Explaining Generalization Power of a DNN Using Interactive Concepts

Huilin Zhou[1], Hao Zhang[1], Huiqi Deng[1], Dongrui Liu[1], Wen Shen[1], Shih-Han Chan[1, 2], Quanshi Zhang[1*]

[1]Shanghai Jiao Tong University
[2]University of California San Diego
{zhouhuilin116, 1603023-zh, denghq7, drliu96, wen_shen}@sjtu.edu.cn
s2chan@ucsd.edu, zqs1022@sjtu.edu.cn

Abstract

This paper explains the generalization power of a deep neural network (DNN) from the perspective of interactions. Although there is no universally accepted definition of the concepts encoded by a DNN, the sparsity of interactions in a DNN has been proved, *i.e.*, the output score of a DNN can be well explained by a small number of interactions between input variables. In this way, to some extent, we can consider such interactions as interactive concepts encoded by the DNN. Therefore, in this paper, we derive an analytic explanation of inconsistency of concepts of different complexities. This may shed new lights on using the generalization power of concepts to explain the generalization power of the entire DNN. Besides, we discover that the DNN with stronger generalization power usually learns simple concepts more quickly and encodes fewer complex concepts. We also discover the detouring dynamics of learning complex concepts, which explains both the high learning difficulty and the low generalization power of complex concepts. The code will be released when the paper is accepted.

Introduction

Although deep neural networks (DNNs) have achieved remarkable success nowadays, the essence for the superior generalization power of a DNN is still unclear. People usually explained DNNs via the flatness of the loss landscape (Keskar et al. 2016) and theoretical bounds for the generalization (Dziugaite and Roy 2017; Neyshabur, Tomioka, and Srebro 2015), or by proposing new metrics for the representation power (Fort et al. 2019; Weng et al. 2018). In recent years, analyzing the capacity or blind spots of encoding specific concepts represents an emerging direction in explaining DNNs (Deng et al. 2021; Cheng et al. 2021a).

Therefore, unlike previous studies, we revisit the generalization of a DNN from a new perspective of concepts. If the inference score of a DNN can be attributed to a set of countable concepts, then the generalization power of a DNN would be explained by the generalization power of elementary concepts encoded by the DNN. This will be a new insight into the generalization power of the DNN.

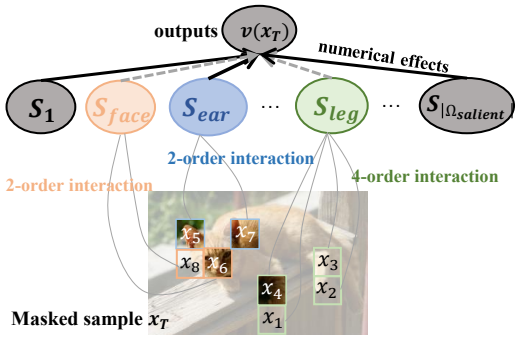

Figure 1: Interactions encoded by the DNN. Each interaction S represents an AND relationship between a set of input variables (*e.g.*, image regions). Masking any patches in S_{face} will deactivate the interaction, making $I(S_{\text{face}}|\boldsymbol{x}) = 0$.

Can we really define concepts encoded by a DNN? How to define concepts encoded by a DNN is still an open problem, and there is no formal and universally-accepted definition so far. To this end, given a trained DNN, Ren et al. (2023a) quantified Harsanyi interactions (Harsanyi 1963) encoded by a DNN as concepts in the DNN. Each Harsanyi interaction represents an AND relationship between a set of input variables. For example, in the natural language processing, a Harsanyi interaction may represent the AND relationship between words. In the face detection, as Figure 1 shows, a Harsanyi interaction may consist of patches of $S = \{\text{eyes}, \text{nose}, \text{mouth}\}$ to represent the AND relationship between image patches for the face. Only if all patches in S are present in the image, the face interaction S is activated and makes a numerical effect $I(S|\boldsymbol{x})$ on the detection score.

Although Ren et al. (2023a) did not convince us that the above interaction really represented a concept that fits human cognition, they did provide mathematical supports for such interactions. (1) Li and Zhang (2023); Ren et al.

*Quanshi Zhang is the corresponding author. He is with the Department of Computer Science and Engineering, the John Hopcroft Center, at the Shanghai Jiao Tong University, China. Correspondence to: Quanshi Zhang < zqs1022@sjtu.edu.cn>.

(2023a) discovered and Ren et al. (2023b) mathematically proved that a well-trained DNN usually encoded just a few interactions between different input variables. (2) We can use these interactions to explain the output of a DNN. (3) Besides, Li and Zhang (2023) found the high transferability of these interactions across samples and across models.

Above findings partially guarantee that Harsanyi interactions can be roughly taken as sparse primitives responsible for the network output. Thus, **these interactions provide us a more straightforward way to redefine the representation power of a DNN.** As in (Ren et al. 2023a), let us just call such sparse interactions as ***interactive concepts*** encoded by the DNN. Let an interactive concept be frequently extracted by the DNN from training samples, *e.g.*, a common face concept $S = \{\text{eyes}, \text{nose}, \text{mouth}\}$ shared by different images. **If this concept also frequently appears in testing samples, then this concept is considered generalizable; otherwise, not generalizable.** Because the network output is proved to be the sum of effects of different interactive concepts, the generalization of concepts can be a deep insight into the generalization of the entire DNN. Consequently, an out-of-the-distribution (OOD) sample will be explained to contain some non-generalizable interactive concepts[1].

Although there is a common intuition that more complex representations usually lead to over-fitting, this study uses an analytic inconsistency of concepts to explain the connection between the complexity of interactive concepts and their generalization power. The complexity of an interactive concept S is defined as the number of input variables in S, which is also termed as the *order* of the concept, *i.e.*, $\text{order}(S) = |S|$. Therefore, a high-order interactive concept contains a large number of input variables, and represents a complex concept. In this way, we use the high inconsistency to noises of high-order concepts to explain the high over-fitting risk of high-order concepts.

Besides, the high over-fitting risk of high-order concepts can also be explained by the detouring dynamic of learning high-order concepts. *I.e.*, we find that a high-order concept is more likely to be mistakenly represented by the DNN as a mixture of low-order concepts. We also find the following four phenomena to explain the high over-fitting risk of high-order concepts.

• For each concept, we compute the distribution of its effects on training samples and such a distribution on testing samples. We find that compared to the distribution of high-order concepts, the distribution of low-order concepts in training samples and that in testing samples are usually more similar to each other. This indicates the strong generalization power of low-order concepts.

• Under adversarial perturbations, high-order concepts are more likely to make inconsistent interaction effects than low-order concepts.

• Let us focus on a set of DNNs with the same architecture, which are trained at different over-fitting levels. We find that over-fitted DNNs usually encode stronger high-order interactive concepts than normal DNNs.

[1]Please see Section 1 in supplemental materials for details.

• Besides, normal DNNs usually learn low-order interactive concepts faster than over-fitted DNNs.

Interactive concepts vs. cognitive concepts and other interaction metrics. Although the Harsanyi interactive concept seems partially aligned with humans' cognition to some extent (Cheng et al. 2021b), we do not think such interactive concepts exactly fit humans' cognition. More crucially, the mathematical generalization power of a concept (defined in Equation (3)) does not depend on whether the concept fits human cognition. To this end, Ren et al. (2023a) have proved that the Harsanyi interaction could represent primitives of inference logic of a DNN, which was already sufficient for our research. Please see Section 2 in supplemental materials for detailed comparisons between the Harsanyi interaction and other interaction metrics.

In general, a DNN's representation complexity is different from the cognitive complexity. For example, let us consider a small ball concept consisting of a few pixels (low-order concept) and a large ball concept consisting of massive pixels (high-order concept) in images. These two balls have similar cognitive difficulty. However, from the perspective of a DNN, a large ball has more pixels, so that the DNN has to examine whether all pixels within the large ball share the same color without exceptions. This is more difficult than examining a few pixels within a small ball.

Explaining Generalization Using Concepts

Currently, there is no formal and universally-accepted definition for concepts encoded by DNNs. In this paper, we follow Ren et al. (2023a) to take the Harsanyi interaction as a simplified definition of concepts or primitives encoded by a DNN. These interactions are proved to well mimic network outputs under different input variations, so we can roughly consider such concepts as primitives to analyze the DNN. Our analysis does not require the exact fitness between the concept and human cognition.

Preliminaries: Sparse Interactive Concepts

Li and Zhang (2023), Ren et al. (2023a) discovered and Ren et al. (2023b) mathematically proved that **a well-trained DNN usually only encoded a small number of interactions between input variables**. Specifically, given a well-trained DNN v and an input sample $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^\top$ with n input variables, let $N = \{1, 2, \ldots, n\}$ denote the indices of all n input variables in $\boldsymbol{x}$, and let $v(\boldsymbol{x}) \in \mathbb{R}$ denote the scalar output of the DNN or a certain output dimension of the DNN[2]. Then, the Harsanyi dividend (or Harsanyi interaction) (Harsanyi 1963) is used to quantify the effect of the interaction between a set $S \subseteq N$ of input variables.

$$\forall S \subseteq N, I(S|\boldsymbol{x}) = \sum\nolimits_{T\subseteq S} (-1)^{|S|-|T|} \cdot v(\boldsymbol{x}_T), \qquad (1)$$

where $\boldsymbol{x}_T$ represents a sample whose input variables in $N \setminus T$ are masked by baseline values[3].

[2]Note that people can apply different settings for $v(\boldsymbol{x})$. In particular, for multi-category classification tasks, we set $v(\boldsymbol{x}) = \log \frac{p(y=y^{\text{truth}}|\boldsymbol{x})}{1-p(y=y^{\text{truth}}|\boldsymbol{x})} \in \mathbb{R}$ by following (Deng et al. 2021).

[3]The baseline value of each input variable is usually implemented as the mean value of this input variable over all sam-

Each interaction with considerable effect $I(S|\boldsymbol{x})$ represents the AND relationship between input variables in S. For example, in Figure 1, the face interaction consists of image patches in the set $S_{\text{face}} = \{\text{eyes}, \text{mouth}, \text{nose}\}$. Only when all three image patches in S_{face} are present in the input sample $\boldsymbol{x}$, the face interaction S_{face} is activated, and makes a numerical effect $I(S_{\text{face}}|\boldsymbol{x})$ on the network output. Otherwise, if any image patch in S_{face} is masked in the input image $\boldsymbol{x}$, then we can no longer measure a numerical effect of the interaction, *i.e.*, getting $I(S_{\text{face}}|\boldsymbol{x}_{\text{masked}}) = 0$ based on Equation (1).

Sparsity & universal matching. Ren et al. (2023b) have proved that although there are 2^n different combinations of variables $S \subseteq N$, as Figure 2 shows, most well-trained DNNs[4] only encode a small number of interactions (combinations) $S \in \Omega_{\text{salient}}$ with salient effects $I(S|\boldsymbol{x})$, subject to $|\Omega_{\text{salient}}| \ll 2^n$. All other interactions measured in Equation (1) have almost zero effects, $I(S|\boldsymbol{x}) \approx 0$, which represent noisy patterns.

Theorem 1. *An input sample $\boldsymbol{x}$ can be masked in 2^n ways by sampling different $T \subseteq N$. For any randomly masked sample $\boldsymbol{x}_T$, Ren et al. (2023a) have proved that*

$$v(\boldsymbol{x}_T) = \sum\nolimits_{S\subseteq T} I(S|\boldsymbol{x}) \approx \sum\nolimits_{S\subseteq T: S\in\Omega_{salient}} I(S|\boldsymbol{x}) \quad (2)$$

Based on the proved sparsity, Theorem 1 further indicates that network outputs on all 2^n randomly masked samples $\{\boldsymbol{x}_T : T \subseteq N\}$ can be **universally approximated** by a small number of salient interactions in Ω_{salient}, subject to $|\Omega_{\text{salient}}| \ll 2^n$. According to Occam's Razor (Blumer et al. 1987), if the inference score can be explained as just a small number of concepts, then the concept is more likely to reflect the essential knowledge encoded by a DNN, instead of a mathematical trick without clear meanings. In this way, interactive concepts can be defined as follows.

Definition of interactive concepts. Considering the proved sparsity, an interactive concept is defined as a salient interaction. Given a threshold τ, the set of interactive concepts are defined as $\Omega_{\text{salient}} = \{S \subseteq N : |I(S|\boldsymbol{x})| > \tau\}$.

Mathematical and experimental supports for interactive concepts. (1) First, although there is no theory to ensure such a simplified definition exactly fits concepts in human cognition, it is proved that this definition mathematically guarantees that the output of DNNs can be approximated by sparse interactive concepts. (2) Besides, Li and Zhang (2023) observed that interactive concepts also had certain **transferability** across samples and across models, *i.e.*, concepts in one sample could also appear in another sample in the same category, and concepts encoded by a DNN were usually also encoded by other DNNs. (3) Finally, a salient interactive concept also exhibited **strong discrimination power**, *i.e.*, if a set of samples all have the same salient concept, then this concept will probably push these samples towards the same category in classification.

Complexity (order) of interactive concepts. The complexity of the interactive concept S is defined as the number of input variables contained in the concept, which is also termed as the ***order*** of the concept, *i.e.*, $order(S) = |S|$. A low-order (simple) interactive concept represents interactions between a small number of input variables. A high-order (complex) interactive concept represents a complex interaction between a large number of input variables.

ples (Dabkowski and Gal 2017).

[4]Please see Section 4 in supplemental materials for detailed common conditions for the emergence of sparse interactions.

High-Order Concepts Are More Over-Fitted

Although there is a common heuristic that complex concepts are usually more likely to be over-fitted, people still do not know the exact definition of concepts with an analytic connection to their generalization power. Because we also find the low generalization power of complex (high-order) interactive concepts, in this study, we make the first attempt to clarify the high inconsistency of complex (high-order) concepts, *i.e.*, complex concepts are more sensitive to small noises in the data than simple concepts, which is responsible for the low generalization power of complex (high-order) concepts. Various experiments have verified our findings. This may shed new lights on how to evaluate the generalization power in terms of concepts.

Illustrating Concepts of Different Orders. Before investigating the relationship between the complexity of concepts and the generalization power of a DNN, let us first visualize concepts extracted from a DNN. Specifically, we trained a seven-layer MLP (MLP-7-census) on the census dataset (Asuncion and Newman 2007) and a seven-layer MLP (MLP-7-TV) on the TV news dataset (Asuncion and Newman 2007), respectively. Each layer of the MLPs contained 100 neurons. We also trained AlexNet (Krizhevsky, Sutskever, and Hinton 2017), ResNet-20 (He et al. 2016), and VGG-11 (Simonyan and Zisserman 2014) on the MNIST dataset (LeCun et al. 1998) (AlexNet-MNIST, ResNet-20-MNIST, VGG-11-MNIST) and the CIFAR-10 dataset (Krizhevsky, Hinton et al. 2009) (AlexNet-CIFAR-10, ResNet-20-CIFAR-10, VGG-11-CIFAR-10).

Although we only analyzed concepts in above DNNs in this very preliminary study, our findings could actually generalize to more diverse network architectures and datasets. It is because the proof for the sparse interactive concepts is agnostic to both network architectures and datasets.

Given a DNN and an input sample $\boldsymbol{x} \in \mathbb{R}^n$, we computed numerical effects $I(S|\boldsymbol{x})$ of all 2^n potential interactions, $S \subseteq N$. We found a common phenomenon that interactions encoded by a DNN were usually very ***sparse***, which ubiquitously existed in different DNNs. As Figure 2 shows, numerical effects of 80%-95% interactive concepts were almost zero, $|I(S|\boldsymbol{x})| \approx 0$, and only a small number of interactive concepts had relatively significant effects $|I(S|\boldsymbol{x})|$. This finding was also consistent with the conclusion found by Ren et al. (2023b). The sparsity of interactive concepts ensured the trustworthiness of such a concept definition.

Moreover, for each DNN, we compared the number of salient concepts of different orders. Given a threshold τ, we defined the set of salient concepts Ω_{salient} as interactive concepts whose strengths were greater than τ, *i.e.*, $\Omega_{\text{salient}} = \{S \subseteq N : |I(S|\boldsymbol{x})| > \tau\}$. Accordingly, $\Omega^{(s)}_{\text{salient}} = \{S \subseteq N : |I(S|\boldsymbol{x})| > \tau \text{ and } |S| = s\}$ represented all salient concepts of

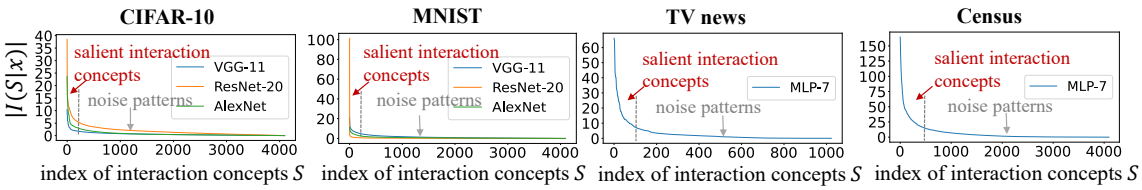

Figure 2: Interactive concepts encoded by a DNN are usually very sparse. This phenomenon exists in various DNNs trained on different datasets. We sort the interactive concepts to a decreasing order of the interaction strength $|I(S|\boldsymbol{x})|$.

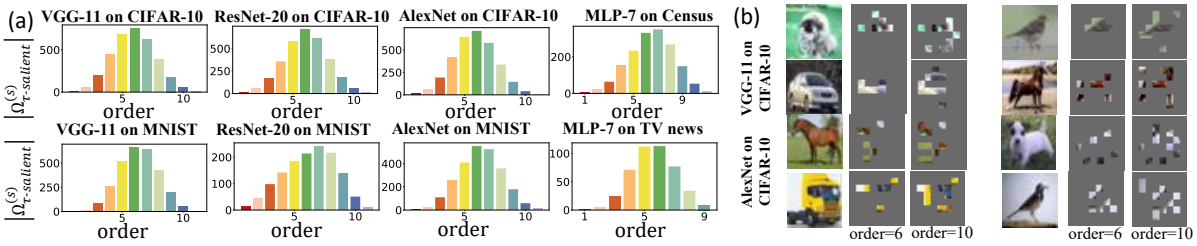

Figure 3: (a) Histogram[5] of salient concepts of different orders, $|\Omega^{(s)}_{\tau\text{-salient}}|$. (b) Visualization of salient concepts of different orders. Salient concepts are usually made up by image patches that contain discriminative parts of the object.

the s-th order. Figure 3 (a) shows the number of salient concepts of different orders, $|\Omega^{(s)}_{\text{salient}}|$, where the threshold was set to be $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$. This figure illustrats that there were more middle-order salient concepts than low-order salient concepts and high-order salient concepts.[5]

Finally, in Figure 3 (b), we visualized several salient concepts, covering middle-order and high-order concepts. We used the aforementioned experimental settings of AlexNet-CIFAR-10 and VGG-11-CIFAR-10, and we set $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$. We found that salient concepts of different orders were usually made up by image patches that contained discriminative parts of the object.

Generalization to Testing Samples. Before the analytic explanation for the generalization power of a DNN, let us first experimentally verify that **compared to high-order interactive concepts, low-order interactive concepts are more likely to have the distribution in training samples being similar to the distribution in testing samples.** To this end, previous studies used the gap of the loss (Neyshabur et al. 2017; Bousquet, Klochkov, and Zhivotovskiy 2020; Deng, He, and Su 2021; Haghifam et al. 2020, 2021) or the smoothness of the loss landscape (Keskar et al. 2016; Li et al. 2018; Foret et al. 2021; Kwon et al. 2021) to investigate the generalization power of a DNN.

In comparison, the decomposition of interactive concepts provides us *a more straightforward way* to define the generalization power of a DNN. *I.e.*, *if an interactive concept is frequently extracted by the DNN from training samples, then it is also supposed to frequently appear in testing samples. Otherwise, this interactive concept is not considered to be well generalized.*

In this way, we can define the generalization power of m-order interactive concepts *w.r.t.* the category c as the similarity between the distribution of m-order interactive concepts in training samples of category c and that in testing samples of category c. Let the vector $I^{(m)}_{\text{train},c} = [I^{(m)}_{\text{train},c}(S_1), I^{(m)}_{\text{train},c}(S_2), ..., I^{(m)}_{\text{train},c}(S_d)]^\top \in \mathbb{R}^d$ represent the distribution of m-order interactive concepts over training samples in the category c, which enumerates all $d = \binom{n}{m}$ possible m-order interactive concepts. The i-th dimension $I^{(m)}_{\text{train},c}(S_i) = \mathbb{E}_{\boldsymbol{x}\in D_{\text{train},c}}[I(S_i|\boldsymbol{x})]$ represents the average effect of the interactive concept S_i over different training samples in the category c. Accordingly, the vector $I^{(m)}_{\text{test},c}$ denotes the distribution of m-order concepts over testing samples in the category c. Then, the similarity of the concept distribution between training samples and testing samples is given as the Jaccard similarity between $\tilde{I}^{(m)}_{\text{train},c}$ and $\tilde{I}^{(m)}_{\text{test},c}$,

$$\text{sim}(\tilde{I}^{(m)}_{\text{train},c}, \tilde{I}^{(m)}_{\text{test},c}) = \frac{\|\min(\tilde{I}^{(m)}_{\text{train},c}, \tilde{I}^{(m)}_{\text{test},c})\|_1}{\|\max(\tilde{I}^{(m)}_{\text{train},c}, \tilde{I}^{(m)}_{\text{test},c})\|_1}, \tag{3}$$

where we extend the d-dimensional vector $I^{(m)}_{\text{train},c}$ into a $2d$-dimensional vector $\tilde{I}^{(m)}_{\text{train},c} = [(I^{(m),+}_{\text{train},c})^\top, (-I^{(m),-}_{\text{train},c})^\top]^\top = [(\max(I^{(m)}_{\text{train},c}, 0))^\top, (-\min(I^{(m)}_{\text{train},c}, 0))^\top]^\top \in \mathbb{R}^{2d}$ with non-negative elements. Similarly, $\tilde{I}^{(m)}_{\text{test},c}$ is constructed on $I^{(m)}_{\text{test},c}$ to contain non-negative elements. Thus, a high similarity $\text{sim}(\tilde{I}^{(m)}_{\text{train},c}, \tilde{I}^{(m)}_{\text{test},c})$ indicates that most m-order interactive concepts in the category c can be well generalized to testing samples in the category c.

Furthermore, we conducted experiments to check whether high-order interactive concepts were more likely to be over-fitted. Specifically, we trained a seven-layer MLP for the census dataset and the TV news dataset, respectively. We also trained AlexNet, VGG-11 and ResNet-20 on the

[5]This conclusion does not conflict with, but actually supports, the representation bottleneck found by Deng et al. (2021), because that work used a different type of interaction. Please see Section 2 and Section 3 in the supplementary material for more discussion.

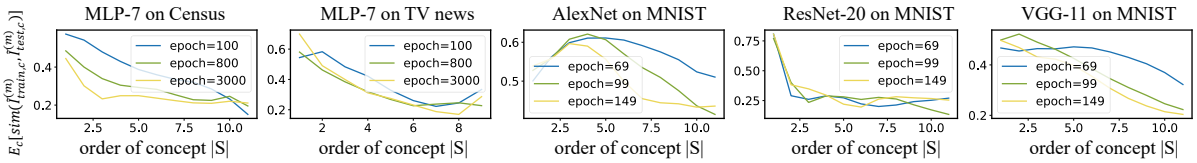

Figure 4: Average similarity between interactive concepts from training samples and those extracted from testing samples.

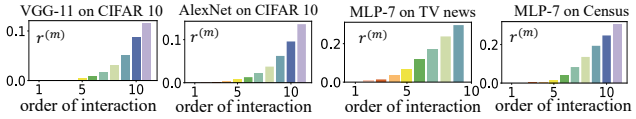

Figure 5: Comparison of the ratio $r^{(m)}$ of inconsistent concepts over different orders. High-order interactive concepts are usually more likely to make inconsistent effects on given noisy data, which verifies Theorem 2.

MNIST dataset. Given each DNN, we computed interactive concepts of each order m. For each category c, we measured the above conceptual similarity for m-order interactive concepts, and Figure 4 reported the average similarity over different categories, *i.e.*, *similarity* $= \mathbb{E}_c[\text{sim}(\tilde{I}^{(m)}_{\text{train},c}, \tilde{I}^{(m)}_{\text{test},c})]$. We found that low-order interactive concepts usually had more similar distributions between training data and testing data than high-order interactive concepts. This meant that compared to high-order concepts, the DNN was more likely to extract similar low-order concepts from the training data and testing data. In other words, low-order concepts in training data could be better generalized to testing data.

Inconsistency of High-Order Concepts. In this study, we try to use the inconsistency of high-order concepts to explain their low generalization power. Before the proof, this subsection first verifies that high-order concepts usually make inconsistent interactive effects on noisy data, *i.e.*, the same high-order concept may push a sample towards a category, but pull another sample away from this category. Intuitively, such inconsistent effects over different samples usually makes an interactive concept perform like a noisy pattern, rather than a generalizable concept.

Specifically, the inconsistency of a concept over different samples can be measured as follows. Given a normal sample $\boldsymbol{x}$, we select a set of salient concepts of each m-th order, $\Omega^{(m)}_{\boldsymbol{x}} = \{S \subseteq N : |S| = m \wedge |I(S|\boldsymbol{x})| > \tau\}$, the threshold is set to be $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$. Then, by adding the adversarial perturbation generated by Madry et al. (2018), we get an adversarial example $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{\delta}$. $I(S|\tilde{\boldsymbol{x}})$ denotes the interaction effect of the originally salient concepts $S \in \Omega_{\text{salient}}$ on the adversarial example $\tilde{x}$. If $I(S|\boldsymbol{x})$ and $I(S|\tilde{\boldsymbol{x}})$ have the same sign, *i.e.*, $I(S|\boldsymbol{x}) \cdot I(S|\tilde{\boldsymbol{x}}) > 0$, we consider that the interactive concept S is consistent in adversarial attacking; otherwise not. Thus, we compute the ratio of inconsistent concepts to all the m-order salient concepts as $r^{(m)} = \mathbb{E}_{\boldsymbol{x}\in D}\frac{|\{S\subseteq N: S\in\Omega^{(m)}_{\boldsymbol{x}} \wedge I(S|\boldsymbol{x})\cdot I(S|\tilde{\boldsymbol{x}})<0\}|}{|\Omega^{(m)}_{\boldsymbol{x}}|}$.

To compare the inconsistency over different orders, we follow experimental settings in the Section 2.2.1 to train AlexNet and VGG-11 on the CIFAR-10 dataset, and to train seven-layer MLPs on the census dataset and the TV news dataset. Figure 5 shows that the ratio $r^{(m)}$ of inconsistent salient concepts increases along with the order m.

Analytic Inconsistency of Concepts. All above experimental findings on the generalization power of concepts are related to the phenomenon of the inconsistency of high-order concepts, *i.e.*, high-order concepts are more sensitive to small noises in the input sample than low-order concepts. Therefore, we aim to prove that **the interaction effect's variance of the concept increases with the concept's order exponentially under a simple setting.** Let us add a Gaussian perturbation $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \delta^2\boldsymbol{I})$ to the input sample $\boldsymbol{x}$ and obtain $\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{\epsilon}$. The added perturbation represents noises/variations that inevitably exist in the data. We admit that there are other types of noises, such as texture variations and the shape deformation in object classification. In this study, we just use the Gaussian perturbation to represent the noises/variations in the data. Our conclusion may still provide conceptual insights into real-world applications.

Lemma 1. *Given a neural network v and an arbitrary perturbed input sample $\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{\epsilon}$, the neural network output $v(\boldsymbol{x}')$ can be rewritten by following the Taylor series expansion at the baseline point $\boldsymbol{b} = [b_1, \ldots, b_n]^\top$,*

$$v(\boldsymbol{x}') = v(\boldsymbol{b}) + \sum_{k=1}^{\infty}\sum_{\boldsymbol{\kappa}\in O_k} C(\boldsymbol{\kappa}) \cdot \nabla_v(\boldsymbol{\kappa}) \cdot \pi(\boldsymbol{\kappa}|\boldsymbol{x}'), \quad (4)$$

including the coefficient $C(\boldsymbol{\kappa}) = \frac{1}{(\kappa_1+\cdots+\kappa_n)!}\binom{\kappa_1+\cdots+\kappa_n}{\kappa_1,\ldots,\kappa_n} \in \mathbb{R}$, the partial derivative $\nabla_v(\boldsymbol{\kappa}) = \frac{\partial^{\kappa_1+\cdots+\kappa_n} v(\boldsymbol{b})}{\partial^{\kappa_1} x_1 \ldots \partial^{\kappa_n} x_n} \in \mathbb{R}$, and the expansion term $\pi(\boldsymbol{\kappa}|\boldsymbol{x}') = \prod_{i=1}^n (x'_i - b_i)^{\kappa_i}$. Here, $\boldsymbol{\kappa} = [\kappa_1, \ldots, \kappa_n] \in \mathbb{N}^n$ denotes the non-negative integer degree vector of each Taylor expansion term. Correspondingly, $O_k = \{\boldsymbol{\kappa} \in \mathbb{N}^n | \kappa_1 + \cdots + \kappa_n = k\}$ represents the set of all expansion terms of the k-th order.

As a prerequisite, Lemma 1 gives the Taylor series expansion when the network output $v(\boldsymbol{x}')$ is expanded at the baseline point $\boldsymbol{b} = [b_1, \ldots, b_n]^\top$. We use the baseline value b_i to represent the masking state of the input variable x_i. Normally, we can set the input variable x_i as the average value of x_i over different samples to remove the information (Ancona, Oztireli, and Gross 2019), *i.e.*, $b_i = \mu_i = \mathbb{E}_{\boldsymbol{x}}[x_i]$. However, pushing the input variable x_i a big distance $\alpha \in \mathbb{R}$ towards μ_i is usually enough to remove the information in real applications. Thus, we temporarily set $b_i = x_i + \alpha$, if $x_i < \mu_i$; and set $b_i = x_i - \alpha$, if $x_i > \mu_i$, to simplify the proof.

Theorem 2. *Given a neural network v and an arbitrary perturbed input sample $\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{\epsilon}$ by adding a Gaussian perturbation $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \delta^2\boldsymbol{I})$, the interactive effect $I(S|\boldsymbol{x}')$ is defined*

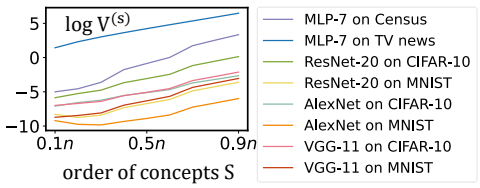

Figure 6: Logarithm of the variance. The interactive effect's variance increased along with the order exponentially. It verifies the inconsistency of high-order concepts in Theorem 2.

by setting $(\boldsymbol{x}'_T)_i = x'_i$ *if* $i \in T$ *and setting* $(\boldsymbol{x}'_T)_i = b_i$ *if* $i \notin T$. *Then, we obtain*

$$\begin{aligned} I(S|\boldsymbol{x}') &= \sum\nolimits_{\boldsymbol{\kappa}\in Q_S} C(\boldsymbol{\kappa}) \cdot \nabla_v(\boldsymbol{\kappa}) \cdot \pi(\boldsymbol{\kappa}|\boldsymbol{x}') \\ &= \sum\nolimits_{\boldsymbol{\kappa}\in Q_S} Z(\boldsymbol{\kappa}) \cdot \hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}'), \end{aligned} \tag{5}$$

where $\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}') = \prod_{i=1}^n (\frac{sign(x_i - b_i)}{\alpha})^{\kappa_i} \cdot \pi(\boldsymbol{\kappa}|\boldsymbol{x}')$ *is a standard AND interaction of the degree vector* $\boldsymbol{\kappa}$, *and it is normalized to satisfy* $\forall \boldsymbol{\kappa} \in Q_S, \mathbb{E}_{\boldsymbol{\epsilon}}[\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{\epsilon})] = 1 + O(\delta^2)$. *In addition,* $Z(\boldsymbol{\kappa}) = \prod_{i=1}^n (\frac{\alpha}{sign(x_i - b_i)})^{\kappa_i} \cdot C(\boldsymbol{\kappa}) \cdot \nabla_v(\boldsymbol{\kappa})$ *denotes the scalar coefficient for* $\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}')$. $Q_S = \{\boldsymbol{\kappa} \in \mathbb{N}^n | \forall i \in S, \kappa_i \in \mathbb{N}^+; \forall i \notin S, \kappa_i = 0\}$ *denotes the set of degree vectors corresponding to all Taylor expansion terms involving only variables in* S. *Furthermore, the second-order moment of the standard AND interaction* $\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x} + \boldsymbol{\epsilon})$ *w.r.t. the Gaussian perturbations* $\boldsymbol{\epsilon}$ *is derived as follows.*

$$\forall \boldsymbol{\kappa} \in Q_S, \mathbb{E}_{\boldsymbol{\epsilon}}[\hat{\pi}^2(\boldsymbol{\kappa}|\boldsymbol{x} + \boldsymbol{\epsilon})] = \prod_{i\in S}[1 + \sum_{m=1}^{\kappa_i} c_m U_{2m}(\epsilon_i)], \tag{6}$$

where $U_{2m}(\epsilon_i) = \mathbb{E}[\epsilon_i^{2m}] > 0$, *and* $c_m = \binom{2\kappa_i}{2m} \frac{1}{\alpha^{2m}} > 0$.

Theorem 2 reformulates the interactive effect $I(S|\boldsymbol{x}')$ **into elementary standard interactions** $\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}')$**.** Specifically, Equation (6) tells us that for a specific standard interaction $\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}')$ subject to $\boldsymbol{\kappa} \in Q_S$, its second-order moment increases along with the order $|S|$ in a roughly exponential manner, but its mean value $\mathbb{E}_{\boldsymbol{\epsilon}}[\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}+\boldsymbol{\epsilon})] \approx 1$ is independent with the order. Therefore, we can roughly consider that its variance $Var_{\boldsymbol{\epsilon}}[\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}+\boldsymbol{\epsilon})] = \mathbb{E}_{\boldsymbol{\epsilon}}[\hat{\pi}^2(\boldsymbol{\kappa}|\boldsymbol{x}+\boldsymbol{\epsilon})] - \mathbb{E}^2_{\boldsymbol{\epsilon}}[\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}+\boldsymbol{\epsilon})]$ also increases along with the order $|S|$ exponentially.

Moreover, according to Equation (5), the interactive effect $I(S|\boldsymbol{x}')$ of the concept S is the weighted sum of all elementary terms $\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}+\boldsymbol{\epsilon})$ satisfying $\boldsymbol{\kappa} \in Q_S$, and different terms of $\hat{\pi}(\boldsymbol{\kappa}|\boldsymbol{x}+\boldsymbol{\epsilon})$ *w.r.t.* the same set S are roughly positively correlated to each other. Thus, in the simple setting of adding Gaussian perturbations to the input, we can consider that the variance of $I(S|\boldsymbol{x}')$ has approximately an exponent relation with the order $|S|$ of the concept S. **Therefore, Theorem 2 shows that high-order concepts usually make more inconsistent effects than low-order concepts.** Although there are other types of noises in the real data, our theory may still provide conceptual insights into real-world applications.

Experimental verification of Theorem 2. We conducted experiments to verify the exponential relation of the interactive effect's variance with the order of the concept, which is predicted by Theorem 2. To this end, given a well-trained DNN and an input sample $\boldsymbol{x}$, we added a Gaussian perturbation $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \delta^2\boldsymbol{I})$ to the input sample. Then, we used $V^{(s)} = \mathbb{E}_{\boldsymbol{x}}[\mathbb{E}_{|S|=s}[Var_{\boldsymbol{\epsilon}}[I(S|\boldsymbol{x}+\boldsymbol{\epsilon})]]]$ to measure the average variance of s-th order concepts *w.r.t.* the Gaussian perturbation $\boldsymbol{\epsilon}$. In experiments, we used DNNs introduced in Section 2.2.1 for testing. Figure 6 shows that the interactive effect's variance $V^{(s)}$ increased along with the order s in a roughly exponential manner. The inconsistency of high-order concepts in Theorem 2 is verified.

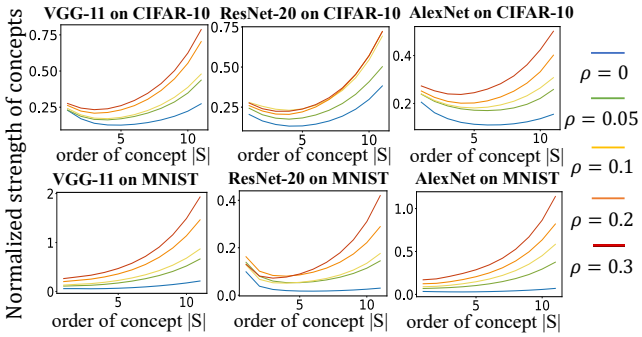

Figure 7: Interaction strength of DNNs trained with different noise levels ρ. We normalized the strength of interactive concepts $\mathbb{E}_{\boldsymbol{x}\in D}\mathbb{E}_{S:|S|=m}[\frac{|I^{(m)}(S|\boldsymbol{x})|}{\mathbb{E}_{\boldsymbol{x}\in D}[|v(N|\boldsymbol{x})-v(\varnothing|\boldsymbol{x})|]}]$.

An Over-Fitted DNN Usually Encodes Strong High-Order Interactive Concepts

In this subsection, we further analyze and explain the generalization power of the entire DNN based on the generalization power of the encoded interactive concepts.

To this end, we need to construct DNNs with different generalization power for investigation. In fact, the generalization power of a DNN is usually affected by various factors, such as the network architecture and training data. In this study, we consider a typical case, *i.e.*, random labels in training data usually push the DNN to be over-fitted to non-generalizable features for classification (Bae et al. 2022). Therefore, in experiments, we trained DNNs by applying different ratios of noise data. Specifically, we trained a DNN on training samples with a ρ $(0 \le \rho \le 1)$ ratio of incorrect labels, which was termed a *DNN with* ρ *noise*. We considered that a DNN trained with more incorrect labels (a high ρ value) was more over-fitted. We trained *AlexNet, ResNet-20 and VGG-11 with* $\rho = 0, 0.05, 0.1, 0.2, 0.3$ *noise* on the MNIST dataset and the CIFAR-10 dataset.

Claim 1: More label noise usually makes DNNs to encode stronger high-order interactive concepts. To verify this claim, for each DNN trained with a ρ ratio of incorrect labels, we computed the average interaction strength of m-order interactive concepts over different training samples, *i.e.*, $\mathbb{E}_{\boldsymbol{x}\in D}\mathbb{E}_{S:|S|=m}[\frac{|I^{(m)}(S|\boldsymbol{x})|}{\mathbb{E}_{\boldsymbol{x}\in D}[|v(N|\boldsymbol{x})-v(\varnothing|\boldsymbol{x})|]}]$. Figure 7 shows that the DNN trained with more incorrect labels (a high ρ value) usually encoded more significant high-order concepts than the DNN trained with fewer incorrect labels. In other words, DNNs trained with more label noise (with poorer generalization power) usually encoded stronger high-order concepts.

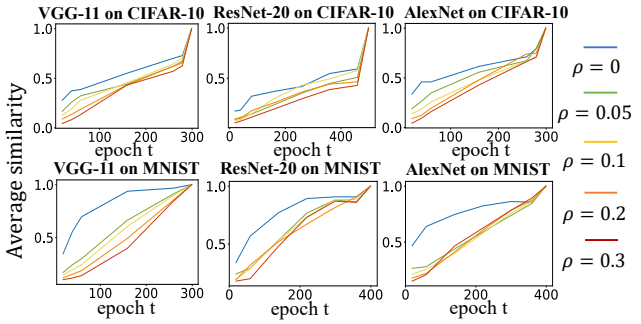

Figure 8: Average similarity $\text{Sim}^{(m=1,t)}$ between low-order interactive concepts encoded after the t-th epoch and those encoded after all epochs. We compare $\text{Sim}^{(m=1,t)}$ between DNNs trained with different noise levels ρ. More results can be found in supplemental materials.

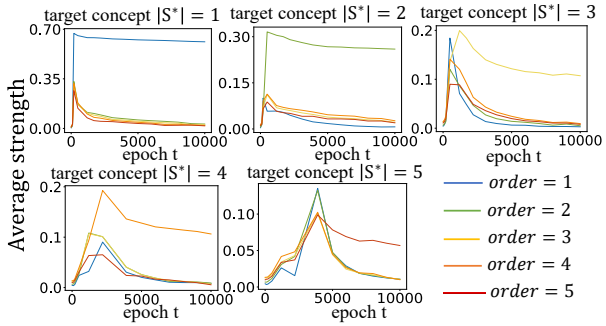

Figure 9: Average strength $\mathbb{E}_{\boldsymbol{x}\in X}[\sum_{S:|S|=m}|I_t^{(m)}(S|\boldsymbol{x})|]$ of interactive concepts of different orders.

Claim 2: Less label noise usually makes DNNs to learn low-order interactive concepts more quickly. To verify this claim, for each DNN trained with a ρ ratio of incorrect labels, we examined the learning progress of interactive concepts of a specific order m. We used the metric $\text{Sim}^{(m,t)} = \mathbb{E}_{\boldsymbol{x}\in D}[\text{sim}(\tilde{I}_t^{(m)}(\boldsymbol{x}), \tilde{I}^{*(m)}(\boldsymbol{x}))]$ to measure the learning progress of m-order interactive concepts at the t-th epoch, which was defined as the average Jaccard similarity between all m-order interactive concepts $\tilde{I}^{*(m)}(\boldsymbol{x})$ encoded by the finally trained DNN and all m-order interactive concepts $\tilde{I}_t^{(m)}(\boldsymbol{x})$ encoded by the DNNs $v^{(t)}$ trained after t epochs. Here, $\text{sim}(\cdot)$ and the two vectors $\tilde{I}^{*(m)}(\boldsymbol{x})$, $\tilde{I}_t^{(m)}(\boldsymbol{x})$ were defined in Equation (3). In this way, if a DNN obtained a high similarity $\text{Sim}^{(m,t)}$ (*i.e.*, achieved a high learning progress) in early epochs, then we considered that this DNN learned m-order interactive concepts quickly.

Then, we conducted experiments to compare the learning speeds of interactive concepts between aforementioned DNNs with different ratios of label noise. Figure 8 shows that a DNN trained with less label noise usually exhibited a higher $\text{Sim}^{(m,t)}$ for low-order interactive concepts. It meant that DNNs trained with less label noise usually learned low-order interactive concepts more quickly.

Detouring Dynamics of High-Order Concepts

In this section, we analyze the learning dynamics of concepts with a simple experimental setting, *i.e.*, using a DNN to fit a boolean polynomial. We find that a high-order concept is not directly learned, but is likely to be mistakenly encoded as a mixture of low-order concepts in early epochs. In spite of the simplicity of experiments, this finding may still provide conceptual insights into the reason why high-order concepts are more likely to be over-fitted.

We trained a DNN v to fit a random concept $S^* \subseteq N$ of the m-th order, $m = |S^*|$. Given an arbitrary input sample $\boldsymbol{x}$, if the input variable x_i was set to the original value, then we set $A_i = 1$; if the input variable was masked, then we set $A_i = 0$. Thus, the m-order target concept was formulated as $u_{S^*}(\boldsymbol{x}) = \prod_{i\in S^*} A_i$. We trained the DNN to fit the function of the concept $u_{S^*}(\boldsymbol{x})$ based on the $Loss = E_{\boldsymbol{x}\in X}[\|v(\boldsymbol{x}) - u_{S^*}(\boldsymbol{x})\|^2]$, *i.e.*, to fit a boolean polynomial. We trained a five-layer MLP on the dataset $X = \{0,1\}^n$, which contained samples corresponding to all masking states $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^\top \in X, \forall i, x_i \in \{0,1\}$. Here, $n = 10$.

If a DNN had been well trained after t epochs, then it was supposed to only extract a single concept with non-zero effect $I_t^{(m)}(S|\boldsymbol{x})$. Therefore, we examined whether the DNN encoded concepts of different orders or a single concept.

To this end, we used $\mathbb{E}_{\boldsymbol{x}\in X}[\sum_{S:|S|=m}|I_t^{(m)}(S|\boldsymbol{x})|]$ to denote the average strength of m-order interactions after the t-th epoch. We tracked the change of the average interaction strength over different epochs. Figure 9 shows that when a DNN was trained to fit a low-order concept, it usually learned such a concept directly. In comparison, when a DNN was trained to fit a high-order concept, the learning dynamics is detouring. Specifically, the DNN usually first learned low-order concepts. Then, the DNN shifted its attention to concepts of higher orders, and later gradually removed mistakenly learned low-order concepts.

In this way, the above detouring dynamics of high-order concepts showed that high-order concepts were more difficult to be learned. A high-order concept was likely to be mistakenly encoded as a mixture of low-order concepts. Therefore, high-order concepts were less likely to be generalized to testing data than low-order concepts.

Conclusion

In this paper, we provide a conceptual understanding of the reason why low-order concepts in training data can usually better generalize to testing data than high-order concepts. Specifically, we prove that the average inconsistency of concepts usually increases exponentially along with the order of concepts. We find that DNNs with poorer generalization power usually encode more high-order concepts, and DNNs with stronger generalization power usually encode low-order concepts more quickly. Moreover, we find that low-order concepts are usually learned directly, but high-order concepts are more likely to be mistakenly encoded as a mixture of various incorrect low-order concepts. These all explain the low generalization power of high-order interactive concepts. Section 8 in supplemental materials will introduce future practical values of this study.

Acknowledgments

This work is partially supported by the National Key R&D Program of China (2021ZD0111602), the National Nature Science Foundation of China (62276165), Shanghai Natural Science Foundation (21JC1403800, 21ZR1434600) and the National Nature Science Foundation of China (62206170). This work is also partially supported by Huawei Technologies Inc.

References

Ancona, M.; Oztireli, C.; and Gross, M. 2019. Explaining deep neural networks with a polynomial time algorithm for shapley value approximation. In *International Conference on Machine Learning*, 272–281. PMLR.

Asuncion, A.; and Newman, D. 2007. UCI machine learning repository.

Bae, H.; Shin, S.; Na, B.; Jang, J.; Song, K.; and Moon, I.-C. 2022. From noisy prediction to true label: Noisy prediction calibration via generative model. In *International Conference on Machine Learning*, 1277–1297. PMLR.

Blumer, A.; Ehrenfeucht, A.; Haussler, D.; and Warmuth, M. K. 1987. Occam's razor. *Information processing letters*, 24(6): 377–380.

Bousquet, O.; Klochkov, Y.; and Zhivotovskiy, N. 2020. Sharper bounds for uniformly stable algorithms. In *Conference on Learning Theory*, 610–626. PMLR.

Cheng, X.; Chu, C.; Zheng, Y.; Ren, J.; and Zhang, Q. 2021a. A game-theoretic taxonomy of visual concepts in dnns. *arXiv preprint arXiv:2106.10938*.

Cheng, X.; Wang, X.; Xue, H.; Liang, Z.; and Zhang, Q. 2021b. A hypothesis for the aesthetic appreciation in neural networks. *arXiv preprint arXiv:2108.02646*.

Dabkowski, P.; and Gal, Y. 2017. Real time image saliency for black box classifiers. *Advances in neural information processing systems*, 30.

Deng, H.; Ren, Q.; Zhang, H.; and Zhang, Q. 2021. DISCOVERING AND EXPLAINING THE REPRESENTATION BOTTLENECK OF DNNS. In *International Conference on Learning Representations*.

Deng, Z.; He, H.; and Su, W. 2021. Toward better generalization bounds with locally elastic stability. In *International Conference on Machine Learning*, 2590–2600. PMLR.

Dziugaite, G. K.; and Roy, D. M. 2017. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv preprint arXiv:1703.11008*.

Foret, P.; Kleiner, A.; Mobahi, H.; and Neyshabur, B. 2021. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*.

Fort, S.; Nowak, P. K.; Jastrzebski, S.; and Narayanan, S. 2019. Stiffness: A new perspective on generalization in neural networks. *arXiv preprint arXiv:1901.09491*.

Haghifam, M.; Dziugaite, G. K.; Moran, S.; and Roy, D. 2021. Towards a unified information-theoretic framework for generalization. *Advances in Neural Information Processing Systems*, 34: 26370–26381.

Haghifam, M.; Negrea, J.; Khisti, A.; Roy, D. M.; and Dziugaite, G. K. 2020. Sharpened generalization bounds based on conditional mutual information and an application to noisy, iterative algorithms. *Advances in Neural Information Processing Systems*, 33: 9925–9935.

Harsanyi, J. C. 1963. A simplified bargaining model for the n-person cooperative game. *International Economic Review*, 4(2): 194–220.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Keskar, N. S.; Mudigere, D.; Nocedal, J.; Smelyanskiy, M.; and Tang, P. T. P. 2016. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2017. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6): 84–90.

Kwon, J.; Kim, J.; Park, H.; and Choi, I. K. 2021. Asam: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In *International Conference on Machine Learning*, 5905–5914. PMLR.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Li, H.; Xu, Z.; Taylor, G.; Studer, C.; and Goldstein, T. 2018. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31.

Li, M.; and Zhang, Q. 2023. Does a Neural Network Really Encode Symbolic Concept? In *International Conference on Machine Learning*.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*.

Neyshabur, B.; Bhojanapalli, S.; McAllester, D.; and Srebro, N. 2017. Exploring generalization in deep learning. *Advances in neural information processing systems*, 30.

Neyshabur, B.; Tomioka, R.; and Srebro, N. 2015. Norm-based capacity control in neural networks. In *Conference on Learning Theory*, 1376–1401. PMLR.

Ren, J.; Li, M.; Chen, Q.; Deng, H.; and Zhang, Q. 2023a. Defining and Quantifying the Emergence of Sparse Concepts in DNNs. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Ren, Q.; Gao, J.; Shen, W.; and Zhang, Q. 2023b. Where We Have Arrived in Proving the Emergence of Sparse Symbolic Concepts in AI Models. *arXiv preprint arXiv:2305.01939*.

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

Weng, T.-W.; Zhang, H.; Chen, P.-Y.; Yi, J.; Su, D.; Gao, Y.; Hsieh, C.-J.; and Daniel, L. 2018. Evaluating the robustness of neural networks: An extreme value theory approach. *arXiv preprint arXiv:1801.10578*.