A Unified Interpretation of Training-Time Out-of-Distribution Detection

Xu Cheng, Xin Jiang, Zechao Li*
School of Computer Science and Engineering, Nanjing University of Science and Technology
{xcheng8, xinjiang, zechao.li}@njust.edu.cn

Abstract

This paper explains out-of-distribution (OOD) detection from a novel view, i.e., interactions between different input variables of deep neural networks (DNNs). Specifically, we provide a unified understanding of the effectiveness of current training-time OOD detection methods, i.e., DNNs trained with these methods all encode more complex interactions for inference than those trained solely with the cross-entropy loss, which contributes to their superior OOD detection performance. We further conduct empirical analyses and verify that complex interactions play a primary role in OOD detection, by developing a simple-yet-efficient method to force the DNN to learn interactions of specific complexities and evaluate the change of OOD detection performances. Besides, we also use interactions to investigate why near-OOD samples are more difficult to distinguish from in-distribution (ID) samples than far-OOD samples, mainly because compared to far-OOD samples, the distribution of interactions in near-OOD samples is more similar to that of ID samples. Moreover, we discover that training-time OOD detection methods can effectively decrease such similarities.

1. Introduction

Out-of-distribution detection is a task to identify whether or not a given test sample is drawn out of the training distribution, which is crucial for ensuring the robustness and reliability of DNNs in real-world applications. To this end, a rich line of literature has dedicated to enhancing OOD detection, and one typical direction is to adjust model training strategy to make OOD samples more distinguishable directly at training [10, 11, 16, 25, 33, 37], which is commonly referred to as training-time OOD detection methods [39, 42].

Existing training-time OOD detection methods are usually designed based on different observations and intuitions. For example, LogitNorm [37] employed logit normalization during training to produce distinguishable confidence scores between ID and OOD samples, while T2FNorm [25] performed feature normalization to improve OOD detection. Some methods [33, 35] added distributionally-shifted augmentations into training to enhance OOD detection. Despite their effectiveness in OOD detection, it is unclear whether a common mechanism underlies the effectiveness of these differently designed methods.

Thus, in this paper, we make the first attempt to explain the common mechanism behind the effectiveness of different training-time OOD detection methods, as well as the underlying reason why near-OOD samples are more difficult to distinguish from ID samples than far-OOD samples, from a novel perspective of interactions, which shed new light on understanding OOD detection.

To this end, we use interactions between different input variables of DNNs to explain OOD detection. It is because given a sample $\boldsymbol{x}$, the DNN usually does not employ each single input variable of $\boldsymbol{x}$ independently to identify whether it is in distribution or out of distribution. Instead, the DNN lets each input variable interact with each other to form a certain pattern for inference. For a better understanding, let us consider the toy example in Fig. 1(a). The DNN v encodes the interaction between a set of input variables $S = \{x_1, x_2\}$ of the input sample $\boldsymbol{x}$ to form the *dog ear* for inference. Each interaction represents an AND relationship between input variables in S. That is, only when all two variables in S are all present, the *dog ear* is activated and makes a numerical effect $I(S|\boldsymbol{x})$ to the network output $v(\boldsymbol{x})$. The absence/masking[1] of any variable in S will deactivate the interaction, and remove its corresponding effect $I(S|\boldsymbol{x})$.

Faithfulness of the interaction-based explanation. More crucially, Ren et al. [28] have derived two theorems to justify the faithfulness of taking such interactions as symbolic primitive inference patterns encoded by the DNN for inference. Specifically, it is proven that **a well-trained DNN usually encodes a few interactions, and its network output $v(\boldsymbol{x})$ can be well explained as the sum of numerical effects of these interactions,** $v(\boldsymbol{x}) = \sum_S I(S|\boldsymbol{x})$, as shown in Fig. 1(a). Besides, Li and Zhang [20] have empirically verified the *generalization property* of the interaction

*Corresponding author.

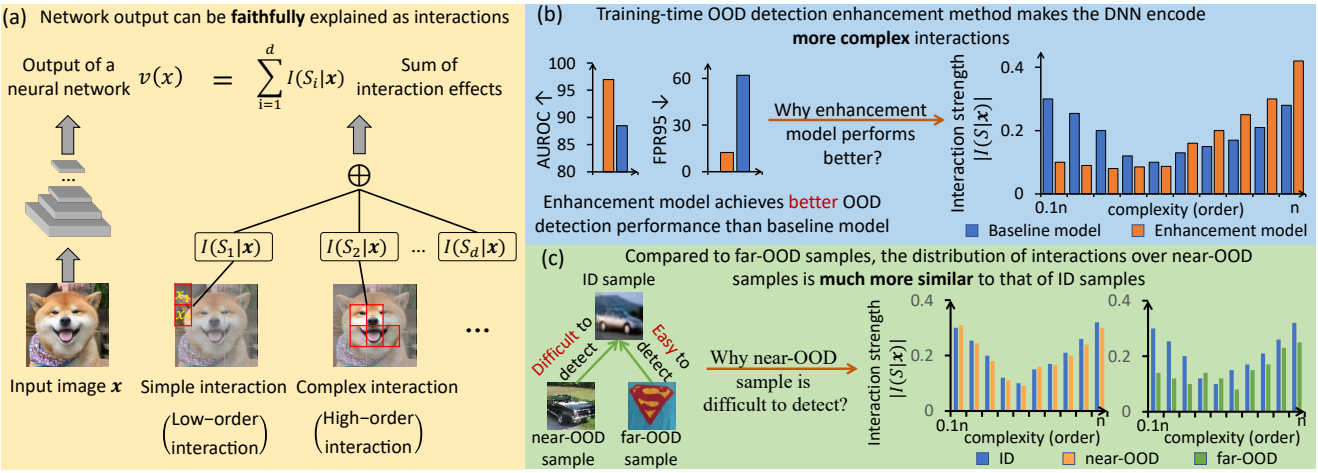

Figure 1. (a) Illustration of interactions between different input variables encoded by the DNN. The network output is proven to be faithfully explained as the sum of interaction effects. Thus, we use interactions to explain the effectiveness of current training-time OOD detection methods in (b), and explore why near-OOD samples are more difficult to detect than far-OOD samples in (c).

(c.f. Section 3.1). These findings serve as convincing evidences to ensure the faithfulness of interaction-based explanations.

Thus, owing to the guaranteed faithfulness of the interaction, we use it to progressively analyze the working principle behind OOD detection as follows.

• **Unified understanding of the effectiveness of current training-time OOD detection methods.** We discover a clear and shared difference between models trained with different training-time OOD detection methods to enhance OOD detection performance (abbreviated as ***enhancement models***) [25, 33, 35, 37] and models trained solely with the cross-entropy loss, without incorporating any training-time methods (abbreviated as ***baseline models***). As shown in Fig. 1(b), Fig. 2 and Fig. 3, **different training-time OOD detection methods all share a common mechanism that they all make the enhancement model encode more complex interactions than baseline models**, although they are originally designed based on different observations and intuitions. This provides a unified view to understand the effectiveness of these training-time methods in boosting OOD detection performance. Notably, we define the complexity of an interaction as the number of input variables in S. As shown in Fig. 1(a), a complex interaction usually represents the complex AND relationship between a large number of variables.

• **Verifying the primary role of complex interactions in OOD detection.** Intrigued by the above finding, we further conduct empirical analyses and verify that **complex interactions play a primary role in OOD detection**, by proposing a new loss function to force the DNN to learn interactions of specific complexities, and comparing the OOD detection performance of DNNs with and without complex interactions. Such an interesting conclusion, validated across diverse DNNs and diverse OOD datasets, may to some extent show a common factor for OOD detection.

• **Explaining why near-OOD samples is more difficult to distinguish from ID samples than far-OOD samples.** We also investigate this important-yet-underexplored problem from a new perspective of interactions. As Fig. 1(c) illustrates, **we discover that the similarity between the distribution of interactions in near-OOD samples and ID samples is higher than that between far-OOD and ID samples**, which increases the difficulty of detecting near-OOD samples. Besides, we also discovered that training-time OOD detection methods can effectively reduce such similarity.

Contributions of this paper are summarized follows. (1) We provide a unified understanding of the effectiveness of current training-time OOD detection methods based on interaction metrics. (2) We develop a simple-yet-efficient method to verify the primary role of complex interactions in OOD detection. (3) We explain the difficulty of detecting near-OOD samples from a new perspective of interactions.

2. Related Work

Explanation of OOD detection. Due to the importance of OOD detection in real-world scenarios, existing works were mainly dedicated to designing effective methods to boost OOD detection performance, by empirically exploring the differences in how DNNs encodes ID samples and OOD samples from different perspectives, such as logits or probabilities [25, 37], features or activations [19], parameter gradients [38], loss landscape [12], *etc*. However, only few previous works paid attention to explaining OOD detection. Kirichenko et al. [16] attributed the failure of nor-

malization flows in detecting OOD samples to their inability to learn latent representations for images based on semantic content. Du et al. [10, 11] considered the information in labels could help OOD detection.

However, there still lacked a clear and unified understanding of the effectiveness of different training-time OOD detection methods, as well as the difficulty of detecting near-OOD samples. Thus, we made the first attempt to address both issues from a novel view of interactions, so as to provide insightful explanations for this field.

Interaction-based explanation. Although DNNs have achieved remarkable performances on different tasks [22, 34], their underlying decision-making process still remains opaque and uninterpretable to humans. Post-hoc explanations of DNNs [21, 31, 32] is a typical direction in explainable AI, but its faithfulness may be disappointing [1, 30]. Fortunately, Li and Zhang [20], Ren et al. [27, 28] proposed interactions as a new perspective to analyze DNNs, and further empirically verified and theoretically proven a set of properties of interactions to ensure the faithfulness of the interaction-based explanation, which also served as the theoretical foundation of this paper. Thus, a series of studies employed the interaction to explicitly quantify concepts/knowledge encoded in the DNN [5, 20, 27, 28], to mathematically explain the representation capacity of the DNN [3, 23, 26, 29, 45], to unify the common underlying mechanisms behind various adversarial transferability methods [36, 44] and diverse attribution methods [9], and to explain the elementary mechanism of previous classical explanation metrics [27], such as the Shapley value [32], the Shapley interaction index [13], and *etc*.

In comparison, this paper aims to use interactions to analyze the shared mechanism behind different training-time OOD detection methods and the difficulty of detecting near-OOD samples to better understand OOD detection.

3. Understanding OOD Detection via Interactions

First, let us revisit OOD detection. Given a DNN v trained for the classification task, let $\mathcal{X}$ be the input space, and P_{in} be the marginal distribution of $\mathcal{X}$. Notably, to enhance OOD detection performance, people usually employ different training-time OOD detection methods when training the model v. Then, the goal of OOD detection is to use this trained model to classify whether a given test sample $\boldsymbol{x}$ is from in-distribution P_{in} or out-of-distribution P_{ood}, as follows.

$$g(\boldsymbol{x}) = \begin{cases} \text{In-distribution}, & S(\boldsymbol{x}) \geq \lambda, \\ \text{Out-of-distribution}, & S(\boldsymbol{x}) < \lambda. \end{cases} \tag{1}$$

$S(\boldsymbol{x})$ is a scoring function calculated based on the DNN v, and λ is a threshold. A high score $S(\boldsymbol{x})$ represents that the sample $\boldsymbol{x}$ is from ID, and vice versa.

3.1. Preliminary: interactions

In this section, we introduce the interaction metric, as well as a set of properties that mathematically support the faithfulness of using interactions to explain DNNs. These serves as the theoretical foundation for the explanation of training-time OOD detection enhancement methods in subsequent sections.

Definition of interactions. Given a well-trained DNN $v : \mathbb{R}^n \to \mathbb{R}$ for the classification task, and an input sample $\boldsymbol{x}$ with n variables indexed by $N = \{1, 2, ..., n\}$, let $v(\boldsymbol{x}) \in \mathbb{R}$ denote the scalar output of the DNN or a certain output dimension of the DNN. Note that people can apply various settings for $v(\boldsymbol{x})$. Here, we follow [8, 28] to set $v(\boldsymbol{x})$ as the confidence of classifying the input sample $\boldsymbol{x}$ to the ground-truth category y_{truth} in multi-category classification tasks, as follows.

$$v(\boldsymbol{x}) = \log \frac{p(y = y_{\text{truth}}|\boldsymbol{x})}{1 - p(y = y_{\text{truth}}|\boldsymbol{x})} \tag{2}$$

Then, Ren et al. [28] employed the Harsanyi Dividend [14], a fundamental metric in game theory, to quantify the numerical effect of the interaction between a set $S \subseteq N$ of input variables on the network output.

$$I(S|\boldsymbol{x}) = \sum\nolimits_{T \subseteq S} (-1)^{|S|-|T|} \cdot v(\boldsymbol{x}_T), \tag{3}$$

where $\boldsymbol{x}_T$ denotes the masked sample. $\boldsymbol{x}_T$ is generated by masking variables in $N \setminus T$ to baseline values[1], and keeping other variables in $T \subseteq N$ unchanged.

Understanding of interactions. Each interaction with the numerical effect $I(S|\boldsymbol{x})$ represents an AND (co-appearance) relationship between input variables in S. For a better understanding, let us consider a toy example in Fig. 1(a). The co-appearance of two variables in $S = \{x_1, x_2\}$ forms the semantic pattern of *dog ear*, and makes a numerical effect $I(S|\boldsymbol{x}) = v(\{x_1, x_2\}) - v(\{x_1\}) - v(\{x_2\}) + v(\boldsymbol{x}_\emptyset)$ on the network output, according to Eq. (3), where $\boldsymbol{x}_\emptyset$ represents the image with all patches masked. The masking of any variable in S will deactivate the *dog ear* interaction, and removes the numerical effect, *i.e.,* making $I(S|\boldsymbol{x}) = 0$.

Faithfulness of the interaction-based explanation. The theoretically proven ***sparsity property*** and ***universal-matching property***, along with the empirically verified ***generalization property*** of interactions, mathematically ensure that the inference logic of the DNN for a given input sample can be faithfully explained as interactions.

***Sparsity property*.** According to Eq. (3), a DNN can encode 2^n interactions *w.r.t.* all 2^n different subsets $\forall S \subseteq N$[2]

[1] In practice of masking input variables in $N \setminus T$, people usually use baseline values $\{b_i\}$ to replace the original values of these variables [2, 7], *i.e.,* setting $x_i = b_i$ if $i \in N \setminus T$. Here, we follow the widely-used setting in [7, 28] to set the baseline value of each variable b_i as the mean value of this variable over all samples in image classification.

[2] To reduce the computational cost, we follow [20, 27, 28] to select a relatively small number of input variables to calculate interactions in experiments. Please see the supplementary material for details.

at most. However, Ren et al. [28] have proven that under some common conditions[3], a well-trained DNN usually encodes very sparse interactions, *i.e.,* only a few interactions make salient effects $|I(S|\boldsymbol{x})|$ on the network output v. In contrast, all other interactions make negligible effects, $|I(S|\boldsymbol{x})| \approx 0$, which can be considered as noise patterns.

Theorem 1 (***universal-matching property***). *Given an input sample $\boldsymbol{x} \in \mathbb{R}^n$, there are totally 2^n different masked samples $\boldsymbol{x}_T$ w.r.t. all subsets $T \subseteq N$. Given a threshold τ[4], let $\Omega = \{S \subseteq N : |I(S|\boldsymbol{x})| > \tau\}$ denote the set of salient interactions, s.t. $|\Omega| \ll 2^n$. Then, Ren et al. [28] have proven that*

$$v(\boldsymbol{x}_T) = \sum\nolimits_{S\subseteq T} I(S|\boldsymbol{x}) \approx \sum\nolimits_{S\subseteq T \& S\in\Omega} I(S|\boldsymbol{x}). \quad (4)$$

Theorem 1 indicates that we can use a small number of salient interactions in Ω to universally match the network outputs $v(\boldsymbol{x}_T)$ on all 2^n masked samples $\{\boldsymbol{x}_T|T \subseteq N\}$.

Generalization property. Li and Zhang [20] have verified the generalization power of interactions. That is, interactions extracted from different samples in the same category or extracted from different models trained for the same task are often similar, and discriminative for classification.

Thus, the sparsity property, the universal-matching property, and the generalization property of interactions ensure that the interaction can be faithfully considered as the primitive inference pattern encoded by the DNN for inference, thereby guaranteeing the trustworthiness of interaction-based explanations for DNNs.

Complexity of interactions. The complexity of an interaction S is defined as the number of input variables involved in the interaction, which is also termed as the ***order*** of the interaction, *i.e.,* $\text{order}(S) = |S|$. Thus, as shown in Fig. 1(a), low-order interactions usually represent simple AND relationships between a small number of input variables, while high-order interactions often represent complex AND relationships between a large number of input variables.

3.2. Unified Understanding of Training-Time OOD Detection Methods

Training-time OOD detection methods usually adjust model training strategy to improve the distinguishability of OOD samples, such as incorporating feature or logit normalization during training [25, 37]. Despite their effectiveness in boosting OOD detection performance, a clear and unified explanation for their effectiveness still remains lacking. To this end, with theoretically guaranteed faithfulness of the interaction, we use it to dive deeply into the common mechanism behind the effectiveness of these different training-time OOD detection methods.

[3]Please see the supplementary material for the detailed introduction of common conditions.

[4]In experiments, we follow [20, 28] to set $\tau = 0.05 \cdot \max_S |I(S|\boldsymbol{x})|$.

We discover that although many training-time OOD detection methods [25, 33, 35, 37] are originally designed based on different intuitions and observations, **they all share a common mechanism that they all make the enhancement model encode more high-order interactions to improve the OOD detection performance than the baseline model.** Here, we refer to a DNN trained with a certain training-time OOD detection method as an enhancement model, and a DNN trained solely with the cross-entropy loss as a baseline model, for simplicity. Thus, this finding presents a unified view to understand the effectiveness of current training-time OOD detection methods.

Specifically, given an OOD sample $\boldsymbol{x}_{\text{OOD}}$, we compare the difference $\Delta R^{(m)}$ between the relative interaction strength of each complexity (order) encoded by the enhancement model v_{enhance}, and those encoded by the baseline model v_{baseline}, as follows.

$$\Delta R^{(m)} = R^{(m)}_{\text{enhance}} - R^{(m)}_{\text{baseline}},$$
$$R^{(m)}_{\text{enhance}} = \frac{\mathbb{E}_{\boldsymbol{x}_{\text{OOD}}}[\mathbb{E}_{S\subseteq N,|S|=m}[|I^{(m)}(S|\boldsymbol{x}_{\text{OOD}}, v_{\text{enhance}})|]]}{\sum_{m'}[\mathbb{E}_{\boldsymbol{x}_{\text{OOD}}}[\mathbb{E}_{S\subseteq N,|S|=m'}[|I^{(m')}(S|\boldsymbol{x}_{\text{OOD}}, v_{\text{enhance}})|]]]},$$
$$R^{(m)}_{\text{baseline}} = \frac{\mathbb{E}_{\boldsymbol{x}_{\text{OOD}}}[\mathbb{E}_{S\subseteq N,|S|=m}[|I^{(m)}(S|\boldsymbol{x}_{\text{OOD}}, v_{\text{baseline}})|]]}{\sum_{m'}[\mathbb{E}_{\boldsymbol{x}_{\text{OOD}}}[\mathbb{E}_{S\subseteq N,|S|=m'}[|I^{(m')}(S|\boldsymbol{x}_{\text{OOD}}, v_{\text{baseline}})|]]]}, \quad (5)$$

where the m-order interaction $I^{(m)}(S|\boldsymbol{x}_{\text{OOD}}, v_{\text{enhance}})$ of the enhancement model is calculated by revising Eq. (2) to $v_{\text{enhance}} = \log p(y_{\text{max}}|\boldsymbol{x}_{\text{OOD}})/(1 - p(y_{\text{max}}|\boldsymbol{x}_{\text{OOD}}))$, considering the fact that the OOD sample $\boldsymbol{x}_{\text{OOD}}$ does not have an ID label. y_{max} denotes the category predicted with the maximum probability, and the m-order interaction $I^{(m)}(S|\boldsymbol{x}_{\text{OOD}}, v_{\text{enhance}})$ is defined as the interaction $I(S|\boldsymbol{x}_{\text{OOD}}, v_{\text{enhance}})$ among the set $\forall S \subseteq N$ involving m input variables, $|S| = m$. The m-order interaction $I^{(m)}(S|\boldsymbol{x}_{\text{OOD}}, v_{\text{baseline}})$ encoded by the baseline model can be computed in a similar way.

Experiments. In order to explain the effectiveness of current training-time OOD detection methods, we followed [25, 37] to conduct experiments on ResNet-18, ResNet-34 [15], and WideResNet-40-2 [43] models. While not exhaustive, we trained five versions for each DNN, which contained a baseline model trained merely with the cross-entropy loss, and four enhancement models learned with different classical training-time OOD detection methods, including the CSI [33], LogitNorm [37], T2FNorm [25], and DAL [35] methods. Please see the supplementary material for training details. We followed [25] to use the CIFAR-10 and CIFAR-100 [17] datasets as ID datasets, and employed three widely-used benchmarks, namely SVHN [24], Textures [6], and Tiny-ImageNet [18], as OOD datasets. We set the CIFAR-10 dataset as the OOD dataset if the CIFAR-100 dataset was used as the ID dataset, and vice versa.

Fig. 2 and Fig. 3 show the difference $\Delta R^{(m)}$ between

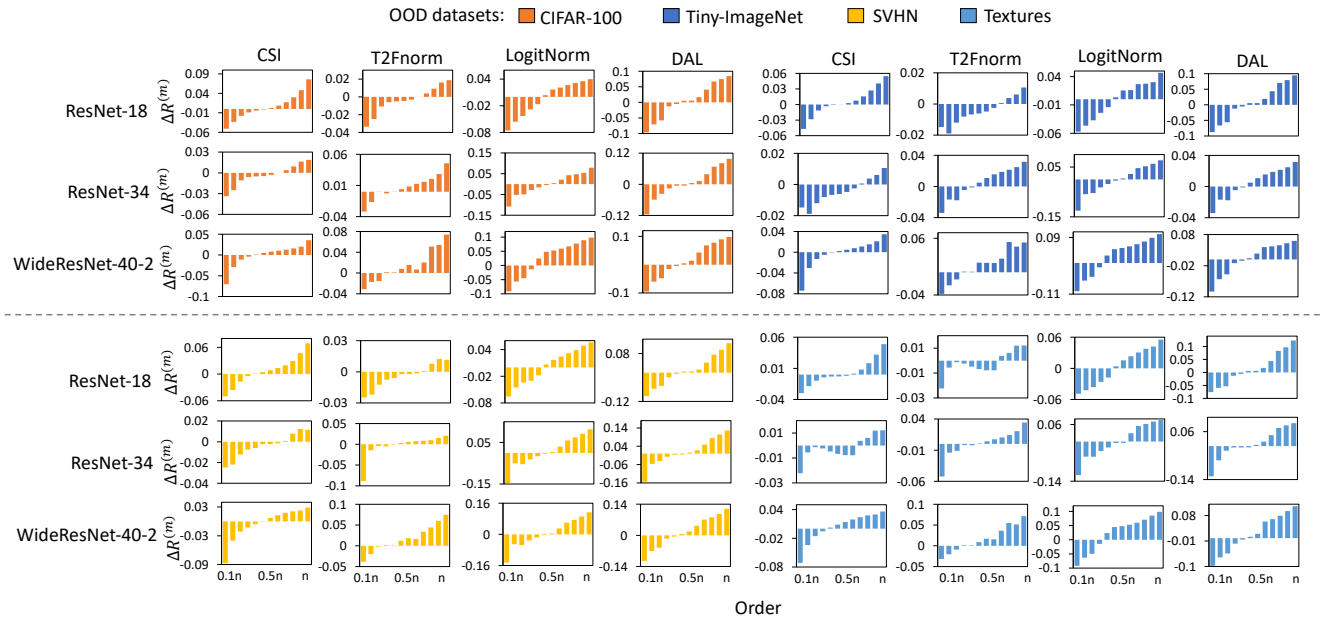

Figure 2. The difference $\Delta R^{(m)} = R^{(m)}_{\text{enhance}} - R^{(m)}_{\text{baseline}}$ between m-order interaction strength encoded by the enhancement model $R^{(m)}_{\text{enhance}}$ and that encoded by the baseline model $R^{(m)}_{\text{baseline}}$. Both the enhancement model and the baseline model are trained on CIFAR-10 dataset. We discover that different training-time OOD detection methods all make the model essentially encode more high-order interactions to boost OOD detection performance, although these methods are originally designed based on different observations and intuitions.

m-order interaction strength encoded by the enhancement model and that encoded by the baseline model. We observed $\Delta R^{(m)} > 0$ for $m > 0.75n$, and $\Delta R^{(m)} < 0$ for $m < 0.25n$, which indicated that the enhancement model encoded more high-order interactions and less low-order interactions than the baseline model to improve OOD detection performance. Moreover, such a phenomenon was shared by different training-time methods, although they were originally not designed to encode more high-order interactions. This provides a unified view to understand the effectiveness of training-time OOD detection methods.

The above shared phenomenon also implicitly reflected that high-order interactions were useful for OOD detection, which can be understood as follows. High-order interactions were proven to represent complex features [4, 45], which encoded sufficient discriminative information to distinguish OOD samples from ID samples. In comparison, low-order interactions were proven to represent small-scale, common and generalized features [4, 45] present in both the OOD samples and ID samples (*e.g.*, a small image patch of the blue sky in the background), thus they were not discriminative enough for OOD-ID differentiation.

3.3. Primary Role of High-Order Interactions in OOD Detection

Intrigued by the above finding, in this section, we further investigate the role of high-order interactions in OOD detection. To this end, we propose a simple-yet-efficient method to examine whether high-order interactions can significantly affect the OOD detection performance, by designing a new loss function to force the DNN to encode interactions of specific orders. We discover that **high-order interactions play a primary role in OOD detection**. This may provide new insights for the working principle of OOD detection.

Specifically, motivated by Theorem 1 that the network output $v(\boldsymbol{x}_T)$ can be well approximated by interaction effects of different orders, we first consider the difference of network outputs between different randomly masked samples, which is the foundation to design the loss.

$$\begin{aligned}\Delta v^{(m_1,m_2)} &= \mathbb{E}_{\substack{T_1,T_2\subseteq N \& T_1\subseteq T_2,\\ |T_1|=m_1 n, |T_2|=m_2 n}}[v(\boldsymbol{x}_{T_2}) - v(\boldsymbol{x}_{T_1})],\\ &= \mathbb{E}_{\substack{T_2\subseteq N,\\ |T_2|=m_2 n}}[v(\boldsymbol{x}_{T_2})] - \mathbb{E}_{\substack{T_1\subseteq N,\\ |T_1|=m_1 n}}[v(\boldsymbol{x}_{T_1})],\end{aligned} \tag{6}$$

where subsets T_1 and T_2 are randomly sampled from the universal set N, and $0 \le m_1 \le m_2 \le 1$. Then, we prove in Theorem 2 that the change of the network output $\Delta v^{(m_1,m_2)}$ mainly encodes interactions of the $[0, m_2 n]$-orders.

Theorem 2 (proven in the supplementary material)**.** *The change of the network output $\Delta v^{(m_1,m_2)}$ is proven to be represented as the sum of interaction effects of different or-*

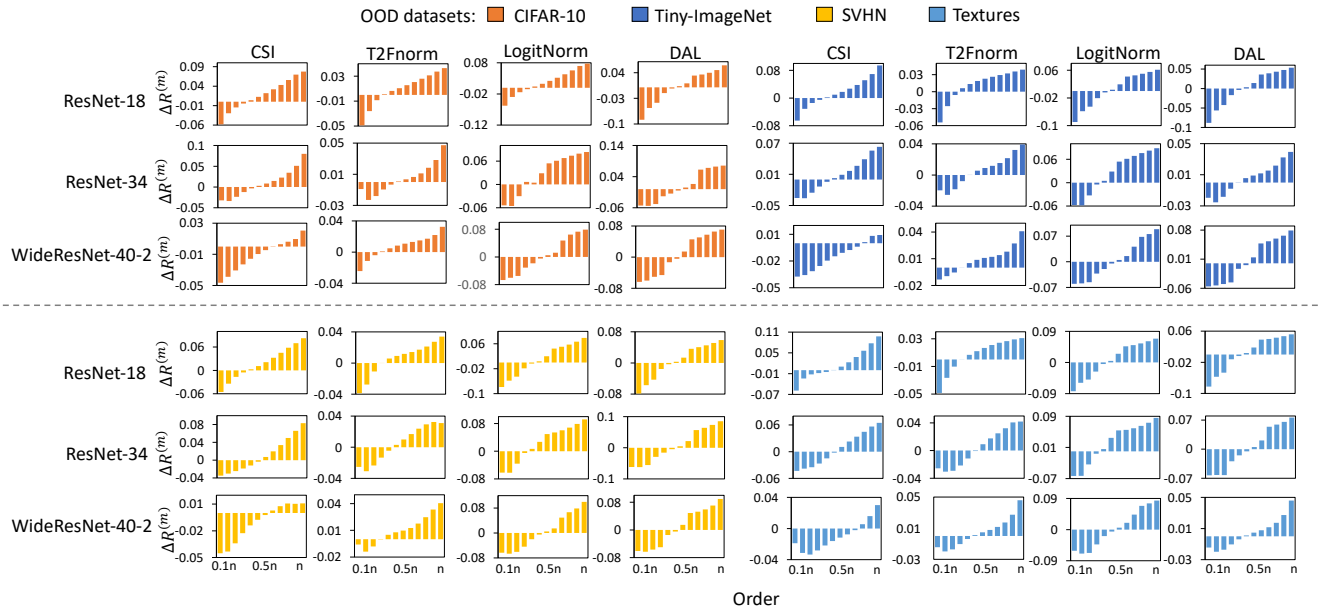

Figure 3. The difference $\Delta R^{(m)} = R^{(m)}_{\text{enhance}} - R^{(m)}_{\text{baseline}}$ between m-order interaction strength encoded by the enhancement model $R^{(m)}_{\text{enhance}}$ and that encoded by the baseline model $R^{(m)}_{\text{baseline}}$. Both the enhancement model and the baseline model are trained on CIFAR-100 dataset. We discover that different training-time OOD detection methods all make the model essentially encode more high-order interactions to boost OOD detection performance, although these methods are not originally designed for this purpose.

ders.

$$\Delta v^{(m_1,m_2)} = \sum\nolimits_{m=0}^{n} w^{(m)} \cdot \mathbb{E}_{S\subseteq N,|S|=m}[I(S|\boldsymbol{x})],$$
$$w^{(m)} = \begin{cases} C^m_{m_2 n} - C^m_{m_1 n}, & m \le m_1 n, \\ C^m_{m_2 n}, & m_1 n < m \le m_2 n, \\ 0, & m_2 n < m \le n. \end{cases} \tag{7}$$

In this way, based on Theorem 2, we can propose a new loss function $L^{(m_1,m_2)}_{\text{inter}}$ to prevent the DNN from encoding interaction of specific orders in the range of $[0, m_2 n]$, by maximizing the cross entropy calculated on $\Delta v^{(m_1,m_2)}$.

$$L^{(m_1,m_2)}_{\text{inter}} = -\mathbb{E}_{\boldsymbol{x}}[\sum\nolimits_{c=1}^{C}[p(\hat{y}=c|\Delta v_c^{(m_1,m_2)}(\boldsymbol{x}))\cdot \log(p(\hat{y}=c|\Delta v_c^{(m_1,m_2)}(\boldsymbol{x})))]], \tag{8}$$

where C and $\hat{y}$ denotes the total class number and the predicted label, respectively. The probability $p(\hat{y} = c|\Delta v_c^{(m_1,m_2)}(\boldsymbol{x}))$ of classifying the input sample $\boldsymbol{x}$ to the category c is computed by inputing the vector $[\Delta v_1^{(m_1,m_2)}, \Delta v_2^{(m_1,m_2)}, \cdots, \Delta v_C^{(m_1,m_2)}]$ into the softmax function, where $\Delta v_c^{(m_1,m_2)} = v_c(\boldsymbol{x}_{T_2}) - v_c(\boldsymbol{x}_{T_1})$ indicates the change of the logits of the category c.

Thus, we can minimize the following function L to train a DNN for classification with penalizing the learning of interactions of specific orders, where L_{ce} denotes the cross-entropy loss, and the small constant $\lambda \ge 0$ is used to balance two loss terms.

$$L = L_{\text{ce}} - \lambda \cdot L^{(m_1,m_2)}_{\text{inter}}. \tag{9}$$

Experiment 1: analyzing effects of the loss $L^{(m_1,m_2)}_{\text{inter}}$. Before investigating the role of high-order interactions in OOD detection, we first conduct experiments to examine whether the proposed loss function $L^{(m_1,m_2)}_{\text{inter}}$ could force the model to encode interactions of specific orders. To this end, we trained ResNet-18, ResNet-34, and WideResNet-40-2 models on CIFAR-10 and CIFAR-100 datasets, respectively. For each DNN, we trained three versions, including a baseline model trained merely with the cross-entropy loss by setting $\lambda = 0$, and two models trained to penalize interactions by setting $[m_1 = 0, m_2 = 0.3]$ (shown as the green line in Fig. 4) and $[m_1 = 0.7, m_2 = 1.0]$ (shown as the orange line in Fig. 4) in $L^{(m_1,m_2)}_{\text{inter}}$ with $\lambda = 0.1$, respectively.

Fig. 4 reports the relative interaction strength $R^{(m)}$ (defined in Eq. (5)). We discovered that the loss function $L^{(m_1,m_2)}_{\text{inter}}$ could successfully penalize interactions of $[m_1 n, m_2 n]$ orders, rather than $[0, m_2 n]$ orders. Explicitly speaking, when we trained the model to penalize interaction of $[0, 0.3n]$ orders ($[0.7n, n]$ orders), its interaction strength $R^{(m)}$ of $[0, 0.3n]$ orders ($[0.7n, n]$ orders) was significantly decreased, compared to the baseline model. Thus, for simplicity, we named these two types of models **high-order models** and **low-order models**, respectively, considering

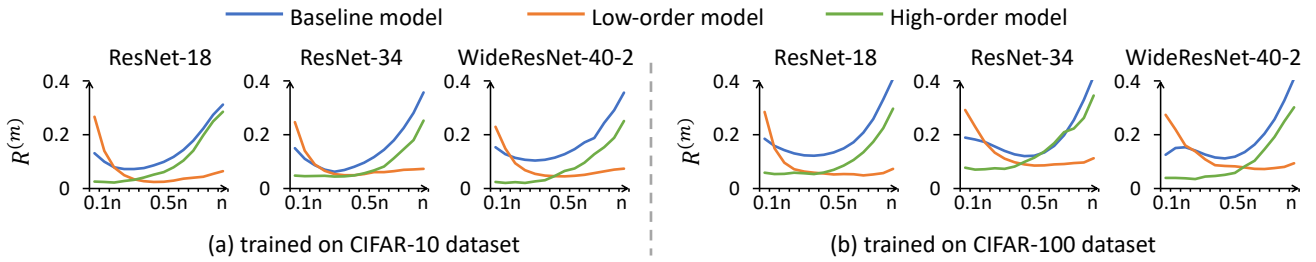

Figure 4. Distributions of the relative interaction strength $R^{(m)}$ of different DNNs. The baseline model is trained without penalizing specific interactions. The low-order (high-order) model is trained to penalize $[0.7n, n]$-orders ($[0, 0.3n]$-orders) interactions, thus mainly encoding low-order (high-order) interactions.

ID dataset	Model	ResNet-18		ResNet-34		WideResNet-40-2	
		FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
CIAFR-10	Baseline model	62.03	88.48	50.09	89.12	56.99	89.02
	Low-order model	$91.45_{+29.42}$	$73.07_{-15.41}$	$88.63_{+38.54}$	$69.29_{-19.83}$	$85.13_{+28.14}$	$70.16_{-18.86}$
	High-order model	$53.05_{-8.98}$	$89.53_{+0.85}$	$51.32_{+1.23}$	$88.97_{-0.15}$	$61.64_{+4.65}$	$86.63_{-2.39}$
CIAFR-100	Baseline model	79.70	78.15	78.95	78.30	78.01	76.90
	Low-order model	$92.69_{+12.99}$	$51.52_{-26.63}$	$89.98_{+11.03}$	$58.77_{-19.53}$	$90.33_{+12.32}$	$54.46_{-22.44}$
	High-order model	$75.82_{-3.88}$	$79.45_{+1.3}$	$81.51_{+2.56}$	$77.28_{-1.02}$	$82.51_{+4.5}$	$73.92_{-2.98}$

Table 1. OOD detection performance of different types of models averaged over four OOD datasets. The performance of low-order models decreases significantly compared to the baseline model, which illustrates that OOD detection mainly relies on high-order interactions. Notably, high-order models sometimes can even outperform the baseline model, which may inspire future research to further explore new methods for OOD detection.

they mainly encoded high-order and low-order interactions.

Experiment 2: high-order interactions played a primary role in OOD detection. To this end, we compared OOD detection performances between the baseline model and models trained to mainly encode interactions of specific order, *i.e.,* high-order models and low-order models. If OOD detection performance of low-order models (mainly encoding low-order interactions) dropped significantly compared to the baseline model, we could consider that high-order interactions played a primary role in OOD detection.

Specifically, based on the trained models in **experiment 1**, we employed two widely-used metrics to evaluate OOD detection performance averaged over four OOD datasets introduced in Section 3.2. Two metrics were the false positive rate of OOD samples at 95% true positive rate of ID samples (FPR95), and the area under the receiver operating characteristic curve (AUROC). A lower FPR95 and a higher AUROC value indicated better detection performance.

Table 1 shows that compared to the baseline model, low-order models exhibited a significant decrease in OOD detection performance, while high-order models only displayed a slight performance drop. Such a phenomenon illustrated that high-order interactions played a primary role in OOD detection, which also indicated that high-order interactions could, to some extent, serve as an effective factor for OOD detection. This might shed new light on understanding OOD detection.

3.4. Explaining the Difficulty of Detecting Near-OOD samples

In addition to explaining the shared mechanism of training-time OOD detection methods, we also investigate another important but underexplored problem: *why near-OOD samples are more challenging to detect than far-OOD samples*, from a new perspective of interactions. We discover that **compared to far-OOD samples, the interaction distribution of near-OOD samples is more similar to that of ID samples**, which increases the difficulty of distinguishing near-OOD samples from ID samples.

To this end, we compare the similarity of the interaction distribution between ID samples and OOD samples. Specifically, for each input sample, let us enumerate its all possible subsets $S \subseteq N$ and obtain d interactions in total. Let the vector $\boldsymbol{I}_{ID}(v) = [\boldsymbol{I}_{\text{ID}}(S_1|v),\ \boldsymbol{I}_{\text{ID}}(S_2|v),\ \ldots,\ \boldsymbol{I}_{\text{ID}}(S_d|v)]^\top \in \mathbb{R}^d$ represents the distribution of interactions over different ID samples, where the i-th dimension $\boldsymbol{I}_{\text{ID}}(S_i|v) = \mathbb{E}_{\boldsymbol{x}}[I(S_i|\boldsymbol{x}, v)]$ denotes the averaged interaction effect. Similarly, $\boldsymbol{I}_{\text{near-OOD}}(v)$ and $\boldsymbol{I}_{\text{far-OOD}}(v)$ denote the interaction distribution over various near-OOD samples and far-OOD

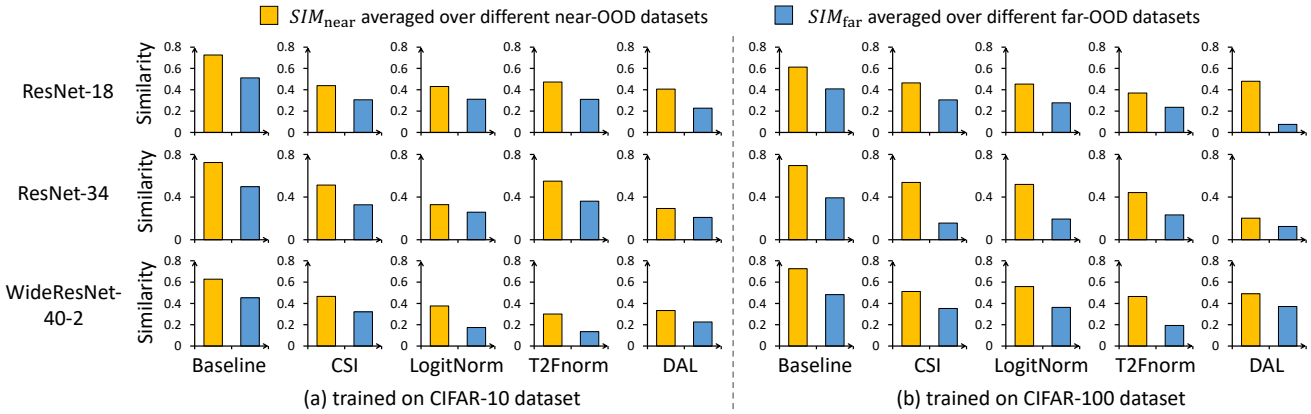

Figure 5. The similarity of interaction distribution between ID samples and near-OOD samples SIM_{near}, and that between ID samples and far-OOD samples SIM_{far}. SIM_{near} and SIM_{far} are averaged over two near-OOD datasets and two far-OOD datasets, respectively.

samples, respectively. Then, we employ the Jaccard similarity to compute the similarity of the interaction distribution between ID samples and near-OOD/far-OOD samples.

$$SIM_{\text{near}} = \frac{\|\min(\tilde{\boldsymbol{I}}_{\text{ID}}(v), \tilde{\boldsymbol{I}}_{\text{near-OOD}}(v))\|_1}{\|\max(\tilde{\boldsymbol{I}}_{\text{ID}}(v), \tilde{\boldsymbol{I}}_{\text{near-OOD}}(v))\|_1},$$
$$SIM_{\text{far}} = \frac{\|\min(\tilde{\boldsymbol{I}}_{\text{ID}}(v), \tilde{\boldsymbol{I}}_{\text{far-OOD}}(v))\|_1}{\|\max(\tilde{\boldsymbol{I}}_{\text{ID}}(v), \tilde{\boldsymbol{I}}_{\text{far-OOD}}(v))\|_1}, \tag{10}$$

where we expand the d-dimensional vector $\boldsymbol{I}_{\text{ID}}(v)$ into a $2d$-dimensional vector $\tilde{\boldsymbol{I}}_{\text{ID}}(v) = [(\max(\boldsymbol{I}_{\text{ID}}(v), 0))^\top, -(\min(\boldsymbol{I}_{\text{ID}}(v), 0))^\top] \in \mathbb{R}^{2d}$ to make it contain non-negative elements. Correspondingly, we extend vectors $\boldsymbol{I}_{\text{near-OOD}}(v)$ and $\boldsymbol{I}_{\text{near-OOD}}(v)$ to $\tilde{\boldsymbol{I}}_{\text{near-OOD}}(v)$ and $\tilde{\boldsymbol{I}}_{\text{far-OOD}}(v)$ without negative elements, respectively. A large SIM_{near} (SIM_{far}) value indicates that the interaction distribution of near-OOD samples (far-OOD samples) is similar to that of ID samples.

Experiments. We conducted experiments to examine whether the distribution of interactions in near-OOD samples was more similar to that in ID samples than far-OOD samples. To this end, we followed [25, 37] to train ResNet-18, ResNet-34, and WideResNet-40-2 models on the CIFAR-10 and CIFAR-100 datasets merely with the cross-entropy loss, respectively. Following [33, 40], we used the SVHN and Textures datasets as far-OOD datasets, and employed the Tiny-ImageNet dataset as near-OOD dataset, when CIFAR-10 and CIFAR-100 datasets were ID datasets. We also used the CIFAR-100 dataset as a near-OOD dataset for CIFAR-10, and vice versa.

Fig. 5 shows that the similarity SIM_{near} between near-OOD samples and ID samples was far more greater than the similarity SIM_{far} between far-OOD samples and ID samples. This illustrated that the interaction distribution of near-OOD samples was more similar to that of ID samples, which increased the difficulty of detecting ID samples. Notably, this conclusion partially echoed heuristic findings in [41] that far-OOD samples had more obvious domain shift to ID samples than near-OOD samples.

Fig 5 also compared the similarities SIM_{near} and SIM_{far} computed using the baseline model with the similarities $SIM_{\text{near, enhance}}$ and $SIM_{\text{far, enhance}}$ calculated using the enhancement model v_{enhance} in Eq. (10). Enhancement models were trained using CSI, LogitNorm, T2FNorm, and DAL methods, respectively. We discovered that both $SIM_{\text{near, enhance}}$ and $SIM_{\text{far, enhance}}$ were lower than SIM_{near} and SIM_{far}, which indicated that training-time methods could effectively decrease the similarity of interaction distributions between OOD samples and ID samples.

4. Conclusion and Discussions

This paper makes the first attempt to use theoretically verifiable interactions to provide a unified understanding of the effectiveness of different training-time OOD detection methods, all of which encode more high-order interactions to improve OOD detection performance. We also explain the primary role of high-order interactions in OOD detection by proposing a simple-yet-efficient method to force the DNN to learn interactions of specific orders. Besides, we investigate why near-OOD samples are more difficult to detect than far-OOD samples based on interactions.

Our work provides avenues for future exploration. We only focus on training-time OOD detection methods implemented on ResNet-based models. However, applying our interaction-based explanations to other types of OOD detection methods or DNNs with other network architectures for classification is theoretically feasible, as the interaction metric is architecture-agnostic [28]. Notably, for other methods, the setting of $v(\boldsymbol{x})$ in Eq. (2) to compute interactions may be changed. Nevertheless, we hope our explanation can serve as a theoretical foundation, inspiring future works to utilize it for OOD detection.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62406142, 62425603), Basic Research Program of Jiangsu Province of China (Grant No. BK20241467, BK20240011), Fundamental Research Funds for the Central Universities (No. 30925010409), Shanghai Key Laboratory of Intelligent Information Processing, Fudan University (Grant No. IIPL-2025-RD1-01).

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018. 3

[2] Marco Ancona, Cengiz Oztireli, and Markus Gross. Explaining deep neural networks with a polynomial time algorithm for shapley value approximation. In *International Conference on Machine Learning*, pages 272–281, 2019. 3

[3] Lu Chen, Siyu Lou, Benhao Huang, and Quanshi Zhang. Defining and extracting generalizable interaction primitives from DNNs. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[4] Xu Cheng, Chuntung Chu, Yi Zheng, Jie Ren, and Quanshi Zhang. A game-theoretic taxonomy of visual concepts in dnns. *arXiv preprint arXiv:2106.10938*, 2021. 5

[5] Xu Cheng, Lei Cheng, Zhaoran Peng, Yang Xu, Tian Han, and Quanshi Zhang. Layerwise change of knowledge in neural networks. In *ICML*, 2024. 3

[6] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014. 4

[7] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. *Advances in neural information processing systems*, 30, 2017. 3

[8] Huiqi Deng, Qihan Ren, Hao Zhang, and Quanshi Zhang. Discovering and explaining the representation bottleneck of DNNS. In *International Conference on Learning Representations*, 2022. 3

[9] Huiqi Deng, Na Zou, Mengnan Du, Weifu Chen, Guocan Feng, Ziwei Yang, Zheyang Li, and Quanshi Zhang. Unifying fourteen post-hoc attribution methods with taylor interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(7):4625–4640, 2024. 3

[10] Xuefeng Du, Zhen Fang, Ilias Diakonikolas, and Yixuan Li. How does unlabeled data provably help out-of-distribution detection? *arXiv preprint arXiv:2402.03502*, 2024. 1, 3

[11] Xuefeng Du, Yiyou Sun, and Yixuan Li. When and how does in-distribution label help out-of-distribution detection? *arXiv preprint arXiv:2405.18635*, 2024. 1, 3

[12] Kun Fang, Qinghua Tao, Xiaolin Huang, and Jie Yang. Revisiting deep ensemble for out-of-distribution detection: A loss landscape perspective. *International Journal of Computer Vision*, pages 1–20, 2024. 2

[13] Michel Grabisch and Marc Roubens. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of game theory*, 28:547–565, 1999. 3

[14] John C Harsanyi. A simplified bargaining model for the n-person cooperative game. *International Economic Review*, 4 (2):194–220, 1963. 3

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[16] Polina Kirichenko, Pavel Izmailov, and Andrew G Wilson. Why normalizing flows fail to detect out-of-distribution data. *Advances in neural information processing systems*, 33:20578–20589, 2020. 1, 2

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 4

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 4

[19] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018. 2

[20] Mingjie Li and Quanshi Zhang. Does a neural network really encode symbolic concepts? In *International conference on machine learning*, pages 20452–20469, 2023. 1, 3, 4

[21] Zechao Li, Jinhui Tang, and Tao Mei. Deep collaborative embedding for social image understanding. *IEEE transactions on pattern analysis and machine intelligence*, 41(9): 2070–2083, 2019. 3

[22] Zechao Li, Yanpeng Sun, Liyan Zhang, and Jinhui Tang. Ctnet: Context-based tandem network for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9904–9917, 2022. 3

[23] Dongrui Liu, Huiqi Deng, Xu Cheng, Qihan Ren, Kangrui Wang, and Quanshi Zhang. Towards the difficulty for a deep neural network to learn concepts of different complexities. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[24] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, page 4. Granada, 2011. 4

[25] Sudarshan Regmi, Bibek Panthi, Sakar Dotel, Prashnna K Gyawali, Danail Stoyanov, and Binod Bhattarai. T2fnorm: Extremely simple scaled train-time feature normalization for ood detection. *arXiv preprint arXiv:2305.17797*, 2023. 1, 2, 4, 8

[26] Jie Ren, Die Zhang, Yisen Wang, Lu Chen, Zhanpeng Zhou, Yiting Chen, Xu Cheng, Xin Wang, Meng Zhou, Jie Shi, et al. Towards a unified game-theoretic view of adversarial perturbations and robustness. *Advances in Neural Information Processing Systems*, 34:3797–3810, 2021. 3

[27] Jie Ren, Mingjie Li, Qirui Chen, Huiqi Deng, and Quanshi Zhang. Defining and quantifying the emergence of sparse

concepts in dnns. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20280–20289, 2023. 3

[28] Qihan Ren, Jiayang Gao, Wen Shen, and Quanshi Zhang. Where we have arrived in proving the emergence of sparse interaction primitives in dnns. In *The Twelfth International Conference on Learning Representations*. 1, 3, 4, 8

[29] Qihan Ren, Huiqi Deng, Yunuo Chen, Siyu Lou, and Quanshi Zhang. Bayesian neural networks avoid encoding complex and perturbation-sensitive concepts. In *International Conference on Machine Learning*, pages 28889–28913, 2023. 3

[30] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019. 3

[31] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 3

[32] Lloyd S Shapley. A value for n-person games. *Contribution to the Theory of Games*, 2, 1953. 3

[33] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. *Advances in neural information processing systems*, 33:11839–11852, 2020. 1, 2, 4, 8

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[35] Qizhou Wang, Zhen Fang, Yonggang Zhang, Feng Liu, Yixuan Li, and Bo Han. Learning to augment distributions for out-of-distribution detection. *Advances in neural information processing systems*, 36:73274–73286, 2023. 1, 2, 4

[36] Xin Wang, Jie Ren, Shuyun Lin, Xiangming Zhu, Yisen Wang, and Quanshi Zhang. A unified approach to interpreting and boosting adversarial transferability. In *ICLR*, 2021. 3

[37] Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating neural network overconfidence with logit normalization. In *International conference on machine learning*, pages 23631–23644, 2022. 1, 2, 4, 8

[38] Yingwen Wu, Tao Li, Xinwen Cheng, Jie Yang, and Xiaolin Huang. Low-dimensional gradient helps out-of-distribution detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2

[39] Kai Xu, Rongyu Chen, Gianni Franchi, and Angela Yao. Scaling for training time and post-hoc out-of-distribution detection enhancement. In *The Twelfth International Conference on Learning Representations*, 2024. 1

[40] Jingkang Yang, Pengyun Wang, Dejian Zou, Zitang Zhou, Kunyuan Ding, WENXUAN PENG, Haoqi Wang, Guangyao Chen, Bo Li, Yiyou Sun, Xuefeng Du, Kaiyang Zhou, Wayne Zhang, Dan Hendrycks, Yixuan Li, and Ziwei Liu. OpenOOD: Benchmarking generalized out-of-distribution detection. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 8

[41] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *International Journal of Computer Vision*, pages 1–28, 2024. 8

[42] Yue Yuan, Rundong He, Yicong Dong, Zhongyi Han, and Yilong Yin. Discriminability-driven channel selection for out-of-distribution detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26171–26180, 2024. 1

[43] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Procedings of the British Machine Vision Conference*, 2016. 4

[44] Quanshi Zhang, Xin Wang, Jie Ren, Xu Cheng, Shuyun Lin, Yisen Wang, and Xiangming Zhu. Proving common mechanisms shared by twelve methods of boosting adversarial transferability. *arXiv preprint arXiv:2207.11694*, 2022. 3

[45] Huilin Zhou, Hao Zhang, Huiqi Deng, Dongrui Liu, Wen Shen, Shih-Han Chan, and Quanshi Zhang. Explaining generalization power of a dnn using interactive concepts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 17105–17113, 2024. 3, 5