNet-101. The datasets include ImageNet-32, CIFAR-10, CIFAR-100, MNIST, Fashion-MNIST, SVHN, corrupted labels counterparts (i.e. the same datasets with varyling levels training labels shuffled), and a random noise dataset. The CNA outperforms recent metrics derived from theoretical generalization error bounds, especially in non-standard settings, showing significantly more robustness to corrupted labels and the random noise dataset.

Contributions Summary: We approach generalization in DNNs from an unconventional angle, where we connect abstractness mechanisms in the human brain to generalization error in DNNs in statistical learning settings. We explicitly translate a neuroscience result into a computationally tractable, differentiable mathematical expression, termed the Cognitive Neural Activation metric (CNA), that is easily implementable and can be applied to any network architecture. The CNA shows interesting connections to test error, and can be adapted such that it is predictive of generalization error in DNNs, outperforming recent generalization metrics based on theoretical generalization error bounds, and shows significantly more robustness to additive noise and label corruption. The strong emprical results encourage further work and exploration into this area of study.

2. Related Work

Comparisons of DNNs to cognitive neuroscience are often qualitative in nature or serve as loose analogies for illustration purposes only, e.g. historically much of the design of MLPs and CNNs were loosely motivated by computational neuroscience. Rigorous or empirical analyses are less common, however important progress has been made in characterizing representational similarity between DNNs and the brain. Seminal works (Yamins et al., 2013; 2014) show significant similarity between the firing patterns of the visual cortex of primates and supervised models, as does the seminal work from (Khaligh-Razavi and Kriegeskorte, 2014) though with the caveat that unsupervised models do not.

Recently, the Brain-Score has been developed (Schrimpf et al., 2018; Kubilius et al., 2019) which uses the mean-score of neural predictivity (how well a given DNN predicts a single neuron's response in various visual systems of the primate) and behavioral similarity (how similar rates of correct and incorrect responses for specific inputs are between a given DNN and the primate) to rank the top-1 performance of state-of-the-art networks on ImageNet. The Brain-Score achieved significant correlation with top-performing networks' performance on ImageNet, showing neural representational similarity between the brain and DNNs can have useful predictive properties and, additionally, developed a high-performing shallow RNN based on it.

The CNA mainly differs from the Brain-Score in that the CNA's primary application is in understanding and measuring the generalization gap between train and test sets across many different datasets and tasks, as opposed to ranking top-performing networks on ImageNet. Additionally, it is a differentiable geometric property or equation that, though grounded in empirical neuroscience data, does not actually use neuroscience data in its computation, i.e. it is a more basic function of a DNN's activation distribution and training distribution, not requiring empirical neuroscience data dur-

ing inference at training or test. Thus, it is closer in nature to statistical-learning-based bounds on the generalization gap (Neyshabur et al., 2017a) and information-theoretic loss functions, e.g. the mutual information objective from Deep InfoMax (Hjelm et al., 2018).

Other related works utilizing cognitive neural activity for practical application include (Shen et al., 2019), which used DNNs to reconstruct accurate, realistic-looking images from fMRI data, (Xu et al., 2018), which made use of empirical neuronal firing data to develop methods for explainable interpretability of DNNs, and (Arend et al., 2018), which show that many one-to-one mappings exist between individual neurons in DNNs and individual neurons in the brain as well as correspondences between population-level groups. In (Saxe et al., 2019), they study the learning dynamics of linear networks and show that the learned representations share phemenoma observed in human semantic development. Lastly, in (Richards et al., 2019), they give strong arguments for shifting the research paradigm of computational neuroscience towards utilizing three essential design components of DNNs: The objective functions, the learning rules, and the architectures.

Besides cognitive neuroscience work, the CNA primarily acts as a stand-in for margin-and-norm-based generalization metrics. We cover this in more detail in our main empirical results section.

3. The Cognitive Neural Activation Metric

The work from (Taylor et al., 2015) demonstrates a relationship between the use of neurons *deeper* in the brain – where *deeper* is defined as farther from sensory cortices – and the abstractness of tasks performed by humans, as measured by Amazon Mechanical Turk surveys. In tasks that are more abstract, e.g. mathematical reasoning, relatively higher activation is seen in the deeper neurons of the brain, in contrast to tasks that are less abstract, e.g. finger-tapping, where relatively higher activation is seen in neurons closer to the sensory cortices. If we consider task abstractness and use of deeper neurons as random variables α and β respectively, then this relationship can be succinctly stated as $\alpha \propto \beta$.

This (α, β) framing can be similarly applied to a conceptually-related phenomon seen in DNNs: That deeper networks are needed for harder tasks and better generalization. For example, in (Huang et al., 2017) and (Dehghani et al., 2018) results show this empirically for feedforward networks for vision and in recurrent architectures respectively. It is argued that one of the primary reasons for the large success of ResNets (Szegedy et al., 2017) is that skip connections and batch normalization allow for stable training of very deep networks (Orhan and Pitkow, 2017; Balduzzi et al., 2017; Santurkar et al., 2018). This empirical phenomenon can be formalized in the following manner:

For a given input x and network f_θ, with parameter assignments θ, let $\alpha(x)$ denote the difficulty of input x and $\beta(x; f_\theta)$ denote the quantified use of deep representations for network f_θ on input x. Then, for modern, well-trained networks that generalize well, we hypothesize a tight relationship between α and β for the data distribution trained on. Further, we hypothesize that some measure of this tightness, denoted $\rho_{\alpha,\beta}$, will correlate with generalization capability across networks.

To test these hypotheses, we must define α and β such that they capture the notion of input difficulty and the use of deep representations.

3.1. Quantifying Input Complexity

For quantification of input difficulty, we propose *compressibility*, or input complexity, as a suitable metric. Input complexity was used in (Serrà et al., 2019) for a different context where GANs perform out-of-distribution detection, and it was shown that input complexity highly correlated with the likelihood ratios. The input complexity of an image was calculated as its compression ratio given a compression algorithm. This relates to (Taylor et al., 2015) as well, where "abstractness" of a task was defined as

> "A process of creating general concepts or representations ... often with the goal of compressing the information content ... and retaining only information which is relevant."

Rather than use compression algorithms, we exploit the fact that Shannon entropy is a lower bound of Kolmogorov complexity and instead make use of Shannon entropy estimation via histogram binning, which is somewhat more convenient implementation-wise. Although we did not perform extensive comparisons, our preliminary experiments suggest that using compression algorithms for calculation of input complexity would not significantly change the results of this paper, at least for MLP networks.

We restrict the tasks in this paper to image classifications. For a single image x, the Shannon entropy of the pixel values, i.e. treating the pixel value as a random variable, can be estimated with histrogram binning via:

$$\alpha(x; B) \triangleq -\sum_{i=1}^{B} p_{bin_i} \log p_{bin_i} \tag{1}$$

for B bins where p_{bin_i} is proportional to the frequency of pixel values in the range defined by bin i for input x. For all datasets in this paper, we did not observe much sensitivity to bin size, e.g. $\forall B \geq 100$, the α values' correlation was greater than 0.98. For all experiments in this paper, we

calculated α before applying image augmentation or input normalization.

There are potential issues with compression as a metric for input complexity, which are also discussed to an extent in (Serrà et al., 2019). For example, random noise will have a higher α value than any natural image, but that does not correspond to the intuitive notion of complexity. Nonetheless, α as defined ended up being practical for the purposes of this paper. Improvements to the definition of α could certainly be worth exploring in future work.

Later, in section 3, we see empirical validation of α as a measure of input difficulty.

3.2. Defining the CNA

With a suitable definition of α, we now turn towards quantifying the use of deep representations, and lastly the CNA definition itself. Following (Taylor et al., 2015), we use the linear regressed slope of the activation values versus depth for β. Precisely stated, $\beta(x)$ is defined, for a given input x and network, as the linear regressed slope of points $(d, z_d(x))$, where $d = 1, \ldots, L$ for an L layer network, and $z_d(x)$ is the sum of pre-activation values for layer d of the network.

As defined, β is not intended to model the relationship between depth and activation since it is likely a non-linear relationship. Instead, this merely serves as a coarse measure for the use of deeper activations in any given network. For example, if the activation values for the last couple layers of a network are abitrarily increased, β would also increase. Lastly, this definition is general to most feedforward architetures, e.g. MLP, VGG, ResNet, DenseNet, and so on.

Finally, we define the Cognitive Neural Activation Metric for a batch of datapoints $\boldsymbol{X}$ and given network as the Pearson correlation between α and β, i.e.

$$CNA(\boldsymbol{X}) \triangleq \frac{\text{cov}(\boldsymbol{\alpha}, \boldsymbol{\beta})}{\sigma_\alpha \sigma_\beta} \tag{2}$$

where cov denotes covariance, σ_α and σ_β denote the standard deviations of α and β, and $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ denote the vectors of α values and β values respectively for the batch of datapoints $\boldsymbol{X}$.

For a clear illustration of the main principles of the CNA, see Figure 1, which shows the relative activation values of an MLP network trained on MNIST by depth for a particularly low-complexity input (the digit 1) and a high-complexity input (the digit 8).

For an overview of the definition of the CNA, see Figure 2

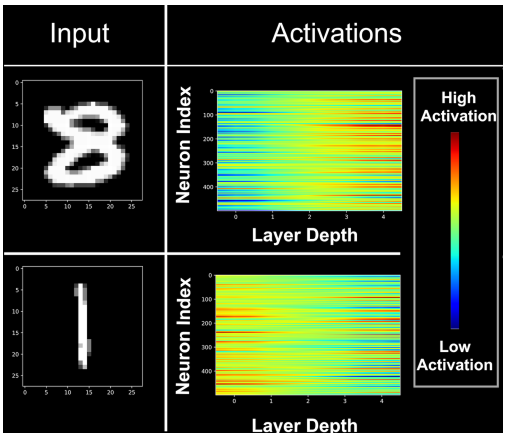

Figure 1. The activation values of a network trained on MNIST plotted against depth for a particularly low-complexity input (the digit 1) and a particularly high complexity input (the digit 8). There are relatively higher activations in the deeper layers of the network for the digit 8. Networks that generalize well exhibit this property, as we see in section 4 in more extensive experiments.

3.3. The CNA-Margin For Generalization Gap Prediction

We next turn our attention to the CNA-Margin, a modification of the CNA so that it is predictive of the *generalization gap*, or the difference between performance on the training and test sets for a given network.

Recall that the CNA is defined as the Pearson correlation between α and β for a given dataset X, denoted as $corr(\boldsymbol{\alpha}(X), \boldsymbol{\beta}(X))$. For networks that perform well on their test sets, as seen in section 4, this correlation is significantly positive, indicating a close relationship between $\boldsymbol{\alpha}(X)$ and $\boldsymbol{\beta}(X)$. For networks that perform poorly, the CNA is of much lower magnitude. However, this says nothing about whether a network's output distribution significantly differs between training and test instances: For example, for a CNA value of zero, no relationship between $\boldsymbol{\alpha}(X)$ and $\boldsymbol{\beta}(X)$ exists but, nonetheless, the loss on both the training and test set could be very similar, resulting in a small generalization gap.

Thus, it is necessary to consider the relationship between $\boldsymbol{\alpha}(X)$ and $\boldsymbol{\beta}(X)$ on both the training and test set. If the distribution significantly differs on the training and test sets, we would expect the distribution of $\boldsymbol{\beta}(X)$ to change as well, altering the relationship.

We define the (α, β)-curve of a dataset X as the set of tuples

$$\{(\alpha(x), \beta(x)) \mid x \in X\} \tag{4}$$

The generalization gap would then be reflected by the difference in the (α, β)-curves of the training and test sets. This is illustrated in Figure 3 for an MLP network trained on

Defining the CNA

For a network architecture A and dataset X with n data points, define

1. $\alpha(x)$ – the input complexity (computed via histogram-binning approximation of Shannon entropy) of every datapoint $x \in X$,
2. $\beta(x)$ – the slope of neuronal activity of network A when presented with $x \in X$,
3. $\boldsymbol{\alpha}, \boldsymbol{\beta}$ – the vectors of length n comprising the complexity and slope values on the whole dataset X.

The CNA is defined by the Pearson correlation between the information complexity and the slope:

$$\rho_{\alpha,\beta} = \frac{\text{cov}(\boldsymbol{\alpha}, \boldsymbol{\beta})}{\sigma_\alpha \sigma_\beta} \tag{3}$$

where $\text{cov}(\boldsymbol{\alpha}, \boldsymbol{\beta})$ is the sample covariance of the two vectors: $\frac{1}{n-1}\sum_i(\boldsymbol{\alpha}_i - \overline{\boldsymbol{\alpha}})(\boldsymbol{\beta}_i - \overline{\boldsymbol{\beta}})$, $\overline{\boldsymbol{\alpha}}$ and $\overline{\boldsymbol{\beta}}$ are the means, and σ_α, and σ_β are the sample standard deviations $\frac{1}{n-1}\sum_i(\boldsymbol{\alpha}_i - \overline{\boldsymbol{\alpha}})^2$ and $\frac{1}{n-1}\sum_i(\boldsymbol{\beta}_i - \overline{\boldsymbol{\beta}})^2$.

Figure 2. **CNA**: A high-level overview of the expressions comprising the CNA.

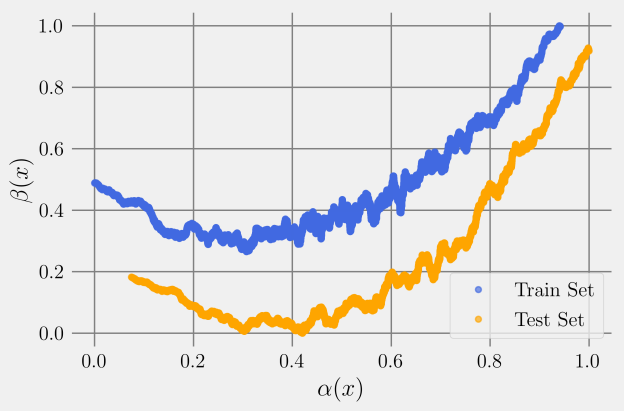

Figure 3. Plot of the (α, β) values of an MLP trained on SVHN. The linearity of the curves correlates well with test performance, and the gap between the two curves correlates well with train-test loss gap.

SVHN.

Quantifying this difference, we define the estimated area between the slope-entropy curves for a given training set X_{train} and test set X_{test} as the maximum area over the set of polygons that can be inscribed between the slope-entropy curves of X_{train} and X_{test}, i.e.

$$CNA_{\mathcal{A}}(X_{\text{train}}, X_{\text{test}}) \triangleq \max_{P \in S} A(P) \tag{5}$$

where $A(P)$ denotes the area of a polygon P, and S denotes the set of polygons that can be inscribed between the curves.

As was done for the generalization metrics in (Neyshabur et al., 2017a), we scale $CNA_{\mathcal{A}}(X_{\text{train}}, X_{\text{test}})$ by the margin of the network. The margin is defined as the minimum distance to a decision boundary(Elsayed et al., 2018) for a given network. To define this in closed-form, consider a classification setting with H classes and a network f with an output layer of size H. For a single datapoint x, the network output $f(x) \in [0,1]^H$ is a vector of probabilities with each index $f(x)[j]$ denoting the probability, estimated by f, that x belongs to class j. The margin is defined, for a single datapoint x and network as

$$\gamma \triangleq f(x)[j_{\text{true}}] - \max_{j \neq j_{\text{true}}} f(x)[j] \tag{6}$$

where j_{true} denotes the correct groundtruth class that x belongs to. In practice, for computational tractability, the margin of a network is taken to be the maximum γ over a set of datapoints, typically between 1% to 10% of the training or validation datapoints, which we denote as γ_{margin}. We then arrive at our final generalization gap metric, termed the CNA-Margin, and denoted as $CNA_{\mathcal{M}}$:

$$CNA_{\mathcal{M}}(X_{\text{train}}, X_{\text{test}}) \triangleq \gamma_{\text{margin}} \cdot CNA_{\mathcal{A}}(X_{\text{train}}, X_{\text{test}}) \tag{7}$$

The $CNA_{\mathcal{M}}$ is an appropriate metric for measuring the generalization gap: Its form is close to that of state-of-the-art norm-and-sharpness-based metrics (Jiang et al., 2019) and achieves strong empirical results, as shown in the next section.

4. Experimental Results

In this section, we show empirical validation of the CNA as a useful measure for understanding training in DNNs and

their generalization error.

In sections 4.1 and 4.2, we show through empirical experiments and visualization of the loss landscape that there is a close connection between information complexity, CNA, and proper training of DNNs.

In section 4.3, we show empirical validation of the CNA-Margin via extensive experiments, showing it outperforms competitive generalization metrics in a range of settings.

4.1. How Does CNA Vary During Training?

Intuitively, the CNA is a measure of how similar a given DNNs' abstraction mechanisms are to that of the human brain on a given dataset. If a DNN generalizes well and has "learned" high-level concepts, it is conceivable that its CNA value, then, would be high. On the other hand, if the mechanisms by which DNNs abstract show no similarity to that of the brain, then we would expect to see no relationship between performance on a task and the CNA value.

As a first step to investigating this, we train a simple MLP on the MNIST dataset and track the CNA value over training time. This is seen in Figure 4. From the figure, the CNA clearly tracks training loss well, suggesting that the DNN is learning specific abstraction mechanisms over training time. Another interpretation of this result is simply that the loss landscape of the CNA and the supervised loss function (in this case, categorical cross entropy) are similar. If the gradient of the CNA function and the gradient of the supervised loss function are well-aligned throughout training time, then there is a significant chance for the CNA and the loss to be correlated. To investigate this, we record all neuronal activation values of the MLP at each minibatch update. We then perform PCA on the recorded neuronal activations values in order to visualize the optimization path of the network over training time, plotting the network state as a function of its principal components. Then, we sample from the x-y plane of principal component values and calculate the CNA value via the inverse PCA transformation, recovering the network state value corresponding to some representative minibatch for that particular network state. This allows us to approximate and visualize the CNA loss landscape in the lower-dimensional space. The network converged to around 98% test accuracy, showing that the high CNA value may be indicative of high test accuracy.

These results and visualization can be seen in Figure 5.

The results of Figure 5 are perhaps surprising given that the CNA does not depend on labels, whereas the classification objective does. How, then, would the gradients of these two very different objectives be well aligned? A cursory look into the gradient expressions of the two terms gives a possible explanation, which we leave in the supplement in the interest of space. In short, the CNA and the supervised

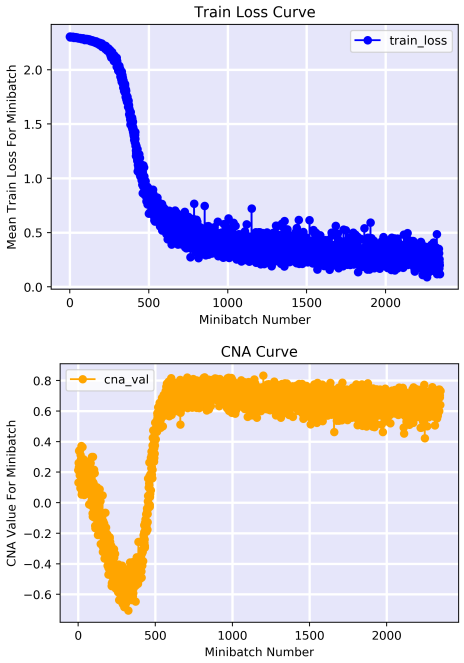

Figure 4. A simple MLP trained on MNIST with curves the training loss values (Top) and CNA values (Bottom) over training time. It is clear that the CNA shows a high correlation with training loss, with inflection points of both curves occuring at roughly the same timestep.

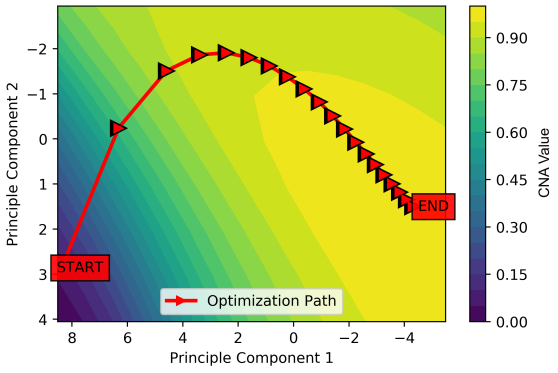

Figure 5. A low-dimensional visualization of the optimization path of an MLP over training time on MNIST, showing the network approximately traverses the CNA loss surface, despite being trained only on classification loss. The network state (all neuronal activation values) was recorded during each training step and then visualized using PCA (the red curve) with "Start" and "End" denoting the start and end points of training, with the x-axis and y-axis corresponding to the principal component values. The contour map behind the red curve (i.e. the CNA loss surface) was generated via sampling from the 2D principal component space and calculating the CNA value at each point. Best viewed in color.

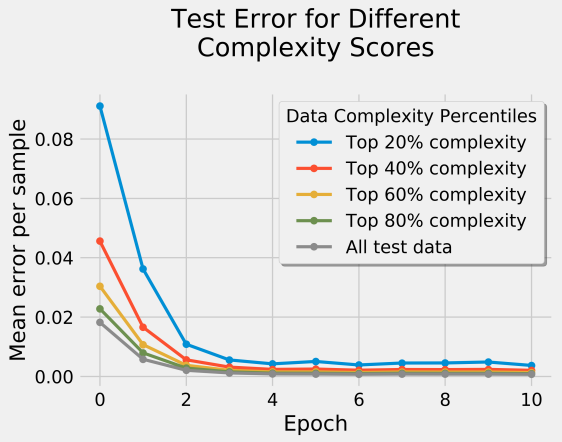

Figure 6. A plot of the mean test error across training time for an MLP trained on MNIST. Bins of datapoints with varying levels of input complexity are shown. There is a clear monotonic relationship between test error and complexity, especially at the beginning of training time.

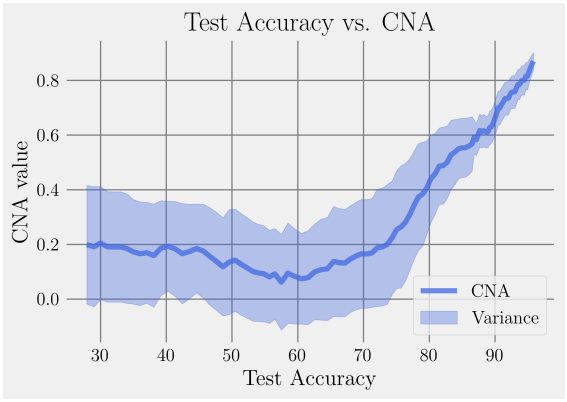

Figure 7. The CNA strongly correlates with test accuracy. Here we show networks grouped according to test accuracy across six different datasets (ImageNet-32, CIFAR-10, CIFAR-100, SVHN, MNIST, Fashion-MNIST), four different architectures (MLP, VGG-18, ResNet-18, and ResNet-101), and measured at multiple stages of training (every 20 epochs) – this makes for a total of 147 network instances. CNA correlates significantly with test accuracy, with a nearly linear relationship at greater than 70%, suggesting neural activation properties of DNNs become more similar to the brain as classification results improve.

loss function could be well aligned in some cases where the error terms and information complexity terms for given datapoints significantly correlate.

To give further credence to this explanation, we empirically analyze the relationship between the test error, which we denote as ε, and α. We look at the mean error of the network on data points of varying α values to see if there is any relationship. Should the gradient argument hold true, we should expect that high-complexity datapoints will show larger error. This analysis is shown in Figure 6. We bin datapoints by their α values and plot their mean test error over training time, e.g. the blue curve corresponds to the datapoints with the top 20% complexity values.

Interestingly the figure shows a very clear, monotonic relationship between α and ε, especially in the beginning of training time. The differences between ε at different complexity levels quickly decreases, though maintains its ordering, as training time increases. This makes sense given the network decreases its loss, and increases its CNA value, the most during the beginning epochs as was seen in Figure 4.

In summary, these results and figures show an interesting relationship between complexity of datapoints, CNA, training of DNNs, and test performance. These experiments by no means warrant broad conclusions – more extensive analysis would be needed to draw more certain conclusions. Nonetheless, these raise questions as to how close the relationship is between CNA and training in DNNs, and whether there is a tight causal relationship between abstraction mechanisms and training.

4.2. Extensive Evaluation of CNA and Test Performance

We now carry out a far more extensive analysis of the CNA and test performance. We train four different architectures (MLP, VGG-18, ResNet-18, and ResNet-101) across six different datasets (MNIST, Fashion-MNIST, SVHN, CIFAR-10, CIFAR-100, and ImageNet) each, recording the network state, test error, and CNA value at every 20th epoch, comprising over 100 dataset-architecture combinations. These results are shown in Figure 7. There is a high correlation between CNA and test accuracy, suggesting close causal relationship between abstraction mechanisms and generalization ability. The result is especially convincing given the broad range of architectures and datasets tested on.

This extensive evaluation gives more credence to the hypothesis that, as was seen in the previous subsection, the CNA, training in DNNs, and generalization in DNNs have an important relationship that warrants further study.

4.3. Extensive Evaluation of CNA and the Generalization Gap

Much work has been done on generalization bounds for DNNs based on norm and spectral properties of the weights (Neyshabur et al., 2017b;a). Others include (Arora et al., 2018), which give bounds for DNNs based on compression properties, and (Neyshabur et al., 2018), which give bounds based on overparameterization of DNNs. In (Jiang et al., 2018), a margin-based metric is developed and shows great

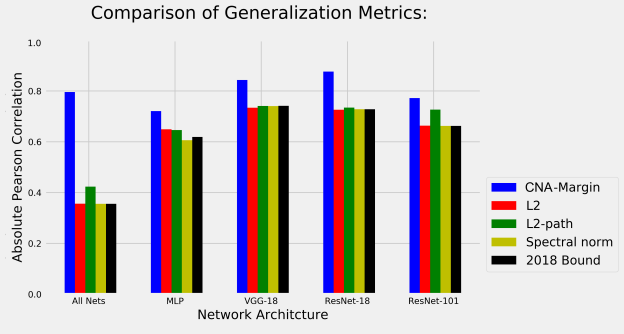

Figure 8. We show the Pearson correlation of various sharpness-based generalization metrics with the train-test generalization gaps of 147 combinations of networks. In total: Datasets (ImageNet, CIFAR-10, CIFAR-100, SVHN, MNIST, Fashion-MNIST) and network architectures (MLP, VGG-18, ResNet-18, ResNet-100), which were analyzed every 20 training epochs. We show the correlation conditional on network architecture as well as for all networks in aggregate ("All Nets").

success in correlating with the generalization gap, although with the drawback that a linear model needs to be fit between the generalization gap and margin parameters for each individual network and dataset.

As empirical validation of CNA, we focus on the very general setting where any network architecture, task, and dataset is allowed, and no model fitting with respect to the generalization gap is done, i.e. the developed metric must be predictive *a priori*.

To this end, we make use of the same 147 networks shown in Figure 7, and evaluate them on the CNA-Margin and the competitive generalization as defined in (Neyshabur et al., 2017a) and (Jiang et al., 2019). Specifically, we record the generalization gap (the difference between train and test accuracy), each generalization metric for all networks, and calculate the Pearson correlation between each metric and the generalization gap. The correlation is shown for all metrics, for each architecture, and for all architectures in aggregate (denoted "All Nets"). This is shown in Figure 8. The CNA-Margin outperforms all metrics especially when considering all architectures in aggregate.

We additionally include a a Gaussian noise dataset. Each point in this dataset is drawn from the standard normal distribution of shape $3 \times 32 \times 32$ and labeled randomly to one of 10 classes – the networks then memorize the training set. The norm-based metrics perform very poorly on this Gaussian noise dataset, whereas the CNA-Margin remains comparatively robust. These results are shown in In Figure 9A.

Lastly, we train a subset of the networks on the same datasets, except with a varying degree of shuffled labels, ranging from 10% to 50% labels shuffled during training time. Similar to Figure 9A, the CNA-Margin remains robust compared to other metrics. All training details are included in the supplement.

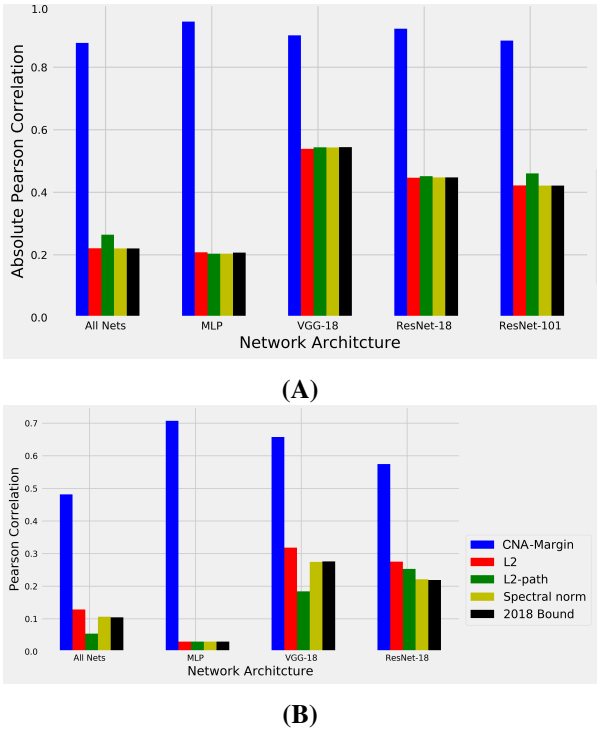

Figure 9. The CNA is comparatively robust to various types of corruption including in **(A)** where a random Gaussian noise dataset is included (detailed in the main text and supplement), and **(B)** where varying degrees of label corruption is present. Each subfigure comprises close to 200 network instances.

5. Conclusion

We provide principled motivation for a generalization metric inspired by cognitive neuroscience results. Interestingly, and perhaps suprisingly, the CNA shows connections with and predictive power for task performance in DNNs across a wide range of scenarios. Our CNA formulations show a practical use-case in predicting the generalization gap, outperforming margin-and-norm-based metrics, especially in the presence of dataset corruption. To our knowledge, our results comprise the first precise empirical formalization the notion that deeper representations are required to classify to difficulty inputs. Through both small-scale and large-scale experiments, we show strong empirical support for the value of future work on understanding the relationship between abstract mechanisms, input complexity, and generalization capabilities in DNNs.

References

Z. Allen-Zhu, Y. Li, and Z. Song. A convergence theory for deep learning via over-parameterization. *arXiv preprint arXiv:1811.03962*, 2018.

Z. Allen-Zhu, Y. Li, and Y. Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. In *Advances in neural information processing systems*, pages 6155–6166, 2019.

L. Arend, Y. Han, M. Schrimpf, P. Bashivan, K. Kar, T. Poggio, J. J. DiCarlo, and X. Boix. Single units in a deep neural network functionally correspond with neurons in the brain: preliminary results. Technical report, Center for Brains, Minds and Machines (CBMM), 2018.

S. Arora, R. Ge, B. Neyshabur, and Y. Zhang. Stronger generalization bounds for deep nets via a compression approach. *arXiv preprint arXiv:1802.05296*, 2018.

D. Arpit and Y. Bengio. The benefits of over-parameterization at initialization in deep relu networks. *arXiv preprint arXiv:1901.03611*, 2019.

D. Balduzzi, M. Frean, L. Leary, J. Lewis, K. W.-D. Ma, and B. McWilliams. The shattered gradients problem: If resnets are the answer, then what is the question? *arXiv preprint arXiv:1702.08591*, 2017.

P. Calvo and T. Gomila. *Handbook of cognitive science: An embodied approach*. Elsevier, 2008.

A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun. The loss surfaces of multilayer networks. In *Artificial Intelligence and Statistics*, pages 192–204, 2015.

Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Advances in neural information processing systems*, pages 2933–2941, 2014.

M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and Ł. Kaiser. Universal transformers. *arXiv preprint arXiv:1807.03819*, 2018.

G. Elsayed, D. Krishnan, H. Mobahi, K. Regan, and S. Bengio. Large margin deep networks for classification. In *Advances in neural information processing systems*, pages 842–852, 2018.

M. Gilead, N. Liberman, and A. Maril. From mind to matter: neural correlates of abstract and concrete mindsets. *Social Cognitive and Affective Neuroscience*, 9(5):638–645, 2014.

B. D. Haeffele and R. Vidal. Global optimality in neural network training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7331–7339, 2017.

R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

G. Huang, D. Chen, T. Li, F. Wu, L. van der Maaten, and K. Q. Weinberger. Multi-scale dense networks for resource efficient image classification. *arXiv preprint arXiv:1703.09844*, 2017.

Y. Jiang, D. Krishnan, H. Mobahi, and S. Bengio. Predicting the generalization gap in deep networks with margin distributions. *arXiv preprint arXiv:1810.00113*, 2018.

Y. Jiang, B. Neyshabur, H. Mobahi, D. Krishnan, and S. Bengio. Fantastic generalization measures and where to find them. *arXiv preprint arXiv:1912.02178*, 2019.

K. Kawaguchi. Deep learning without poor local minima. In *Advances in Neural Information Processing Systems*, pages 586–594, 2016.

S.-M. Khaligh-Razavi and N. Kriegeskorte. Deep supervised, but not unsupervised, models may explain it cortical representation. *PLoS computational biology*, 10(11), 2014.

J. Kubilius, M. Schrimpf, K. Kar, R. Rajalingham, H. Hong, N. Majaj, E. Issa, P. Bashivan, J. Prescott-Roy, K. Schmidt, et al. Brain-like object recognition with high-performing shallow recurrent anns. In *Advances in Neural Information Processing Systems*, pages 12785–12796, 2019.

A. K. Lampinen and S. Ganguli. An analytic theory of generalization dynamics and transfer learning in deep linear networks. *arXiv preprint arXiv:1809.10374*, 2018.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436, 2015.

H. Li, Z. Xu, G. Taylor, and T. Goldstein. Visualizing the loss landscape of neural nets. *arXiv preprint arXiv:1712.09913*, 2017.

P. Mianjy, R. Arora, and R. Vidal. On the implicit bias of dropout. *arXiv preprint arXiv:1806.09777*, 2018.

A. S. Morcos, D. G. Barrett, N. C. Rabinowitz, and M. Botvinick. On the importance of single directions for generalization. *arXiv preprint arXiv:1803.06959*, 2018.

B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems*, pages 5947–5956, 2017a.

B. Neyshabur, S. Bhojanapalli, and N. Srebro. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017b.

B. Neyshabur, Z. Li, S. Bhojanapalli, Y. LeCun, and N. Srebro. Towards understanding the role of over-parametrization in generalization of neural networks. *arXiv preprint arXiv:1805.12076*, 2018.

Q. Nguyen and M. Hein. The loss surface of deep and wide neural networks. *arXiv preprint arXiv:1704.08045*, 2017.

A. E. Orhan and X. Pitkow. Skip connections eliminate singularities. *arXiv preprint arXiv:1701.09175*, 2017.

S. Oymak and M. Soltanolkotabi. Towards moderate overparameterization: global convergence guarantees for training shallow neural networks. *arXiv preprint arXiv:1902.04674*, 2019.

J. F. Peters, A. Tozzi, S. Ramanna, and E. İnan. The human brain from above: an increase in complexity from environmental stimuli to abstractions. *Cognitive neurodynamics*, 11(4):391–394, 2017.

B. A. Richards, T. P. Lillicrap, P. Beaudoin, Y. Bengio, R. Bogacz, A. Christensen, C. Clopath, R. P. Costa, A. de Berker, S. Ganguli, et al. A deep learning framework for neuroscience. *Nature neuroscience*, 22(11): 1761–1770, 2019.

S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry. How does batch normalization help optimization? In *Advances in Neural Information Processing Systems*, pages 2483–2493, 2018.

A. M. Saxe, J. L. McClelland, and S. Ganguli. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116(23):11537–11546, 2019.

M. Schrimpf, J. Kubilius, H. Hong, N. J. Majaj, R. Rajalingham, E. B. Issa, K. Kar, P. Bashivan, J. Prescott-Roy, K. Schmidt, et al. Brain-score: Which artificial neural network for object recognition is most brain-like? *BioRxiv*, page 407007, 2018.

J. Serrà, D. Álvarez, V. Gómez, O. Slizovskaia, J. F. Núñez, and J. Luque. Input complexity and out-of-distribution detection with likelihood-based generative models. *arXiv preprint arXiv:1909.11480*, 2019.

G. Shen, T. Horikawa, K. Majima, and Y. Kamitani. Deep image reconstruction from human brain activity. *PLoS computational biology*, 15(1):e1006633, 2019.

R. Shivhare and C. A. Kumar. On the cognitive process of abstraction. *Procedia-Procedia Computer Science*, 89: 243–252, 2016.

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

C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

P. Taylor, J. Hobbs, J. Burroni, and H. Siegelmann. The global landscape of cognition: hierarchical aggregation as an organizational principle of human cortical networks and functions. *Scientific reports*, 5(1):1–18, 2015.

N. Tishby and N. Zaslavsky. Deep learning and the information bottleneck principle. In *Information Theory Workshop (ITW), 2015 IEEE*, pages 1–5. IEEE, 2015.

T. Xu, J. Zhan, O. G. Garrod, P. H. Torr, S.-C. Zhu, R. A. Ince, and P. G. Schyns. Deeper interpretability of deep networks. *arXiv preprint arXiv:1811.07807*, 2018.

D. L. Yamins, H. Hong, C. Cadieu, and J. J. DiCarlo. Hierarchical modular optimization of convolutional networks achieves representations similar to macaque it and human ventral stream. In *Advances in neural information processing systems*, pages 3093–3101, 2013.

D. L. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.

C. Yun, S. Sra, and A. Jadbabaie. Small relu networks are powerful memorizers: a tight analysis of memorization capacity. In *Advances in Neural Information Processing Systems*, pages 15532–15543, 2019.

C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.