Does a Neural Network Really Encode Symbolic Concepts?

Mingjie Li [1] Quanshi Zhang [1]

Abstract

Recently, a series of studies have tried to extract interactions between input variables modeled by a DNN and define such interactions as concepts encoded by the DNN. However, strictly speaking, there still lacks a solid guarantee whether such interactions indeed represent meaningful concepts. Therefore, in this paper, we examine the trustworthiness of interaction concepts from four perspectives. Extensive empirical studies have verified that a well-trained DNN usually encodes sparse, transferable, and discriminative concepts, which is partially aligned with human intuition. The code is released at https://github.com/sjtu-xai-lab/interaction-concept.

1. Introduction

Understanding the black-box representation of deep neural networks (DNNs) has received increasing attention in recent years. Unlike graphical models with interpretable internal logic, the layerwise feature processing in DNNs makes it naturally difficult to explain DNNs from the perspective of symbolic concepts. Instead, previous studies interpreted DNNs from other perspectives, such as illustrating the visual appearance that maximizes the inference score (Simonyan et al., 2013; Yosinski et al., 2015), and estimating attribution/importance/saliency of input variables (Ribeiro et al., 2016; Sundararajan et al., 2017; Lundberg & Lee, 2017a). Zhou et al. (2015); Bau et al. (2017); Kim et al. (2018) visualized the potential correspondence between convolutional filters in a DNN and visual concepts in an empirical manner.

Unlike previous studies, a series of studies (Ren et al., 2021a; 2023a; Deng et al., 2022a) tried to define and propose an exact formulation for the concepts encoded by a DNN. Specifically, these studies discovered that a well-trained DNN usually encoded various interactions between different input variables, and the inference score on a specific input sample could be explained by numerical effects of different interactions. Thus, they claimed that each interaction pattern was a symbolic concept encoded by the DNN.

Specifically, let us consider a DNN given a sample with n input variables $N = \{1, 2, ..., n\}$ *e.g.*, given a sentence with five words "*he is a green hand.*" The DNN usually does not use each individual input variable for inference independently. Instead, the DNN lets different input variables interact with each other to construct concepts for inference. For example, a DNN may memorize the interaction between words in $S = \{green, hand\}$ with a specific numerical contribution $I(S)$ to push the DNN's inference towards the meaning of a "*beginner*." Such a combination of words is termed an *interaction concept*. Each interaction concept $S \subseteq N$ represents the AND relationship between input variables in S. Only the co-appearance of input variables in S can make an interaction effect $I(S)$ on the network output. In contrast, masking any word in $\{green, hand\}$ removes the interaction effect $I(S)$. In this way, Ren et al. (2021a) proved that the inference score y of a trained DNN on each sample can be written as the sum of effects of all potential symbolic concepts $S \subseteq N$, *i.e.* $y = \sum_{S \subseteq N} I(S)$.

However, the claim that "a DNN encodes symbolic concepts" is too counter-intuitive. Ren et al. (2021a) just formulated $I(S)$ that satisfied $y = \sum_{S \subseteq N} I(S)$. Current studies have not provided sufficient support for the claim that a DNN really learns symbolic concepts. *In fact, we should not ignore another potential situation that the defined effect $I(S)$ is just a mathematical transformation that ensures the decomposition of network output into concepts $y = \sum_{S \subseteq N} I(S)$, rather than faithfully representing a meaningful and transferable concept learned by a DNN.*

Therefore, in this paper, we aim to give a quantitative verification of the concept-emerging phenomenon, *i.e.*, whether a well-trained DNN summarizes transferable symbolic knowledge from chaotic raw data, like what human brain does. Or the defined effect $I(S)$ is just a mathematical game without clear meanings. To this end, we believe that if a well-trained

[*] Equal contribution [1] Shanghai Jiao Tong University. Correspondence to: Quanshi Zhang. 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. <zqs1022@sjtu.edu.cn>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

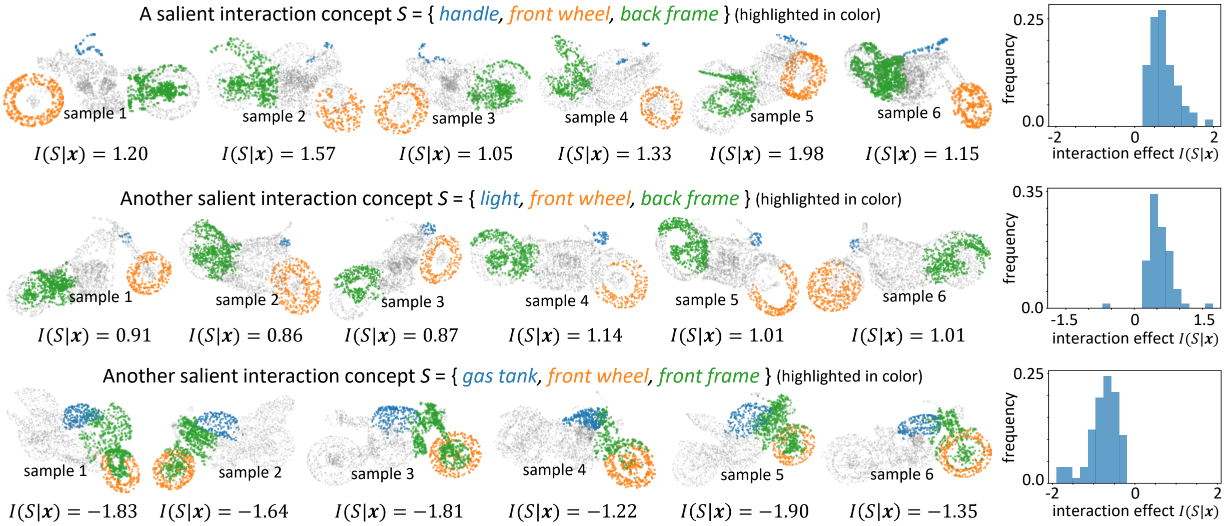


Figure 1. Visualization of interaction concepts S extracted by PointNet on different samples in the ShapeNet dataset. The histograms show the distribution of interaction effects $I(S|\boldsymbol{x})$ over samples in the "motorbike" category, where S is extracted as a salient concept.

DNN really encodes certain concepts, then the concepts are supposed to satisfy the following four requirements.

• **Representing network inference using a few concepts.** If a DNN really learns symbolic concepts, then the DNN's inference on a specific sample is supposed to be concisely explained by a small number of salient concepts, rather than a large number of concepts, according to both Occam's Razor and people's intuitive understanding towards concepts. In fact, the sparsity of concepts in a specific sample has been discussed by Ren et al. (2021a). In this paper, we further conduct extensive experiments on more diverse DNNs, in order to verify that a well-trained DNN usually extracts sparse concepts from each sample for inference.

• **A transferable concept dictionary through different samples.** If a DNN can use a relatively small set $\mathbf{D}$ of salient concepts, namely a concept dictionary, to approximate inference scores on different samples in a category, *i.e.*, $\forall \boldsymbol{x}, y \approx \sum_{S \in \mathbf{D}} I(S|\boldsymbol{x})$, then we consider the concept dictionary $\mathbf{D}$ represents common features shared by different samples in the category. Otherwise, if Ren et al. (2021a) extract a fully different set of concepts from each different sample in the same category, then these concepts probably represent noisy signals. In other words, convincing concepts must be stably extracted with high transferability through different samples in the same category.

• **Transferability of concepts across different DNNs.** Similarly, when we train different DNNs for the same task, different DNNs probably learn similar sets of concepts if they really memorize the defined "concepts" as basic inference patterns for the task.

• **Discrimination power of concepts.** Furthermore, if a DNN learns meaningful concepts, then these concepts are supposed to exhibit certain discrimination power in the classification task. The same concept extracted from different samples needs to consistently push the DNN towards the classification of a certain category.

To this end, we conducted experiments on various DNNs trained on different datasets for classification tasks, including tabular datasets, image datasets, and point-cloud datasets. We found that all these trained DNNs encoded transferable concepts. On the other hand, we also investigated a few extreme cases, in which the DNN either collapsed into simple linear models or failed to learn transferable and discriminative concepts. In sum, although we cannot theoretically prove the phenomenon of the emergence of transferable concepts, this phenomenon indeed happened for most tasks in our experiments.

Contributions of this paper can be summarized as follows. (1) Besides the sparsity of concepts, we propose three more perspectives to examine the concept-emerging phenomenon of a DNN, *i.e.*, whether the DNN summarizes transferable symbolic knowledge from chaotic raw data. (2) Extensive empirical studies on various tasks have verified that a well-trained DNN usually encodes transferable interaction concepts. (3) We also discussed three extreme cases, in which a DNN is unlikely to learn transferable concepts.

2. Related works

2.1. Understanding black-box representation of DNNs

Many explanation methods have been proposed to explain DNNs from different perspectives. Typical explanation methods include visualizing patterns encoded by a DNN (Simonyan et al., 2013; Zeiler & Fergus, 2014; Yosinski et al., 2015; Dosovitskiy & Brox, 2016), estimating the attribution/importance/saliency of each input variable (Ribeiro et al., 2016; Sundararajan et al., 2017; Lundberg & Lee, 2017b; Fong & Vedaldi, 2017; Zhou et al., 2016; Selvaraju et al., 2017), and learning feature vectors potentially correspond to semantic concepts (Kim et al., 2018). Unlike attribution methods, some studies focused on quantifying interactions between input variables (Sorokina et al., 2008; Murdoch et al., 2018; Singh et al., 2018; Jin et al., 2019; Janizek et al., 2020). In game theory, Grabisch & Roubens (1999); Sundararajan et al. (2020); Tsai et al. (2022) proposed interaction metrics from different perspectives. Some studies explained a DNN by distilling the DNN into another interpretable model (Frosst & Hinton, 2017; Che et al., 2016; Wu et al., 2018; Zhang et al., 2018; Vaughan et al., 2018; Tan et al., 2018). However, most explanation methods did not try to disentangle concepts encoded by a DNN.

2.2. Explainable AI (XAI) theories based on game-theoretic interactions

Our research group developed a theoretical framework based on game-theoretic interactions, which aims to tackle the following two challenges in XAI, *i.e.*, (1) extracting and quantifying concepts from implicit knowledge representations of DNNs and (2) utilizing these explicit concepts to explain the representational capacity of DNNs. Furthermore, we discovered that game-theoretic interactions provide a new perspective for analyzing the common underlying mechanism shared by previous XAI applications.

• *Using game-theoretical interactions to define concepts encoded by DNNs.* Quantifying the interactions between input variables is one of the ultimate problems facing XAI (Sundararajan et al., 2020; Tsai et al., 2022). Based on game theory, we introduced multi-variate interactions (Zhang et al., 2021a;c) and multi-order interactions (Zhang et al., 2021b) to analyze interactions encoded by the DNN. Recently, Ren et al. (2021a) proposed the mathematical formulation for concepts encoded by a DNN, and Ren et al. (2023a) further used such concepts to define the optimal baseline values for Shapely values. Based on this, recent studies have also observed (Ren et al., 2023a) and mathematically proved (Ren et al., 2023c) the concept-emerging phenomenon in DNNs. **However, strictly speaking, there still lacks theoretical guarantee for the interaction to prove whether the interaction represents the true concept encoded by a DNN or just a tricky metric without a clear meaning. Therefore, in this study, we examined the trustworthiness of the interaction concepts from four perspectives.**

• *Explaining the representation power of DNNs based on game-theoretic interactions.* Game-theoretical interactions facilitate the explanation of the representation capacity of a DNN from different perspectives, including the adversarial robustness (Wang et al., 2021a; Ren et al., 2021b), adversarial transferability (Wang et al., 2021b), and generalization power (Zhang et al., 2021b; Zhou et al., 2023). Besides, the game-theoretical interactions can also be utilized to explain the signal processing behavior of DNNs. For example, Cheng et al. (2021a) analyzed the distinctive behavior of a DNN encoding shape/texture features based on such interactions. Cheng et al. (2021b) discovered that salient interactions often represented different prototype features encoded by a DNN. Deng et al. (2022a) proved that it was difficult for a DNN to encode mid-order interactions, which reflected a representation bottleneck of DNNs. In comparison, Ren et al. (2023b) proved that a Bayesian neural network was less likely to encode high-order interactions, thereby alleviating the over-fitting problem.

• *Unifying empirical findings in the framework of game-theoretic interactions.* To unify different attribution methods, Deng et al. (2022b) used interactions as a unified reformulation of different attribution methods. They proved that attributions estimated by each of 14 attribution methods could all be represented as a certain allocation of interaction effects to different input variables. In addition, Zhang et al. (2022a) proved that the reduction of interactions was the common mathematical mechanism shared by a total of 12 previous approaches to enhance adversarial transferability.

3. Emergence of transferable concepts

3.1. Preliminaries: representing network inferences using interaction concepts

It is widely believed that the learning of a DNN can be considered as a regression problem, instead of explicitly learning symbolic concepts like how graphical models do. However, a series of studies (Ren et al., 2021a; 2023a; Deng et al., 2022a) have discovered that given a sufficiently-trained DNN for a classification task, its inference logic on a certain sample can usually be rewritten as the detection of specific concepts. In other words, the DNN's inference score on a specific sample can be sparsely disentangled into effects of a few concepts.

Disentangling the DNN's output as effects of interaction concepts. Specifically, let us consider a trained DNN $v : \mathbb{R}^n \to \mathbb{R}$ and an input sample $\boldsymbol{x}$ with n input variables indexed by $N = \{1, 2, ..., n\}$. Here, we assume the network output $v(\boldsymbol{x}) \in \mathbb{R}$ is a scalar. Note that different settings can be applied to $v(\boldsymbol{x})$. For example, for multi-category classification tasks, we may 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., 2022a). Then, given a function $v(\boldsymbol{x})$, Ren et al. (2021a) have proposed the following metric to quantify the interaction concept that is comprised of input variables in $S \subseteq N$.

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

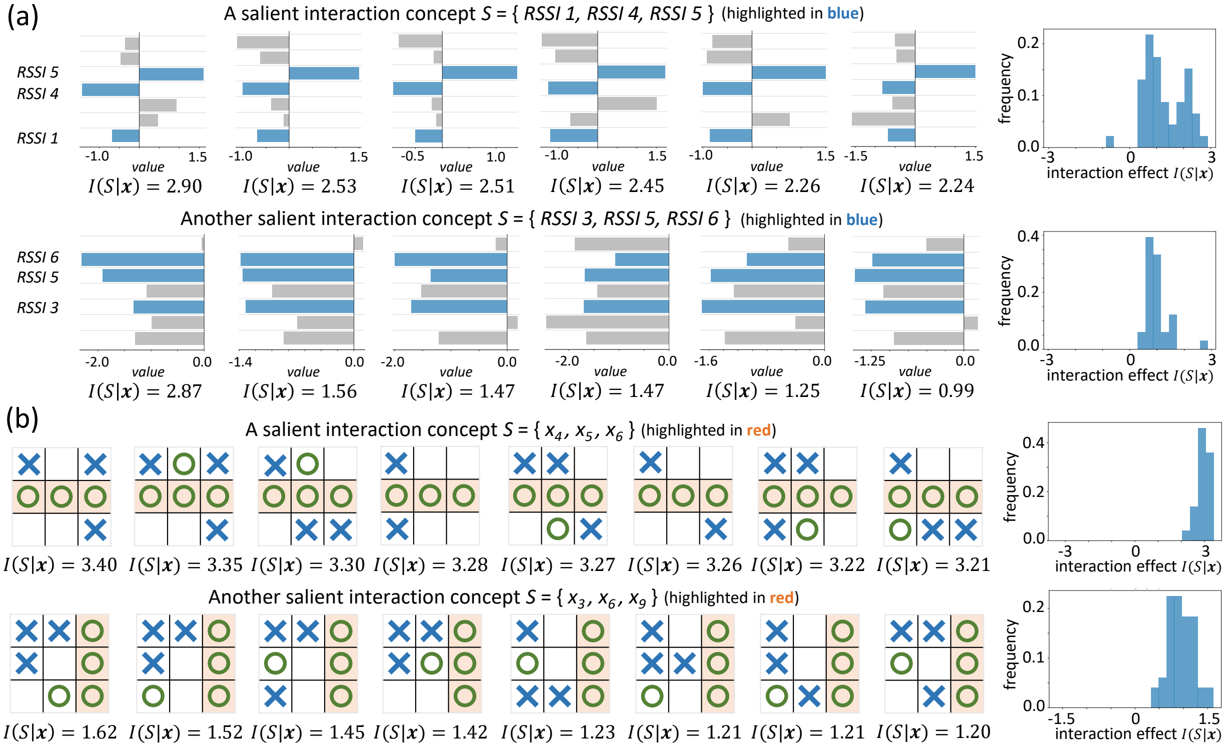


Figure 2. Visualization of interaction concepts S extracted by two *MLP-5* networks[3], which are trained on (a) the *wifi* dataset[3] and (b) the *tic-tac-toe* dataset[3]. The histograms show (a) the distribution of interaction effects $I(S|\boldsymbol{x})$ over samples in the 4th category, and (b) the distribution of interaction effects $I(S|\boldsymbol{x})$ over samples in sub-categories[5] with patterns $x_4 = x_5 = x_6 = 1$ and $x_3 = x_6 = x_9 = 1$.

where $\boldsymbol{x}_T$ denotes the input sample when we keep variables in $T \subseteq N$ unchanged and mask variables in $N \setminus T$ using baseline values[1].

Here, the interaction concept $I(S|x)$ extracted from the input x encodes an AND relationship (interaction) between input variables in S. For example, let us consider three image regions of $S = \{$*eyes, nose, mouth*$\}$ that form the *"face"* concept in a face classification task. Then, $I(S|\boldsymbol{x})$ measures the numerical effect of the concept on the classification score $v(\boldsymbol{x})$. Only when all image regions of *"eyes"*, *"nose"*, and *"mouth"* co-appear in the input image, the *"face"* concept is activated, and contributes a numerical effect $I(S|\boldsymbol{x})$ to the classification score. Otherwise, if any region is masked, then the *"face"* concept cannot be formed, which removes the interaction effect, making $I(S|\boldsymbol{x}_{\text{masked}}) = 0$.

Mathematically, the above definition for an interaction concept can be understood as the Harsanyi dividend (Harsanyi, 1963) of S *w.r.t.* the DNN v. It has been proven that the DNN's inference score $v(\boldsymbol{x})$ can be disentangled into the sum of effects of all potential concepts, as follows.

$$v(\boldsymbol{x}) = \sum\nolimits_{S \subseteq N} I(S|\boldsymbol{x}) \approx \sum\nolimits_{S \in \Omega_{\boldsymbol{x}}} I(S|\boldsymbol{x}) \qquad (2)$$

In particular, all interaction concepts can be further categorized into a set of **salient concepts** $S \in \Omega_{\boldsymbol{x}}$ with considerable effects $I(S|\boldsymbol{x})$, and a set of ignorable **noisy patterns** with almost zero effects $I(S|\boldsymbol{x}) \approx 0$.

Note that the Harsanyi dividend $I(S|\boldsymbol{x})$ also satisfies many desirable axioms/theorems, as introduced in Appendix A. The interaction effects can be further optimized using the trick of disentangling OR interactions[4] introduced in both (Li & Zhang, 2023) and Appendix H.4 of (Ren et al., 2021a), to pursue higher sparsity of interaction concepts.

3.2. Visualization of interaction concepts

In this section, we visualize interaction concepts extracted from point-cloud data and tabular data. Note that a sample in the ShapeNet dataset (Yi et al., 2016) usually contains 2500 3D points. To simplify the visualization, we simply consider 8-10 semantic parts on the point cloud $\boldsymbol{x}$, which has been provided by the dataset.[2] Each semantic part is

[1] For all tabular datasets and the image datasets (the *CelebA-eyeglasses* dataset and the *CUB-binary* dataset), the baseline value of each input variable was set as the mean value of this variable over all samples (Dabkowski & Gal, 2017). For grayscale digital images in the *MNIST-3* dataset, the baseline value of each pixel was set as zero (Ancona et al., 2019). For the point-cloud dataset, the baseline value was set as the center of the entire point cloud (Shen et al., 2021).

[2] For example, the ShapeNet dataset has provided the annotated parts for the *motorbike* category, including *gas tank*, *seat*, *handle*, *light*, *front wheel*, *back wheel*, *front frame*, *mid frame*, and *back*

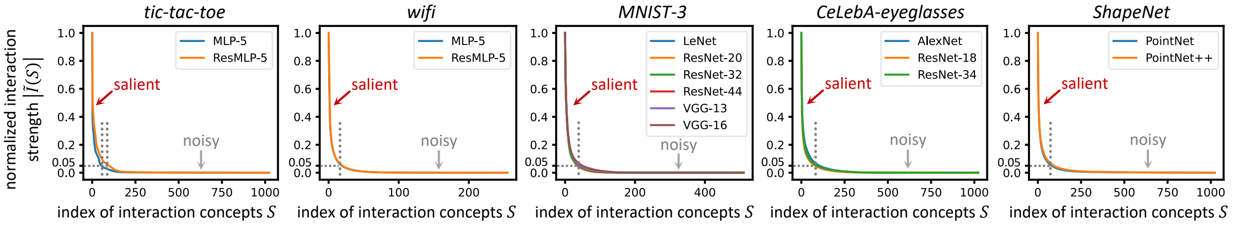


Figure 3. Normalized strength of interaction effects of different concepts in a descending order. DNNs trained for different tasks all encode sparse salient concepts.

taken as a "single" input variable to the DNN. In this way, we visualize concepts consisting of semantic parts.

Fig. 1 shows interaction concepts S and the corresponding effects $I(S|\boldsymbol{x})$ extracted by PointNet (Qi et al., 2017a) from different samples $\boldsymbol{x}$ in the ShapeNet dataset. We find that the interaction concept $S = \{$*light, front wheel, mid frame*$\}$ on five samples all makes positive effects $I(S|\boldsymbol{x}) > 0$ to the PointNet's output, whereas the interaction concept $S = \{$*handle, front wheel, front frame*$\}$ usually makes negative effects $I(S|\boldsymbol{x}) < 0$ to the PointNet's output. Similarly, Fig. 2 shows interaction concepts extracted from two tabular datasets, *i.e.*, the *wifi* dataset[3], and the *tic-tac-toe* dataset[3]. We visualize interactions between different *received signal strength indicatons* (RSSIs) in the *wifi* dataset, and interactions between *board states* in the *tic-tac-toe* dataset. We also find that the same interaction concept usually makes similar effects to the network output on different input samples, which supports the conclusion in Section 3.3.2.

3.3. Does a DNN really learn symbolic concepts?

Although (Ren et al., 2021a) have claimed that the metric $I(S|\boldsymbol{x})$ in Eq. (1) quantifies symbolic concepts encoded by a DNN, there is still no theory to guarantee a subset $S \subseteq N$ with a salient effect $I(S|\boldsymbol{x})$ faithfully represents a meaningful and transferable concept. Instead, we should not ignore the possibility that $I(S|\boldsymbol{x})$ is just a mathematical transformation that ensures $v(\boldsymbol{x}) = \sum_{S\subseteq N} I(S|\boldsymbol{x})$ in Eq. (2) on each specific sample. Therefore, in this study, we examine the counter-intuitive conjecture that a DNN learns symbolic concepts from the following four perspectives.

3.3.1. Sparsity of the encoded concepts

According to Eq. (2), the DNN may encode at most 2^n symbolic concepts in $2^N \triangleq \{S : S \subseteq N\}$ *w.r.t.* the 2^n different combinations of input variables. **However, a distinctive property of symbolism, which is different from connectionism, is that people usually would like to use *a small number of explicit* symbolic concepts to represent the knowledge, instead of using *extensive fuzzy* features.**

Thus, we hope to examine whether a DNN's inference score $v(\boldsymbol{x})$ on a specific sample can be summarized into effects of a small number of salient concepts $v(\boldsymbol{x}) \approx \sum_{S\in\Omega_{\boldsymbol{x}}} I(S|\boldsymbol{x})$, rather than using an exponential number of concepts *w.r.t.* all subsets $S \subseteq N$. To be precise, a faithful conceptual representation requires most concepts $S \subseteq N$ to be noisy patterns with negligible effects $I(S|\boldsymbol{x}) \approx 0$. Only a few salient concepts S in $\Omega_{\boldsymbol{x}}$ make considerable effects $I(S|\boldsymbol{x})$.

To this end, Ren et al. (2021a) have made a preliminary attempt to explain a DNN's inference score $v(\boldsymbol{x})$ on a specific sample $\boldsymbol{x}$ as interaction effects $I(S|\boldsymbol{x})$ of a small number of concepts. Specifically, they used a few top-ranked salient interaction concepts to explain the inference score of LSTMs (Hochreiter & Schmidhuber, 1997) and CNNs (Kim, 2014) trained for sentiment classification and linguistic acceptance classification tasks on the SST-2 dataset (Socher et al., 2013) and the CoLA dataset (Warstadt et al., 2019).

Experiments. In this paper, we further examined whether most DNNs, which were trained for much more diverse tasks on different datasets, all encoded very sparse salient concepts. To this end, we trained various DNNs[3] on tabular datasets (the *tic-tac-toe* dataset[3] and the *wifi* dataset[3]), image datasets (the *MNIST-3* dataset[3] and the *CelebA-eyeglasses* dataset[3]), and a point-cloud dataset (the *ShapeNet* dataset[3]). The interaction effects can be further optimized using the trick of disentangling OR interactions[4] introduced in both (Li & Zhang, 2023) and Appendix H.4 of (Ren et al., 2021a), to pursue higher sparsity of interaction concepts. Fig. 3 shows the normalized interaction strength of different concepts $|\tilde{I}(S|\boldsymbol{x})| \triangleq |I(S|\boldsymbol{x})| / \max_{S'} |I(S'|\boldsymbol{x})|$ in a descending order for each DNN. Each curve shows the strength averaged over different samples in the dataset. We found that most concepts had little effects on the output $|I(S|\boldsymbol{x})| \approx 0$, which verified the sparsity of the encoded concepts.

Salient concepts. According to the above experiments, we can define the set of salient concepts as $\Omega_{\boldsymbol{x}} = \{S : |I(S|\boldsymbol{x})| > \tau\}$, subject to $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$. As Fig. 3 shows, there were only about 20-80 salient concepts

frame. Please see Appendix B.2 for details on the annotation of semantic parts.

[3] Please see the *experimental settings* paragraph at the end of Section 3 for details on datasets and DNNs.

[4] This study also extracts the OR interaction, which is proved to be a specific AND interaction.

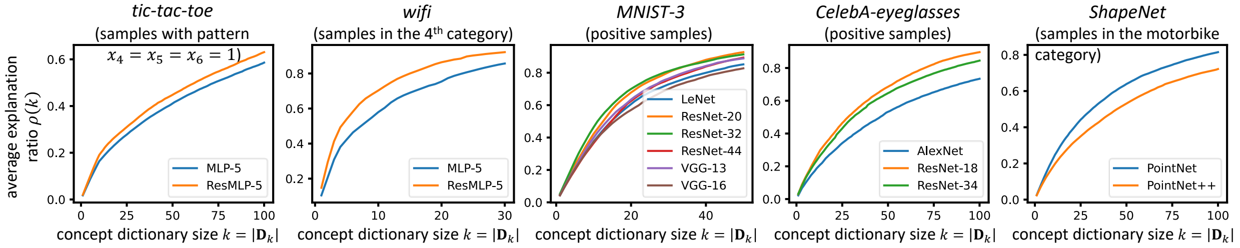


Figure 4. The change of the average explanation ratio $\rho(k)$ along with the size k of the concept dictionary $\mathbf{D}_k$.

extracted from an input sample, and all other concepts have ignorable effects on the network output.

3.3.2. Transferability over different samples

Beyond sparsity, the transferability of concepts is more important. If $I(S|\boldsymbol{x})$ is just a tricky mathematical transformation on $v(\boldsymbol{x}_S)$ without representing meaningful concepts, then each salient concept $I(S|\boldsymbol{x})$ extracted from the input sample $\boldsymbol{x}$ probably cannot be transferred to another input sample, *i.e.,* we cannot extract the same salient concept consisting of variables in S in the second sample, due to sparsity of salient concepts.

Therefore, in this section, **we aim to verify whether concepts extracted from a sample can be transferred to other samples.** This task is actually equivalent to checking whether there exists a common concept dictionary, which contains most salient concepts extracted from different samples in the same category.

Given a well-trained DNN, we construct a relatively small dictionary $\mathbf{D}_k \subseteq 2^N$ containing the top-k frequent concepts in different samples, and check whether such a dictionary contains most salient concepts in $\Omega_{\boldsymbol{x}}$ extracted from each sample $\boldsymbol{x}$. The concept dictionary $\mathbf{D}_k$ is constructed based on a greedy strategy. Specifically, we first extract a set of salient concepts $\Omega_{\boldsymbol{x}}$ from each input sample $\boldsymbol{x}$. Then, we compute the frequency of each concept S being a salient concept over different samples. Finally, the concept dictionary $\mathbf{D}_k$ is constructed to contain the top-k interaction concepts with the highest frequencies.

Then, we design the metric $\rho(k) \triangleq \mathbb{E}_{\boldsymbol{x}}[|\mathbf{D}_k \cap \Omega_{\boldsymbol{x}}|/|\Omega_{\boldsymbol{x}}|]$ to evaluate the average ratio of concepts extracted from an input sample that is covered by the concept dictionary $\mathbf{D}_k$. Theoretically, if we construct a larger dictionary $\mathbf{D}_k$ with more concepts (a larger k value), then the dictionary can explain more concepts.

Experiments. We conducted experiments to show whether there existed a small concept dictionary that could explain most concepts encoded by the DNN. Specifically, we constructed a concept dictionary to explain samples in a certain category in each dataset[5]. In this experiment, we temporarily extracted salient concepts using $\tau = 0.1 \cdot \max_S |I(S|\boldsymbol{x})|$ to construct $\Omega_{\boldsymbol{x}}$ [6]. Fig. 4 shows the increase of the average explanation ratio $\rho(k)$ along with the increasing size k of the concept dictionary $\mathbf{D}_k$. We found that there usually existed a concept dictionary consisting of 30-100 concepts, which could explain more than 60%-80% salient concepts encoded by the DNN. Besides, Fig. 1(right) and Fig. 2(right) also show histograms of effects $I(S|\boldsymbol{x})$ over different samples[5], where S was extracted as a salient concept. We found that the same interaction concept usually made similar effects on different samples. This verified that the DNN learned transferable concepts over different samples.

3.3.3. Transferability across different DNNs

In addition to sample-wise transferability of concepts, another aspect is model-wise transferability. If the concepts extracted from an input sample really represent meaningful knowledge for the task, then these concepts are supposed to be stably learned by different DNNs towards the same task, although we cannot directly align intermediate-layer features between different DNNs.

Therefore, in this section, **we aim to verify whether concepts extracted from a DNN can be transferred to another DNN trained for the same task.** In other words, we actually check whether salient concepts encoded by one DNN are also encoded by another DNN learned for the same task. To this end, let us consider two DNNs, v_1 and v_2, trained for the same task. Given an input sample $\boldsymbol{x}$, let $\Omega_{\boldsymbol{x}}^{v_1}$ and $\Omega_{\boldsymbol{x}}^{v_2}$ denote the sets of salient concepts extracted by v_1 and v_2 from the input sample $\boldsymbol{x}$, respectively. We evaluate the the ratio of concepts in $\Omega_{\boldsymbol{x}}^{v_1}$ encoded by v_1, which are also encoded by v_2 in $\Omega_{\boldsymbol{x}}^{v_2}$, *i.e.* $\gamma(\Omega_{\boldsymbol{x}}^{v_1}|\Omega_{\boldsymbol{x}}^{v_2}) \triangleq |\Omega_{\boldsymbol{x}}^{v_1} \cap \Omega_{\boldsymbol{x}}^{v_2}|/|\Omega_{\boldsymbol{x}}^{v_1}|$, to measure the transferability of salient concepts in $\Omega_{\boldsymbol{x}}^{v_1}$. A larger ratio $\gamma(\Omega_{\boldsymbol{x}}^{v_1}|\Omega_{\boldsymbol{x}}^{v_2})$ indicates that the extracted salient concepts have higher

[5] In this paper, when we needed to analyze of samples in a specific category, we used positive samples in the *MNIST-3*, *CelebA-eyeglasses*, and *CUB-binary* datasets, samples in the 4th category in the *wifi dataset*, and samples in the "motorbike" category in the ShapeNet dataset. For the *tic-tac-toe* dataset, since there exists eight sub-categories among positive samples, we used samples in the sub-category with the pattern $x_4 = x_5 = x_6 = 1$.

[6] Here, we increased the threshold from $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$ to $\tau = 0.1 \cdot \max_S |I(S|\boldsymbol{x})|$ to analyze those highly salient concepts. Appendix C.1 shows results computed by using the vanilla threshold $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$ to compute $\Omega_{\boldsymbol{x}}$.

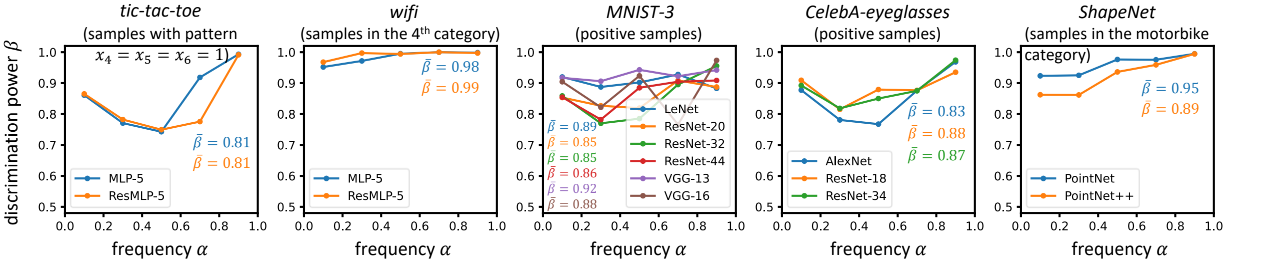


Figure 5. The average discrimination power of concepts in different frequency intervals, *i.e.* $\alpha \in (0.0, 0.2], (0.2, 0.4], ..., (0.8, 1.0]$. The weighted average discrimination power $\bar{\beta}$ over concepts of all frequencies is shown beside the curve.

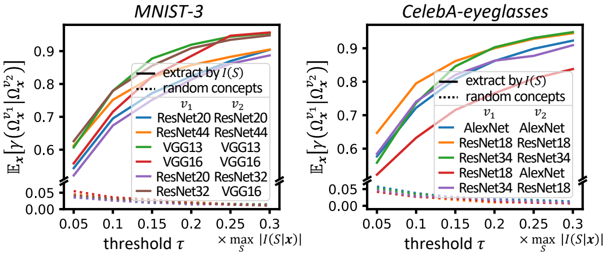


Figure 6. Concepts extracted by a higher threshold τ (*i.e.* concepts with more significant effects $I(S|\boldsymbol{x})$) usually have higher transferability across different DNNs.

transferability across different DNNs.

Experiments. In this experiment, we examined the transferability of concepts in both the case when DNNs v_1 and v_2 had the same network architecture but were trained with different parameter initializations, and the case when v_1 and v_2 had different network architectures. Then, given each sample $\boldsymbol{x}$, $\Omega_{\boldsymbol{x}}^{v_2}$ contains all salient concepts with interaction strength $I_{v_2}(S|\boldsymbol{x}) \geq 0.05 \cdot \max_S |I_{v_2}(S|\boldsymbol{x})|$, as defined in Section 3.3.1. Whereas, we used different thresholds τ ranging from $\tau = 0.05 \cdot \max_S |I_{v_1}(S|\boldsymbol{x})|$ to $\tau = 0.3 \cdot \max_S |I_{v_1}(S|\boldsymbol{x})|$ to generate different sets $\Omega_{\boldsymbol{x}}^{v_1}$. A larger τ value usually generated a smaller set of salient concepts with more significant effects. These concepts were more likely to be stably learned by different DNNs. Fig. 6 shows that concepts with higher saliency usually exhibited higher transferability from DNN v_1 to DNN v_2. This indicated that more salient concepts were more likely to be stably learned by different DNNs, which was aligned with intuition. As a baseline for comparison, we also randomly extracted two sets of concepts $\tilde{\Omega}_{\boldsymbol{x}}^{v_1}$ and $\tilde{\Omega}_{\boldsymbol{x}}^{v_2}$ from all the 2^n interaction concepts, which had the same size as $\Omega_{\boldsymbol{x}}^{v_1}$ and $\Omega_{\boldsymbol{x}}^{v_2}$, *i.e.* $|\Omega_{\boldsymbol{x}}^{v_1}| = |\tilde{\Omega}_{\boldsymbol{x}}^{v_1}|$ and $|\Omega_{\boldsymbol{x}}^{v_2}| = |\tilde{\Omega}_{\boldsymbol{x}}^{v_2}|$. Fig. 6 shows that the transferability $\mathbb{E}_{\boldsymbol{x}}[\gamma(\Omega_{\boldsymbol{x}}^{v_1}|\Omega_{\boldsymbol{x}}^{v_2})]$ of concepts extracted by $I(S|\boldsymbol{x})$ increased in the range of 0.5-0.95 along with the increase of τ. In comparison, the transferability $\mathbb{E}_{\boldsymbol{x}}[\gamma(\tilde{\Omega}_{\boldsymbol{x}}^{v_1}|\tilde{\Omega}_{\boldsymbol{x}}^{v_2})]$ of random concepts was usually less than 0.05. This verified the high transferability of concepts across different DNNs.

3.3.4. Discrimination power of concepts

Furthermore, if a DNN encodes faithful symbolic concepts, then these concepts are supposed to exhibit certain discrimination power in the classification task. In other words, **for each concept S, if the concept is saliently activated on a set of samples, then interaction effects $I(S)$ of the same concept are supposed to push the classification of these samples towards a certain category in most cases.** Note that different concepts extracted from a sample may push the sample towards different categories, and the classification is the result of the competition between these concepts.

In order to verify the above discrimination power of concepts, in this section, we extract the concept S from m different input samples $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_m$ in the same category c, and check whether this concept consistently exhibits a positive (or negative) interaction effects $I(S)$ on the m samples. If the concept S pushes the classification of most of the m samples towards the target category, *i.e.*, $I(S|\boldsymbol{x}_i) > 0$ (or opposite to the target category, *i.e.*, $I(S|\boldsymbol{x}_i) < 0$), then the discrimination power of the concept S is high. On the other hand, if the concept S pushes half of the samples towards the positive direction $I(S|\boldsymbol{x}_i) > 0$, but pushes the other half towards the negative direction $I(S|\boldsymbol{x}_i) < 0$, then the discrimination power of the concept S is low.

Specifically, we use the following metric to measure the discrimination power of concept S among the above m samples in category c. Let $\Omega_{\boldsymbol{x}_i}$ denote a set of salient concepts extracted from the sample $\boldsymbol{x}_i$. Then, $m_S^+ \triangleq \sum_i \mathbb{1}_{S \in \Omega_{\boldsymbol{x}_i}} \cdot \mathbb{1}_{I(S|\boldsymbol{x}_i)>0}$ denote the number of samples where the concept S makes a salient and positive effect on the classification score. Similarly, we can define $m_S^- \triangleq \sum_i \mathbb{1}_{S \in \Omega_{\boldsymbol{x}_i}} \cdot \mathbb{1}_{I(S|\boldsymbol{x}_i)<0}$ to denote the number of samples where the concept S makes a salient and negative effect on the classification score. In this way, the discrimination power of a salient concept S can be measured as $\beta(S) = \max(m_S^+, m_S^-)/(m_S^+ + m_S^-)$. A larger value of $\beta(S)$ indicates a higher discrimination power of the concept S.

Experiments. Note that different concepts are of different importance in the classification of a category. Some concepts frequently appear in different samples and make salient effects, while other concepts only appear in very few concepts. Therefore, we use the frequency of the concept $\alpha(S)$ as a weight to compute the average discrimination power of all concepts, which is given as $\bar{\beta} \triangleq \sum_S [\alpha(S) \cdot \beta(S)] / \sum_S [\alpha(S)]$. The frequency of a concept is

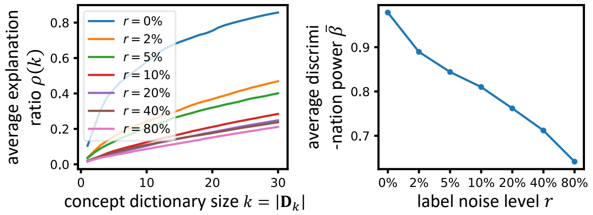


Figure 7. The (left) transferability and (right) discrimination power of concepts decreased when we added more label noises.

defined as $\alpha(S) \triangleq (m_S^+ + m_S^-)/m$. The selection of datasets and the training of DNNs are introduced in the *experimental settings* paragraph at the end of Section 3. Fig. 5 shows the average discrimination power of concepts in different frequency intervals. We found that the average discrimination power $\bar{\beta}$ of concepts was usually higher than 0.8, which verified the discrimination power of extracted concepts.

3.4. When DNNs do not learn transferable concepts

It is worth noting that all the above work just conducts experiments to show the concept-emerging phenomenon in different DNNs for different tasks. We do not provide, or there may even do not exist, a theoretical proof for such a concept-emerging phenomenon, although the concept-emerging phenomenon **does exist** in DNNs for most applications. Therefore, in this subsection, we would like to discuss the following three extreme cases, in which a DNN does not learn symbolic concepts.

In the three extreme cases, DNNs may either collapse to simple models close to linear regressions, or learn non-transferable indiscriminative concepts, although the network output can still be represented as the sum of interaction effects of these concepts.

• **Case 1: When there exists label noise.** If the ground-truth label for classification is incorrectly annotated on some samples, then the DNN usually has to memorize each incorrectly-labeled training sample for classification without summarizing many common features from such chaotic annotations. Thus, in this case, the DNN usually encodes more non-transferable concepts.

Experimental verification. In this experiment, we constructed datasets with noisy labels to check whether DNNs trained on these datasets did not learn transferable concepts with high discrimination power. To this end, given a clean dataset, we first selected and randomly labeled a certain portion r of training samples in the dataset, so as to train a DNN. Specifically, we constructed a series of datasets by assigning different ratios r of samples with incorrect labels. We used the *wifi* dataset[3] to construct new datasets by adding different ratios r of noisy labels. Then, we trained an MLP-5 network[3] on each of these datasets. We examined the transferability and discrimination power of the extracted concepts[5] on each MLP-5 network. We extracted concept dictionaries of different sizes based on each MLP-5 network (please see Section 3.3.2 for details). Fig. 7(left) shows that if there was significant label noise in the dataset, the concept dictionary usually explained much fewer concepts encoded by the network, which indicated low transferability of the learned concepts. Besides, Fig. 7(right) shows the average discrimination power $\bar{\beta}$ of the extracted concepts also decreased when we assigned more training samples with random labels. This verified that the DNN usually could not learn transferable and discriminative concepts from samples that were incorrectly labeled.

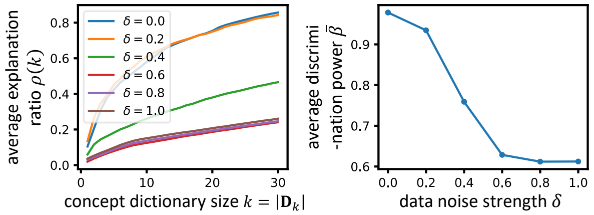


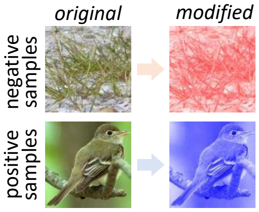
Figure 8. The (left) transferability and (right) discrimination power of concepts decreased when input data were noisy.

• **Case 2: When input samples are noisy.** In fact, this case can be extended to a more general scenario, *i.e.*, when the task is difficult to learn, there is no essential difference between the difficult data and noisy data for the DNN. Specifically, when training samples are noisy and lack meaningful patterns, it is difficult for a DNN to learn transferable concepts from noisy training samples.

Experimental verification. In this experiment, we injected noise into training samples to examine whether DNNs trained on datasets with noisy samples did not learn transferable concepts. Just like the experiment in "Case 1," we constructed such datasets by corrupting a clean dataset. To this end, we added Gaussian noises $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ to each input sample $\boldsymbol{x}$ in the clean dataset by modifying it to $(1-\delta) \cdot \boldsymbol{x} + \delta \cdot \boldsymbol{\epsilon}$, where $\delta \in [0, 1]$ denotes the strength of noise injected into the sample $\boldsymbol{x}$. Each dimension of the clean sample $\boldsymbol{x}$ was normalized to unit variance over the dataset beforehand.

We constructed a series of datasets by injecting noises of different strength δ into samples in the *wifi* dataset[3]. We trained MLP-5's[3] based on these datasets. Just like experiments in "Case 1," we examined the transferability and discrimination power of concepts. Fig. 8(left) shows that the transferability of concepts was usually low when the DNN was learned from noisy input data. Fig. 8(right) shows that the average discrimination power $\bar{\beta}$ of concepts decreased along with the increasing strength δ of injected noise. This verified that DNN usually did not learn transferable and discriminative concepts when input data were noisy.

• **Case 3: When the task has a simple shortcut solution.** In the above two cases, both label noise and data noise corrupted the original discriminative patterns in each category,



Relative concept strength κ	Dataset	
	the original *CUB-binary* dataset	the modified *CUB-binary* dataset
AlexNet	0.32	0.05
ResNet-18	0.20	0.07
VGG-13	0.24	0.04

Figure 9. (left) We constructed a dataset where the "*color*" information was a shortcut solution. (right) The relative concept strength κ extracted from DNNs trained on different datasets.

thus making the DNN unlikely to learn transferable concepts. In comparison, here, let us discuss a new case, *i.e.*, even if there exist meaningful patterns in training data, the DNN may still not learn these concepts.

To be precise, if a classification task can be conducted with some shortcut solutions without requiring the DNN to encode complex concepts, then the DNN probably converges to the shortcut solution. For example, in an image classification task, if pixel-wise colors are sufficient to conduct the image-classification task, then the DNN is more likely to only use the color information for classification without modeling complex visual concepts. The simple shortcut solution usually prevents the DNN from summarizing complex visual concepts.

Experimental verification. We constructed a dataset for image classification, where the *"color"* information was a shortcut solution for the task. Specifically, we modified images in the *CUB-binary* dataset[3], such that all negative samples were red-colored background regions, and all positive samples were blue-colored foreground birds, as shown in Fig. 9(left). We trained AlexNet, ResNet-18, and VGG-13 on both the original dataset and the modified dataset. Compared with DNNs learned on the original dataset, DNNs learned on the modified dataset were more likely to simply used the color information for classification. We used the metric $\kappa \triangleq \mathbb{E}_{\boldsymbol{x}}[\sum_{S\in\Omega_{\boldsymbol{x}},|S|\geq 2}|I(S)|/\sum_{S\in\Omega_{\boldsymbol{x}}}|I(S)|]$ to measure the relative strength of all concepts consisting of multiple variables. Fig. 9(right) shows that the κ values were usually low for DNNs learned to classify red-colored backgrounds and blue-colored birds. This indicated that the DNN collapsed to a simple model without encoding interactions between different image patches when the task had a simple shortcut solution.

Experimental settings. For tabular datasets, we used the UCI tic-tac-toe endgame dataset (Dua & Graff, 2017) for binary classification, and used the UCI wireless indoor localization dataset (Dua & Graff, 2017) for multi-category classification. These datasets were termed *tic-tac-toe* and *wifi* for simplicity. We trained the following two MLPs on each tabular dataset. *MLP-5* contained five fully connected layers with 100 neurons in each hidden layer (Ren et al., 2021a). *ResMLP-5* was constructed by adding a skip connection to each layer of an *MLP-5*. For image data, we used the following three datasets. We took images corresponding to digit "three" in the MNIST dataset (LeCun, 1998) as positive samples, and took other images as negative samples to train DNNs. We took images with the attribute "eyeglasses" in the CelebA dataset (Liu et al., 2015) as positive samples, and took other images as negative samples to train DNNs. We trained DNNs to classify birds in bounding boxes in the CUB-200-2011 dataset (Wah et al., 2011) from randomly cropped background regions around the bird. These three datasets were termed *MNIST-3*, *CelebA-eyeglasses*, and *CUB-binary* for short. We trained LeNet (LeCun et al., 1998), AlexNet (Krizhevsky et al., 2017), ResNet-18/20/32/34/44 (He et al., 2016), VGG-13/16 (Simonyan & Zisserman, 2014) on these image datasets. Based on the ShapeNet dataset (Yi et al., 2016) for the classification of 3D point clouds, we trained PointNet (Qi et al., 2017a) and PointNet++ (Qi et al., 2017b). Please see Appendix B.1 for the classification accuracy of the above DNNs.

4. Conclusion

In this paper, we have analyzed the interaction concepts encoded by a DNN. Specifically, we quantitatively examine the concept-emerging phenomenon of a DNN from four perspectives. Extensive empirical studies have verified that a well-trained DNN usually encodes sparse, transferable, and discriminative interaction concepts. Our experiments also prove the faithfulness of the interaction concepts extracted from DNNs. Besides, we also discussed three cases in which a DNN is unlikely to learn transferable concepts.

Acknowledgement. This work is partially supported by the National Nature Science Foundation of China (62276165), National Key R&D Program of China (2021ZD0111602), Shanghai Natural Science Foundation (21JC1403800,21ZR1434600), National Nature Science Foundation of China (U19B2043). This work is also partially supported by Huawei Technologies Inc.

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

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

Bau, D., Zhou, B., Khosla, A., Oliva, A., and Torralba, A. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE* *conference on computer vision and pattern recognition*, pp. 6541–6549, 2017.

Che, Z., Purushotham, S., Khemani, R., and Liu, Y. Interpretable deep models for icu outcome prediction. In *AMIA annual symposium proceedings*, volume 2016, pp. 371. American Medical Informatics Association, 2016.

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

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

Covert, I., Lundberg, S. M., and Lee, S.-I. Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33, 2020.

Dabkowski, P. and Gal, Y. Real time image saliency for black box classifiers. *arXiv preprint arXiv:1705.07857*, 2017.

Deng, H., Ren, Q., Zhang, H., and Zhang, Q. Discovering and explaining the representation bottleneck of DNNs. In *International Conference on Learning Representations*, 2022a.

Deng, H., Zou, N., Du, M., Chen, W., Feng, G., Yang, Z., Li, Z., and Zhang, Q. Understanding and unifying fourteen attribution methods with taylor interactions. *arXiv preprint*, 2022b.

Dosovitskiy, A. and Brox, T. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4829–4837, 2016.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Fong, R. C. and Vedaldi, A. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE international conference on computer vision*, pp. 3429–3437, 2017.

Frosst, N. and Hinton, G. Distilling a neural network into a soft decision tree. *arXiv preprint arXiv:1711.09784*, 2017.

Grabisch, M. and Roubens, M. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of game theory*, 28(4):547–565, 1999.

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

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

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Janizek, J. D., Sturmfels, P., and Lee, S.-I. Explaining explanations: Axiomatic feature interactions for deep networks. *arXiv preprint arXiv:2002.04138*, 2020.

Jin, X., Wei, Z., Du, J., Xue, X., and Ren, X. Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. In *International Conference on Learning Representations*, 2019.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pp. 2668–2677. PMLR, 2018.

Kim, Y. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1181. URL https://aclanthology.org/D14-1181.

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

LeCun, Y. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

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

Li, M. and Zhang, Q. Defining and quantifying and-or interactions for faithful and concise explanation of DNNs. *arXiv preprint arXiv:2304.13312*, 2023.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017a. URL https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pp. 4768–4777, 2017b.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Murdoch, W. J., Liu, P. J., and Yu, B. Beyond word importance: Contextual decomposition to extract interactions from lstms. In *International Conference on Learning Representations*, 2018.

Qi, C. R., Su, H., Mo, K., and Guibas, L. J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017a.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017b.

Ren, J., Li, M., Chen, Q., Deng, H., and Zhang, Q. Towards axiomatic, hierarchical, and symbolic explanation for deep models. *arXiv preprint arXiv:2111.06206v5*, 2021a. URL https://arxiv.org/pdf/2111.06206v5.pdf.

Ren, J., Zhang, D., Wang, Y., Chen, L., Zhou, Z., Chen, Y., Cheng, X., Wang, X., Zhou, M., Shi, J., et al. Towards a unified game-theoretic view of adversarial perturbations and robustness. *Advances in Neural Information Processing Systems*, 34:3797–3810, 2021b.

Ren, J., Zhou, Z., Chen, Q., and Zhang, Q. Can we faithfully represent absence states to compute Shapley values on a DNN? In *International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=YV8tP7bW6Kt.

Ren, Q., Deng, H., Chen, Y., Lou, S., and Zhang, Q. Bayesian neural networks avoid encoding complex and perturbation-sensitive concepts. In *International Conference on Machine Learning*, 2023b.

Ren, Q., Gao, J., Shen, W., and Zhang, Q. Where we have arrived in proving the emergence of sparse symbolic concepts in AI models. *arXiv preprint arXiv:2305.01939*, 2023c.

Ribeiro, M. T., Singh, S., and Guestrin, C. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Shapley, L. S. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

Shen, W., Ren, Q., Liu, D., and Zhang, Q. Interpreting representation quality of DNNs for 3d point cloud processing. *Advances in Neural Information Processing Systems*, 34: 8857–8870, 2021.

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

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Singh, C., Murdoch, W. J., and Yu, B. Hierarchical interpretations for neural network predictions. In *International Conference on Learning Representations*, 2018.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Sorokina, D., Caruana, R., Riedewald, M., and Fink, D. Detecting statistical interactions with additive groves of trees. In *Proceedings of the 25th international conference on Machine learning*, pp. 1000–1007, 2008.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *International conference on machine learning*, pp. 3319–3328. PMLR, 2017.

Sundararajan, M., Dhamdhere, K., and Agarwal, A. The shapley taylor interaction index. In *International Conference on Machine Learning*, pp. 9259–9268. PMLR, 2020.

Tan, S., Caruana, R., Hooker, G., Koch, P., and Gordo, A. Learning global additive explanations for neural nets using model distillation. *arXiv preprint arXiv:1801.08640*, 2018.

Tsai, C.-P., Yeh, C.-K., and Ravikumar, P. Faith-shap: The faithful shapley interaction index. *arXiv preprint arXiv:2203.00870*, 2022.

Vaughan, J., Sudjianto, A., Brahimi, E., Chen, J., and Nair, V. N. Explainable neural networks based on additive index models. *arXiv preprint arXiv:1806.01933*, 2018.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. 2011.

Wang, X., Lin, S., Zhang, H., Zhu, Y., and Zhang, Q. Interpreting attributions and interactions of adversarial attacks. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 1075–1084. IEEE, 2021a.

Wang, X., Ren, J., Lin, S., Zhu, X., Wang, Y., and Zhang, Q. A unified approach to interpreting and boosting adversarial transferability. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, 2021b.

Warstadt, A., Singh, A., and Bowman, S. R. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641, 2019.

Wu, M., Hughes, M. C., Parbhoo, S., Zazzi, M., Roth, V., and Doshi-Velez, F. Beyond sparsity: Tree regularization of deep models for interpretability. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Yi, L., Kim, V. G., Ceylan, D., Shen, I.-C., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., and Guibas, L. A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics (ToG)*, 35(6):1–12, 2016.

Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., and Lipson, H. Understanding neural networks through deep visualization. In *International Conference on Machine Learning*, 2015.

Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

Zhang, D., Zhang, H., Zhou, H., Bao, X., Huo, D., Chen, R., Cheng, X., Wu, M., and Zhang, Q. Building interpretable interaction trees for deep NLP models. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 14328–14337. AAAI Press, 2021a.

Zhang, H., Li, S., Ma, Y., Li, M., Xie, Y., and Zhang, Q. Interpreting and boosting dropout from a game-theoretic view. In *International Conference on Learning Representations*, 2021b.

Zhang, H., Xie, Y., Zheng, L., Zhang, D., and Zhang, Q. Interpreting multivariate shapley interactions in DNNs. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 10877–10886. AAAI Press, 2021c.

Zhang, Q., Cao, R., Shi, F., Wu, Y. N., and Zhu, S.-C. Interpreting cnn knowledge via an explanatory graph. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Zhang, Q., Wang, X., Ren, J., Cheng, X., Lin, S., Wang, Y., and Zhu, X. Proving common mechanisms shared by twelve methods of boosting adversarial transferability. *arXiv preprint arXiv:2207.11694*, 2022a.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022b.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. Object detectors emerge in deep scene cnns. *In ICLR*, 2015.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.

Zhou, H., Zhang, H., Deng, H., Liu, D., Shen, W., Chan, S.-H., and Zhang, Q. Concept-level explanation for the generalization of a DNN. *arXiv preprint arXiv:2302.13091*, 2023.

A. Axioms and theorems of the Harsanyi dividend

As mentioned in Section 3.1 of the paper, the definition for an interaction concept S in Eq. (1) can be understood as the Harsanyi dividend of the set of variables in S *w.r.t.* the DNN v. In fact, the Harsanyi dividend $I(S|\boldsymbol{x})$ also satisfies many desirable axioms and theorems, as follows.

The Harsanyi dividend $I(S|\boldsymbol{x})$ satisfies seven desirable axioms in game theory (Ren et al., 2021a), including the *efficiency, linearity, dummy, symmetry, anonymity, recursive* and *interaction distribution* axioms.

(1) *Efficiency axiom.* The output score of a model can be decomposed into interaction effects of different patterns, *i.e.* $v(\boldsymbol{x}) = \sum_{S\subseteq N} I(S|\boldsymbol{x})$.

(2) *Linearity axiom.* If we merge output scores of two models w and v as the output of model u, *i.e.* $\forall S \subseteq N,\ u(\boldsymbol{x}_S) = w(\boldsymbol{x}_S) + v(\boldsymbol{x}_S)$, then their interaction effects $I_v(S|\boldsymbol{x})$ and $I_w(S|\boldsymbol{x})$ can also be merged as $\forall S \subseteq N, I_u(S|\boldsymbol{x}) = I_v(S|\boldsymbol{x}) + I_w(S|\boldsymbol{x})$.

(3) *Dummy axiom.* If a variable $i \in N$ is a dummy variable, *i.e.* $\forall S \subseteq N\backslash\{i\}, v(\boldsymbol{x}_{S\cup\{i\}}) = v(\boldsymbol{x}_S) + v(\boldsymbol{x}_{\{i\}})$, then it has no interaction with other variables, $\forall\emptyset \neq T \subseteq N\backslash\{i\}, I(T \cup \{i\}|\boldsymbol{x}) = 0$.

(4) *Symmetry axiom.* If input variables $i, j \in N$ cooperate with other variables in the same way, $\forall S \subseteq N\backslash\{i, j\}, v(\boldsymbol{x}_{S\cup\{i\}}) = v(\boldsymbol{x}_{S\cup\{j\}})$, then they have same interaction effects with other variables, $\forall S \subseteq N\backslash\{i, j\}, I(S \cup \{i\}|\boldsymbol{x}) = I(S \cup \{j\}|\boldsymbol{x})$.

(5) *Anonymity axiom.* For any permutations π on N, we have $\forall S\subseteq N, I_v(S|\boldsymbol{x}) = I_{\pi v}(\pi S|\boldsymbol{x})$, where $\pi S \triangleq \{\pi(i)|i\in S\}$, and the new model πv is defined by $(\pi v)(\boldsymbol{x}_{\pi S}) = v(\boldsymbol{x}_S)$. This indicates that interaction effects are not changed by permutation.

(6) *Recursive axiom.* The interaction effects can be computed recursively. For $i \in N$ and $S \subseteq N\backslash\{i\}$, the interaction effect of the pattern $S\cup\{i\}$ is equal to the interaction effect of S with the presence of i minus the interaction effect of S with the absence of i, *i.e.* $\forall S\subseteq N\backslash\{i\}, I(S\cup\{i\}|\boldsymbol{x}) = I(S|i \text{ is always present}, \boldsymbol{x}) - I(S|\boldsymbol{x})$. $I(S|i \text{ is always present}, \boldsymbol{x})$ denotes the interaction effect when the variable i is always present as a constant context, *i.e.* $I(S|i \text{ is always present}, \boldsymbol{x}) = \sum_{L\subseteq S}(-1)^{|S|-|L|} \cdot v(\boldsymbol{x}_{L\cup\{i\}})$.

(7) *Interaction distribution axiom.* This axiom characterizes how interactions are distributed for "interaction functions" (Sundararajan et al., 2020). An interaction function v_T parameterized by a subset of variables T is defined as follows. $\forall S \subseteq N$, if $T \subseteq S$, $v_T(\boldsymbol{x}_S) = c$; otherwise, $v_T(\boldsymbol{x}_S) = 0$. The function v_T models pure interaction among the variables in T, because only if all variables in T are present, the output value will be increased by c. The interactions encoded in the function v_T satisfies $I(T|\boldsymbol{x}) = c$, and $\forall S \neq T$, $I(S|\boldsymbol{x}) = 0$.

The Harsanyi dividend $I(S|\boldsymbol{x})$ can also explain the elementary mechanism of existing game-theoretic metrics (Ren et al., 2021a), including *the Shapley value*, *the Shapley interaction index*, and *the Shapley-Taylor interaction index*.

(1) *Connection to the Shapley value (Shapley, 1953).* Let $\phi(i|\boldsymbol{x})$ denote the Shapley value of an input variable i, given the input sample $\boldsymbol{x}$. Then, the Shapley value $\phi(i|\boldsymbol{x})$ can be explained as the result of uniformly assigning attributions of each Harsanyi dividend to each involving variable i, *i.e.*, $\phi(i|\boldsymbol{x}) = \sum_{S\subseteq N\backslash\{i\}} \frac{1}{|S|+1} I(S \cup \{i\}|\boldsymbol{x})$. This also proves that the Shapley value is a fair assignment of attributions from the perspective of Harsanyi dividend.

(2) *Connection to the Shapley interaction index (Grabisch & Roubens, 1999).* Given a subset of variables $T \subseteq N$ in an input sample $\boldsymbol{x}$, the Shapley interaction index $I^{\text{Shapley}}(T|\boldsymbol{x})$ can be represented as $I^{\text{Shapley}}(T|\boldsymbol{x}) = \sum_{S\subseteq N\backslash T} \frac{1}{|S|+1} I(S \cup T|\boldsymbol{x})$. In other words, the index $I^{\text{Shapley}}(T|\boldsymbol{x})$ can be explained as uniformly allocating $I(S'|\boldsymbol{x})$ s.t. $S' = S \cup T$ to the compositional variables of S', if we treat the coalition of variables in T as a single variable.

(3) *Connection to the Shapley Taylor interaction index (Sundararajan et al., 2020).* Given a subset of variables $T \subseteq N$ in an input sample $\boldsymbol{x}$, the k-th order Shapley Taylor interaction index $I^{\text{Shapley-Taylor}}(T|\boldsymbol{x})$ can be represented as weighted sum of interaction effects, *i.e.*, $I^{\text{Shapley-Taylor}}(T|\boldsymbol{x}) = I(T|\boldsymbol{x})$ if $|T| < k$; $I^{\text{Shapley-Taylor}}(T|\boldsymbol{x}) = \sum_{S\subseteq N\backslash T} \binom{|S|+k}{k}^{-1} I(S \cup T|\boldsymbol{x})$ if $|T| = k$; and $I^{\text{Shapley-Taylor}}(T|\boldsymbol{x}) = 0$ if $|T| > k$.

B. Experimental details

B.1. Accuracy of DNNs

In this paper, we conducted experiments on various DNNs trained on different types of datasets, including tabular datasets, image datasets, and a point-cloud dataset. Table 1 reports the classification accuracy of DNNs trained on the above datasets.

Table 1. Classification accuracy of different DNNs.

Dataset	DNN					
tic-tac-toe	MLP-5 100%	ResMLP-5 100%				
wifi	MLP-5 97.75%	ResMLP-5 97.75%				
MNIST-3	LeNet 99.99%	ResNet-20 100%	ResNet-32 100%	ResNet-44 100%	VGG-13 100%	VGG-16 100%
CelebA-eyeglasses	AlexNet 99.53%	ResNet-18 99.66%	VGG-13 99.65%			
CUB-binary	AlexNet 95.67%	ResNet-18 96.41%	ResNet-34 96.43%			
the modified *CUB-binary* dataset	AlexNet 100%	ResNet-18 100%	ResNet-34 100%			
ShapeNet	PointNet 97.36%	PointNet++ 98.64%				

B.2. The annotation of semantic parts

This section discusses the annotation of semantic parts in the point-cloud dataset and image datasets. As mentioned in Section 3.3.1, given an input sample $\boldsymbol{x}$ with n input variables, the DNN may encode at most 2^n interaction concepts. The computational cost for extracting salient concepts is high, when the number of input variables n is large. For example, if we take each 3D point of a point-cloud (or each pixel of an image) as a single input variable, the computation is usually prohibitive. In order to overcome this issue, we simply annotate 8-10 semantic parts in each input sample, such that the annotated semantic parts are aligned over different samples[7]. Then, each semantic part in an input sample is taken as a "single" input variable to the DNN.

• For point-cloud data in the ShapeNet dataset, we annotated semantic parts for 100 samples in the *motorbike* category. These semantic parts were generated based on original annotations provided by (Yi et al., 2016). In the original annotation, Yi et al. (2016) provided semantic parts including *gas tank*, *seat*, *handle*, *light*, *wheel*, and *frame* for each *motorbike* sample. As shown in Fig. 10, we further modified the original annotation into more fine-grained semantic parts, *i.e. gas tank*, *seat*, *handle*, *light*, *front wheel*, *back wheel*, *front frame*, *mid frame*, and *back frame*.

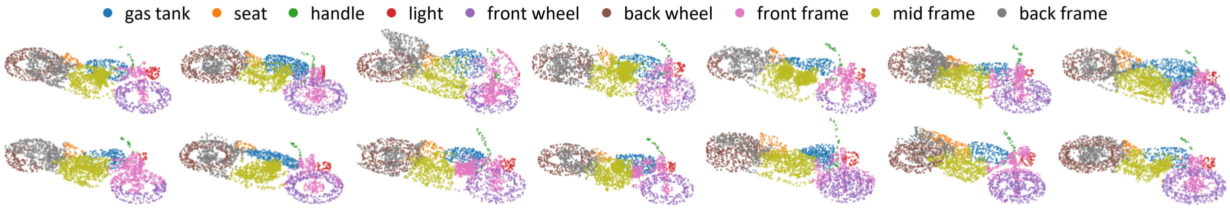


Figure 10. Examples of annotated semantic parts for samples in the *motorbike* category of the ShapeNet dataset.

• For image data, we annotated semantic parts for 50 samples in the *CelebA-eyeglasses* dataset. Specifically, as shown in Fig. 11, we annotated semantic parts including *forehead*, *left eye*, *right eye*, *nose*, *left cheek*, *right cheek*, *mouth*, *chin*, and *hair* for each sample in the *CelebA-eyeglasses* dataset[8]. Similarly, we annotated semantic parts for 20 samples in the *CUB-binary* dataset. These semantic parts include *head*, *neck*, *throat*, *wing*, *tail*, *leg*, *belly*, and *breast*. For images in the *MNIST-3* dataset, we annotated semantic parts for 100 positive samples. Please see the source code for details.

B.3. The setting of $v(\boldsymbol{x})$ in experiments

As mentioned in Section 3.1 of the paper, in the computation of the interaction effect $I(S|\boldsymbol{x})$, people can apply different settings for $v(\boldsymbol{x})$. For example, Covert et al. (2020) computed $v(\boldsymbol{x})$ as the cross-entropy loss of the sample $\boldsymbol{x}$ in the

[7] Actually, we can extract sparse and transferable interaction concepts without pre-annotated parts. Please see Appendix C.2 for experimental results.

[8] Note that we only considered interactions within foreground regions in each image, due to the high computational cost mentioned above. Therefore, the annotated semantic parts did not cover regions in the background. Please see Section B.4 for details on how to handle background regions in the computation of $I(S|\boldsymbol{x})$ for image data.

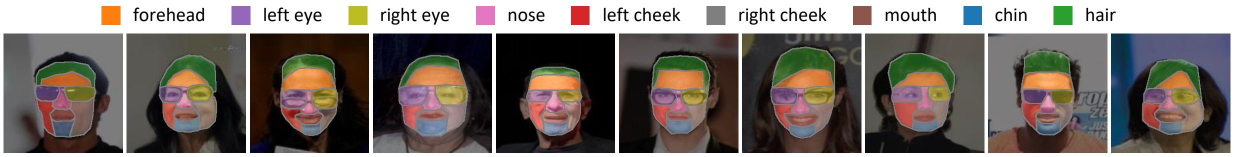


Figure 11. Examples of annotated semantic parts for positive samples in the *CelebA-eyeglasses* dataset.

classification task. Lundberg & Lee (2017b) directly set $v(\boldsymbol{x}) = p(y = y^{\text{truth}}|\boldsymbol{x}) \in \mathbb{R}$. In this paper, we followed (Deng et al., 2022a) and used $v(\boldsymbol{x}) = \log \frac{p(y=y^{\text{truth}}|\boldsymbol{x})}{1-p(y=y^{\text{truth}}|\boldsymbol{x})} \in \mathbb{R}$ for both binary classification tasks and multi-category classification tasks.

B.4. Computation details of $I(S|\boldsymbol{x})$ for image data

As mentioned in Section B.2, given an input sample $\boldsymbol{x}$ with n input variables, the DNN may encode at most 2^n interaction concepts. The computational cost for extracting salient concepts is high, when the number of input variables n is large. In order to overcome this issue, we only considered interaction concepts formed by foreground regions. For image data, the annotated semantic parts for each sample only covered regions in the foreground. In order to handle the uncovered regions in the background, in the extraction of interaction concepts, we averaged the interaction effect $I(S|\boldsymbol{x})$ when we were given multiple background with different strengths, which was similar to (Sundararajan et al., 2017).

Specifically, let the input image $\boldsymbol{x} \in \mathbb{R}^n$ be divided into the foreground region $\boldsymbol{x}^{\text{fg}} \in \mathbb{R}^{n^{\text{fg}}}$ and the background region $\boldsymbol{x}^{\text{bg}} \in \mathbb{R}^{n^{\text{bg}}}$, where $n = n^{\text{fg}} + n^{\text{bg}}$ and $\boldsymbol{x} = \boldsymbol{x}^{\text{fg}} \sqcup \boldsymbol{x}^{\text{bg}}$. The foreground region $\boldsymbol{x}^{\text{fg}}$ consisted of all pixels covered by the annotated semantic parts in Section B.2, and the background region $\boldsymbol{x}^{\text{bg}}$ consisted of all other uncovered pixels. Let $\boldsymbol{b}^{\text{bg}} \in \mathbb{R}^{n^{\text{bg}}}$ denote the baseline value for pixels in the background region $\boldsymbol{x}^{\text{bg}} \in \mathbb{R}^{n^{\text{bg}}}$. We defined the background region $\boldsymbol{x}^{\text{bg}}_\alpha$ with strength α *w.r.t.* the baseline value $\boldsymbol{b}^{\text{bg}}$ as $\boldsymbol{x}^{\text{bg}}_\alpha = \alpha \cdot \boldsymbol{x}^{\text{bg}} + (1-\alpha) \cdot \boldsymbol{b}^{\text{bg}}$, where the strength $\alpha \in [0, 1]$. When $\alpha = 0$, the background region was masked by the baseline value, *i.e.* $\boldsymbol{x}^{\text{bg}}_{\alpha=0} = \boldsymbol{b}^{\text{bg}}$. When $\alpha = 1$, the background region remained its original value, *i.e.* $\boldsymbol{x}^{\text{bg}}_{\alpha=1} = \boldsymbol{x}^{\text{bg}}$. When we computed the effect of each interaction concept S, we averaged the interaction effect when we were given multiple background regions with different strengths α, as follows.

$$I(S|\boldsymbol{x}) = \mathbb{E}_{\alpha\sim\mathcal{U}[0,1]} \left[I\left(S \mid \boldsymbol{x}^{\text{fg}} \sqcup \boldsymbol{x}^{\text{bg}}_\alpha \right) \right] = \int_0^1 I\left(S \mid \boldsymbol{x}^{\text{fg}} \sqcup \boldsymbol{x}^{\text{bg}}_\alpha \right) \, \mathrm{d}\alpha \tag{3}$$

B.5. The eight sub-categories in the *tic-tac-toe* dataset

In this section, we provide more details on the eight sub-categories for positive samples in the *tic-tac-toe* dataset, as mentioned in the footnote[5] of the paper.

Each sample $\boldsymbol{x}$ in the *tic-tac-toe* dataset (Dua & Graff, 2017) encodes one possible board configurations at the end of tic-tac-toe games. Specifically, each variable $\boldsymbol{x}_i$ indicates the state at the i-th position of the board, where $\boldsymbol{x}_i = 1$ indicates the player "X" has taken this position, $\boldsymbol{x}_i = -1$ indicates the player "O" has taken this position, and $\boldsymbol{x}_i = 0$ indicates this position is blank. If one of the player in the tic-tac-toe game creates a "three-in-a-row", then this player wins the game. In the *tic-tac-toe* dataset, positive samples includes all configurations where the player "X" wins the game. Since there are eight possible ways for the player "X" to create a "three-in-a-row", there are eight corresponding sub-categories for positive samples in the *tic-tac-toe* dataset. Specifically, these sub-categories contain patterns $\boldsymbol{x}_1 = \boldsymbol{x}_2 = \boldsymbol{x}_3 = 1$ (three-in-the-first-row), $\boldsymbol{x}_4 = \boldsymbol{x}_5 = \boldsymbol{x}_6 = 1$ (three-in-the-second-row), $\boldsymbol{x}_7 = \boldsymbol{x}_8 = \boldsymbol{x}_9 = 1$ (three-in-the-third-row), $\boldsymbol{x}_1 = \boldsymbol{x}_4 = \boldsymbol{x}_7 = 1$ (three-in-the-first-column), $\boldsymbol{x}_2 = \boldsymbol{x}_5 = \boldsymbol{x}_8 = 1$ (three-in-the-second-column), $\boldsymbol{x}_3 = \boldsymbol{x}_6 = \boldsymbol{x}_9 = 1$ (three-in-the-third-column), $\boldsymbol{x}_1 = \boldsymbol{x}_5 = \boldsymbol{x}_9 = 1$ (three-in-the-main-diagonal), $\boldsymbol{x}_3 = \boldsymbol{x}_5 = \boldsymbol{x}_7 = 1$ (three-in-the-anti-diagonal), respectively.

C. More experimental results

C.1. More verification on the existence of a concept dictionary

As a supplement to Fig. 4, Section 3.3.2 of the paper, we conducted another experiment to show the existence of a small concept dictionary $\mathbf{D}_k$ that could explain most concepts encoded by the DNN. Different from the experiment in Section 3.3.2, we extracted salient concepts $\Omega_{\boldsymbol{x}}$ by using the vanilla threshold $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$. Fig. 12 shows that there usually existed a concept dictionary consisting of 40-150 concepts, which could explain more than 60%-80% salient concepts encoded by the DNN. This still verified that the DNN learned transferable concepts over different samples.

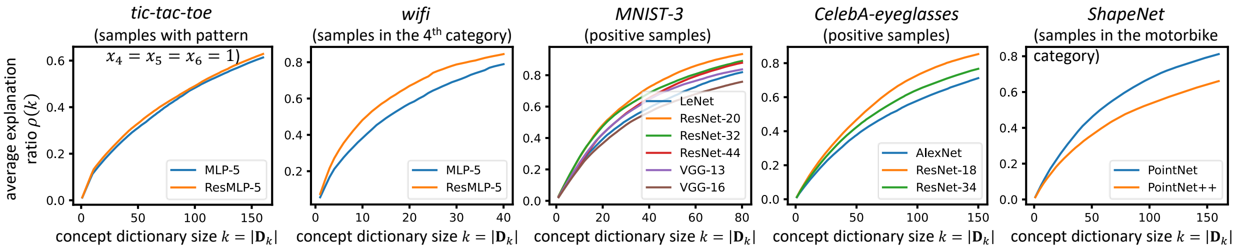


Figure 12. The change of the average explanation ratio $\rho(k)$ along with the size k of the concept dictionary $\mathbf{D}_k$, when we extracted salient concepts using the vanilla threshold $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$.

C.2. Extracting interaction concepts without the annotation of semantic parts

In this section, we conducted an experiment to extract concepts without pre-defined semantic parts. In this experiment, we first extracted super-pixels from the image, and consider each super-pixel as a basic input unit of the DNN. Then, we can extract interaction concepts between these super-pixels, based on $I(S|x)$ in Eq. (1). Specifically, we first segmented super-pixels from images in the CelebA dataset using the SLIC method (Achanta et al., 2012). Then, we extracted concepts encoded by ResNet-18 and ResNet-34 trained on the CelebA dataset for the classification of the eyeglasses attribute. Following the experimental settings in Fig. 3 of the paper, we visualized normalized strength of interaction effects of different concepts in a descending order. Experimental results in Fig. 13 show that the extracted concepts were still sparse. We also visualized some salient concepts extracted from ResNet-18 formed by super-pixels. We found that the salient concepts were usually meaningful to humans (super-pixels forming the "half face" concept, the "two eyes" concept, *etc.*), and they were also transferable across different samples.

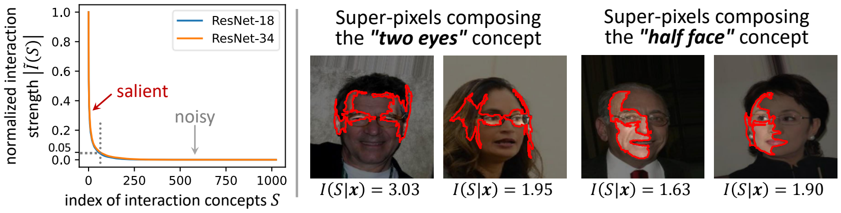


Figure 13. (left) Normalized strength of interaction effects of different concepts in a descending order. Concepts extracted from super-pixels are still sparse. (right) Visualization of salient concepts extracted from ResNet-18. The concepts are usually meaningful to humans, and they are also transferable across different samples.

C.3. Extracting interaction concepts encoded by NLP models

In this section, we extracted interaction concepts encoded by DNNs trained on NLP tasks.

In the first experiment, we trained a CNN network (Kim, 2014) on the SST-2 dataset for the sentiment classification task. Then, we extracted interaction concepts from this DNN. Table 2 shows the effects of salient concepts to the DNN's output for positive sentiment. We found that the extracted concepts were meaningful to human. For example, given the input sentence *"It's just not very smart,"* two salient interaction concepts {*not, smart*} and {*just, not*} contributed negative scores to the positive sentiment, while the interaction concepts {*just, very*} and {*just, very, smart*} contributed positive scores to the positive sentiment.

In the second experiment, we explained concepts encoded by a large language model (OPT-1.3B (Zhang et al., 2022b)) for the text generation task. Given the first k words in the sentence, we focused on the probability distribution of generating the $(k+1)$-th word. For example, given a partial sentence x =*"Diabetes is a chronic condition that affects how the body uses and stores,"* we focused on the output logits of the next word *"glucoses," i.e.*, $v(x) = \log \frac{p_{next}}{1-p_{next}}$, where $p_{next} = p(\text{glucoses}|\text{Diabetes ... stores})$. Thus, we extracted interaction concepts with salient effects on generating the target word. In Table 3, we showed that the model encoded meaningful concepts. For example, in Sentence 1, the model encoded concepts formed by relevant verbs ({*affects, and, stores*}), and concepts formed by both relevant nouns and verbs

Table 2. Effects of salient concepts extracted from a CNN network for the semantic classification task.

Sentence 1: It's just not very smart. Output: negative sentiment	Sentence 2: It is not too fast and not too slow. Output: positive sentiment
$I(\{\text{not, very}\} \mid x) = -2.56$	$I(\{\text{not, too}\} \mid x) = 6.49$ (the first "not too")
$I(\{\text{just, not}\} \mid x) = -1.52$	$I(\{\text{not, too}\} \mid x) = 4.87$ (the second "not too")
$I(\{\text{just, very}\} \mid x) = 0.95$	$I(\{\text{too}\} \mid x) = -3.33$
$I(\{\text{just, very, smart}\} \mid x) = 0.84$	$I(\{\text{not, slow}\} \mid x) = 1.59$
$I(\{\text{not, smart}\} \mid x) = -0.77$	$I(\{\text{and, not}\} \mid x) = -1.16$

({*body, stores*}). Both the interaction between {*affects, and, stores*} and the interaction between {*body, stores*} contributed to the correct generation of the output word *"glucoses."*

Table 3. Effects of salient concepts extracted from OPT-1.3B for the text generation task.

Sentence 1: Diabetes is a chronic condition that affects how the body uses and stores, Output: glucoses	Sentence 2: Physicist Isaac newton was born in 1642 in the village of Newton, Output: Abbot
$I(\{\text{Diabetes}\} \mid x) = 3.10$	$I(\{\text{village}\} \mid x) = 2.07$
$I(\{\text{body, stores}\} \mid x) = 3.08$	$I(\{\text{village,Newton}\} \mid x) = 1.05$
$I(\{\text{affects, and, stores}\} \mid x) = 2.62$	$I(\{\text{Issac,village,Newton}\} \mid x) = 0.90$
$I(\{\text{Diabetes,body,stores}\} \mid x) = -2.01$	$I(\{\text{was}\} \mid x) = -0.74$
$I(\{\text{how,body,stores}\} \mid x) = 1.95$	$I(\{\text{1642,in,village,Newton}\} \mid x) = -0.71$

C.4. Discussion on the relationship between interaction concepts and adversarial robustness

In this section, we analyze the relationship between different interaction concepts encoded by the DNN and the adversarial robustness of the DNN.

In the first experiment, we studied the robustness of different concepts. In this experiment, we found that high-order concepts (*i.e.* concepts which contain massive input variables) were less robust than low-order concepts (*i.e.* concepts which contain a small number of input variables). Therefore, we can examine different models based on the extracted concepts, and select models that encode less high-order non-robust concepts. In this way, the selected model would potentially be more robust.

Mathematically, we defined the *order* of a concept S as the number of input variables composing this concept, *i.e.* $\text{order}(S) = |S|$. Then, we evaluated the sensitivity of concepts S with different orders $|S|$, which was encoded by the VGG-16 model trained on the MNIST-3 dataset, when adversarial perturbations (Madry et al., 2018) were injected into the input sample. Given an input sample x, the adversarial perturbation δ was obtained via the L_∞ PGD attack (Madry et al., 2018), subject to $\|\delta\|_\infty < \frac{64}{255}$. The attack was iterated for 20 steps with the step size $\frac{4}{255}$. The sensitivity of concepts S with s-order (*i.e.* $|S| = s$) was defined as $\text{sensitivity}_s \triangleq \mathbb{E}_x\left[\frac{\sum_{S:|S|=s}|I(S|x+\delta)-I(S|x)|}{\sum_{S:|S|=s}|I(S|x)|}\right]$. Table 4 shows the sensitivity of concepts with different orders.

We found that high-order concepts usually exhibited higher sensitivity, thereby being less robust for inference. Notice that Eq. (2) in the paper shows the network output can be written as the sum of effects of all interaction concepts. Therefore, if a model encodes massive high-order concepts, the model would probably be less robust to adversarial attacks. This indicated that we could select models that encode less high-order non-robust concepts. In this way, the selected model would potentially be more robust.

Table 4. Sensitivity of concepts with different orders. High-order concepts are sensitive to adversarial noise, thereby being less robust.

	$s=1$	$s=2$	$s=3$	$s=4$	$s=5$	$s=6$
sensitivity$_s$	0.81	0.94	2.45	3.46	9.90	14.89

In the second experiment, we compared the transferability of concepts encoded by normally trained models with adversarially trained models (Madry et al., 2018). We found that besides improving the robustness of the model, adversarial training also improved the generalization power of features, *i.e.*, the transferability of the encoded concepts. Therefore, we can select models that encoded more transferable concepts, which would potentially be more reliable.

To this end, we trained another two VGG-13 networks and another two VGG-16 networks on the MNIST-3 dataset using adversarial training. Then, following the experimental settings in Fig. 6, Section 3.3.3, we checked the model-wise transferability of the concepts encoded by these DNNs. Specifically, let us suppose a pair of models v_1 and v_2 were trained for the same task. Given an input sample x, let $\Omega_x^{v_1}$ and $\Omega_x^{v_2}$ denote the sets of salient concepts extracted by v_1 and v_2 from sample x, respectively. We evaluated the ratio of concepts in $\Omega_x^{v_1}$ encoded by v_1 that were also encoded by v_2 in $\Omega_x^{v_2}$, *i.e.* $\gamma(\Omega_x^{v_1}|\Omega_x^{v_2}) \triangleq |\Omega_x^{v_1} \cap \Omega_x^{v_2}|/|\Omega_x^{v_1}|$, to measure the transferability of salient concepts in $\Omega_{\boldsymbol{x}}^{v_1}$. Following experimental settings in Section 3.3.3, given each sample x, $\Omega_{\boldsymbol{x}}^{v_2}$ contained all salient concepts with interaction strength $I_{v_2}(S|x) \geq 0.05 \cdot \max_S |I_{v_2}(S|x)|$. We used different thresholds τ ranging from $\tau = 0.05 \cdot \max_S |I_{v_1}(S|x)|$ to $\tau = 0.3 \cdot \max_S |I_{v_1}(S|x)|$ to generate different sets $\Omega_x^{v_1}$. A larger τ value usually generated a smaller set of salient concepts with more significant effects. We computed the average ratio over different samples $\mathbb{E}_x[\gamma(\Omega_x^{v_1}|\Omega_x^{v_2})]$ to measure the transferability of concepts between a pair of models.

Table 5 and Table 6 show that concepts encoded by adversarially trained models usually exhibit higher model-wise transferability. This may explain the robustness of adversarially trained models, to some extent. Therefore, besides improving the robustness of the model, adversarial training also improved the generalization power of features, *i.e.*, the transferability of the encoded concepts. This indicated that we could select models that encoded more transferable concepts, which would potentially be more reliable.

Table 5. Transferability of concepts between a pair of VGG-13 networks, when we extract salient concepts $\Omega_x^{v_1}$ under different thresholds.

λ, the threshold $\tau = \lambda \cdot \max_S |I_{v_1}(S|x)|$	0.05	0.10	0.15	0.20	0.25	0.30
a pair of normally trained VGG-13 networks	0.61	0.77	0.87	0.91	0.94	0.95
a pair of adversarially trained VGG-13 networks	**0.66**	**0.80**	0.87	**0.93**	**0.96**	**0.98**

Table 6. Transferability of concepts between a pair of VGG-16 networks, when we extract salient concepts $\Omega_x^{v_1}$ under different thresholds.

λ, the threshold $\tau = \lambda \cdot \max_S |I_{v_1}(S|x)|$	0.05	0.10	0.15	0.20	0.25	0.30
a pair of normally trained VGG-16 networks	0.56	0.71	0.82	0.88	**0.95**	0.96
a pair of adversarially trained VGG-16 networks	**0.62**	**0.76**	**0.85**	**0.91**	0.94	0.96