Machine Intelligence Research
www.mi-research.net
DOI: 10.1007/s11633-023-1419-7

Interpretability of Neural Networks Based on Game-theoretic Interactions

Huilin Zhou[1] Jie Ren[1] Huiqi Deng[1] Xu Cheng[1]
Jinpeng Zhang[2] Quanshi Zhang[1]

[1]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

[2]XLAB, The Second Academy of China Aerospace Science and Industry Corporation, Beijing 100854, China

Abstract: This paper introduces the system of game-theoretic interactions, which connects both the explanation of knowledge encoded in a deep neural networks (DNN) and the explanation of the representation power of a DNN. In this system, we define two game-theoretic interaction indexes, namely the multi-order interaction and the multivariate interaction. More crucially, we use these interaction indexes to explain feature representations encoded in a DNN from the following four aspects: 1) Quantifying knowledge concepts encoded by a DNN; 2) Exploring how a DNN encodes visual concepts, and extracting prototypical concepts encoded in the DNN; 3) Learning optimal baseline values for the Shapley value, and providing a unified perspective to compare fourteen different attribution methods; 4) Theoretically explaining the representation bottleneck of DNNs. Furthermore, we prove the relationship between the interaction encoded in a DNN and the representation power of a DNN (e.g., generalization power, adversarial transferability, and adversarial robustness). In this way, game-theoretic interactions successfully bridge the gap between "the explanation of knowledge concepts encoded in a DNN" and "the explanation of the representation capacity of a DNN" as a unified explanation.

Keywords: Model interpretability and transparency, explainable AI, game theory, interaction, deep learning.

Citation: H. Zhou, J. Ren, H. Deng, X. Cheng, J. Zhang, Q. Zhang. Interpretability of neural networks based on game-theoretic interactions. *Machine Intelligence Research*. http://doi.org/10.1007/s11633-023-1419-7

1 Introduction

In recent years, deep neural networks (DNNs) have shown significant success in various fields. However, the black-box nature of DNNs makes it difficult for people to understand their internal behavior. Essentially, the field of interpretability usually has two directions. The first is to explain semantic concepts corresponding to feature representations learned by DNNs. The second is to mathematically analyze the representation capacity of DNNs. Although there has been a lot of previous studies in both directions, they have been developed on different theoretical foundations, and there is no unified theory to connect the two directions.

Specifically, in the scope of explaining concepts encoded by DNNs, previous studies usually focus on three perspectives. 1) The visualization of network features is the most direct way of explaining the DNN. Dosovitskiy et al.[1–3] reconstructed the input image from intermediate-layer features to explain the information expressed by features. 2) Other studies usually quantify the attribution/importance/saliency of input variables to the output of a DNN, e.g., [4–6]. 3) In addition, the learning of a DNN with interpretable features is another typical way of boosting the network interpretability. Capsule networks[7] used capsules to encode interpretable representations that modeled the position, posture and other information of objects. InfoGAN[8] and β-VAE[9] trained generative networks with somewhat interpretable intermediate-layer features directly. However, these studies of explaining DNNs at the semantic level only visualized features modeled by DNNs, or quantified the importance of input variables to the output of a DNN, but these studies did not directly explain the representation capacity of DNNs, which was a more crucial problem in deep learning.

On the other hand, in the scope of mathematically analyzing the representation capacity of DNNs, most previous studies defined various metrics to evaluate the performance (e.g., adversarial robustness and generalization power) of DNNs. Previous studies usually used a single metric to analyze the entire complex system of a DNN. For example, Weng et al.[10] defined the CLEVER metric to evaluate the adversarial robustness of DNNs. Fort et al.[11] defined the stiffness metric, and Novak et al.[12] defined the sensitivity metric to evaluate the generaliza-

Review
Manuscript received on June 13, 2022; accepted on January 31, 2023
Recommended by Associate Editor Jun Zhu
Colored figures are available in the online version at https://link.springer.com/journal/11633

tion power of DNNs. However, the DNN can be considered to contain lots of potential causal factors that determine its generalization power (or adversarial robustness) due to the complex architecture and massive parameters of DNNs. In comparison, a single metric in previous studies was supposed not to be powerful enough to explain all potential factors responsible for generalization power (or adversarial robustness) of the DNN.

Therefore, in this paper, we revisit our several recent studies, which build up a new theoretical system to connect the direction of explaining concepts encoded in a DNN and the direction of mathematically analyzing the representation capacity of a DNN. Specifically, we define interactions in game theory as the theoretical foundation first. We investigate the multivariate interaction[13], define the multi-order interaction[14], and prove different properties of such interactions in game theory. Each specific interaction among multiple input units (e.g., the interaction between several words in a sentence, or the interaction between different regions in an image) can be considered as a specific concept encoded by the DNN. In this way, we can quantify concepts memorized by a DNN by using the interaction. We prove that we can explain the DNN as a mixture model of numerous concepts. For example, given a cat image as an input, a DNN may activate various concepts, such as eyes, nose, ears, mouth, etc. All these concepts make certain contributions to the inference of the cat (see Fig. 1). In this way, we can explain the representation capacity of DNNs from the perspective of concept representation. For example, we can explain the adversarial robustness of a DNN by disentangling robust concepts and non-robust concepts from a DNN. More specifically, we use game-theoretic interactions to explain the concept representation and the performance of DNNs (see Fig. 2) from three perspectives.

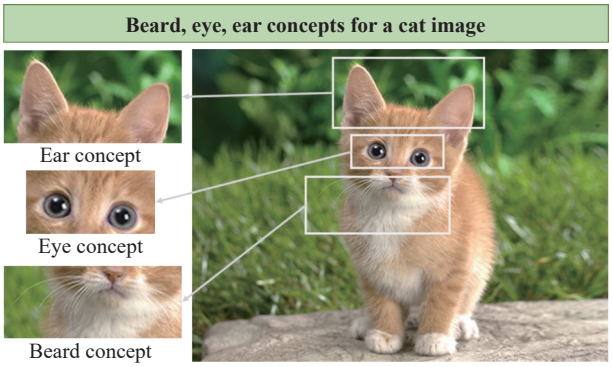


Fig. 1 DNN can be explained as a mixture model of massive concepts. For example, for a cat image, a DNN may encode beard concept, eye concept and ear concept at the same time, all these concepts make certain contributions to the inference of the cat.

Firstly, based on the multivariate interaction and the multi-order interaction, we further prove and fix theoretical flaws in attribution methods. For the computation of the Shapley value, we use game-theoretic interactions to define and learn optimal baseline values for the Shapley value[15]. Furthermore, we reformulate previous fourteen attribution methods from the perspective of the Taylor interaction. Based on the Taylor interaction, the attribution estimated by each method can be explained as effects caused by various interactions. The unified system enables people to fairly compare different attribution methods and discover theoretical flaws of different attribution methods[16].

Secondly, we prove that we can use game-theoretic interactions to explain feature representations of a DNN. 1) We find that we can use game-theoretic interactions to quantify knowledge points (concepts) encoded by the DNN in a hierarchical manner. For example, We represent causal factors in a DNN into a hierarchical interaction tree[17]. We use game-theoretic interactions to explain interactions among different input words encoded in natural language processing (NLP) models[18]. 2) We discover that we can use game-theoretic interactions to explain the representation power of a DNN for concept representations. For example, we prove that there exists a limitation (or bottleneck) when a DNN encodes feature representations[19], i.e., a DNN usually tends to encode both very simple interactions and very complex interactions, but it is difficult for a DNN to learn intermediate-complexity interactions. 3) We can also use game-theoretic interactions to explain signal-processing behaviors of deep neural networks for certain visual concepts. For example, we identify distinctive signal-processing behaviors of a DNN encoding different visual concepts[20]. Specifically, we clarify the difference between encoding textural concepts and encoding shape concepts. We use the multi-order interaction to explore prototypical concepts encoded by DNNs[21].

Thirdly, besides the explanation for concept representations encoded in a DNN, we can use game-theoretic interactions to explain the representation capacity of a DNN from the perspective of concept representation. Specifically, we find that the number and the reliability of concepts modeled by DNNs directly determines the DNN′s performance (e.g., adversarial robustness and generalization power). In contrast to traditional metrics, the diversity of concepts comprehensively and precisely explains diverse reasons for the performance of a DNN. In terms of adversarial robustness, we prove that high-order interactions are much more sensitive to adversarial perturbations than low-order interactions. We discover that adversarial training improves the robustness of high-order interactions, thereby boosting the robustness of the DNN. We propose a unified theoretic system to summarize the essential mechanism shared by four adversarial defense methods from the perspective of game-theoretic interactions[22]. Meanwhile, we prove the correlation between the adversarial transferability and interactions encoded by a DNN, and we explain five transferability-

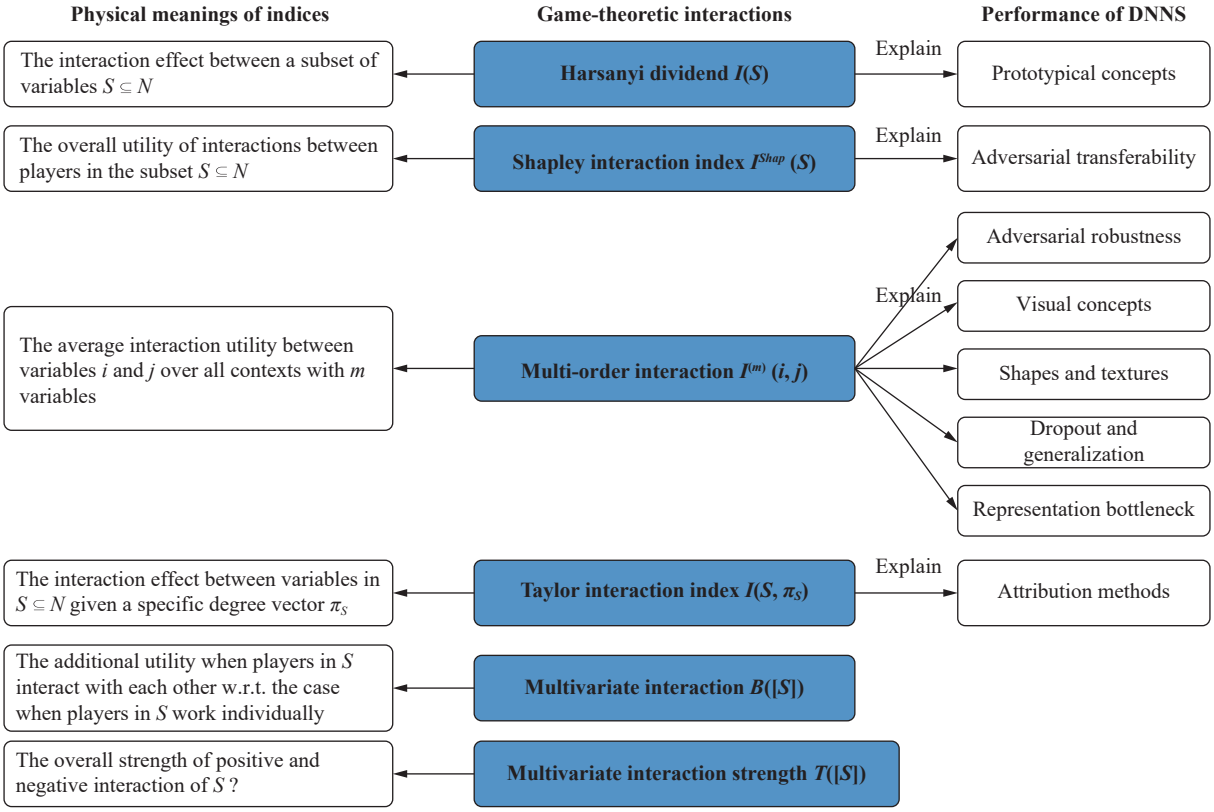


Fig. 2 Overview of the connection between the interaction and feature representations of the DNN

boosting attack methods as the decrease of interactions[23]. In terms of generalization power, we explore influences of the dropout operation on interactions encoded by a DNN, verify the relationship between a DNN's generalization power and its interactions, and propose a new training method to replace the dropout operation to precisely alleviate the over-fitting problem of DNNs[14].

In particular, we believe that game-theoretic interactions help us define and quantify knowledge points (or concepts) encoded by DNNs. Here we consider that analyzing the representation capacity from the perspective of concepts is a promising direction in explainable AI. In fact, game-theoretic interactions help us solve three key challenges in deep learning, i.e., verifying the trustworthiness of the explanation, clarifying specific reasons for network inference, extracting the common mechanism of different methods. 1) The unified theoretic system based on game-theoretic interactions verifies the trustworthiness of the explanation. Because there is usually no ground truth for the explanation of DNNs, it is difficult to prove that whether an explanation is trustworthy or not if without clarifying concepts encoded in a DNN. 2) Massive concepts extracted from a DNN can explain diverse and mixed reasons for the performance of a DNN. For example, a DNN is robust to adversarial perturbations because the DNN encodes a large number of robust concepts. With various internal interaction concepts in a DNN, we can precisely analyze the adversarial robustness and the generalization power of the DNN. 3) Based on game-theoretic interactions, we summarize the common mechanism shared by various methods into a single unified theoretic system. For example, in recent years, many methods of boosting adversarial transferability have been proposed based on different heuristics, but their essence may be actually the same. Based on such a unified theoretic system, we further detect potential flaws of existing methods, and revise these methods to further improve the performance of the DNN.

2 Definitions and properties

In fact, a DNN does not make the inference based on each individual input variable. Instead, a set of input variables interact with each other to form some inference patterns to make the inference together. For example, for a face image, a DNN may encode the eye pattern, the nose pattern and the mouth pattern, and these patterns collaborate with each other to form a larger pattern for inference. The game-theoretic interaction is a typical perspective to quantify interaction utilities between a set of input variables.

In 1952, the Shapley value[24] was proposed, and could be considered as a fair allocation of total rewards gained by all players to each individual player in the game. Later, many interaction metrics were proposed to quantify interaction utilities between players in a cooperative game. For example, Harsanyi[25] proposed the Harsanyi dividend in game theory, and Grabisch and Roubens[26]

proposed the Shapley interaction index.

We further propose several interaction metrics by extending previous classical interaction metrics. In order to quantify interaction utilities between two input variables under contexts of different complexities, we propose the multi-order interaction in [14]. Most previous studies mainly investigated interaction utilities between two input variables. In comparison, we quantify interaction utilities between multiple input variables. To this end, we propose the multivariate interaction in [13]. Because positive interactions and negative interactions contained in the multivariate interaction may neutralize each other, the multivariate interaction cannot precisely reflect the interaction strength. Therefore, we propose the multivariate interaction strength to measure the overall significance of both positive and negative interactions in [13].

More crucially, a strong interaction between multiple input variables can be considered as a concept consisting of these input variables, which is encoded in the DNN. Thus, such interactions provide a new perspective to analyze flaws of conceptual representation of a DNN, and the elimination of such representation flaws may improve the DNN′s performance. For ease of reading, we have listed the symbols used in the paper and the corresponding descriptions for each symbol in Table 1.

2.1 Preliminary: Shapley value

As the foundation of game-theoretic interactions, we first revisit the Shapley value. The Shapley value[24] was proposed to quantify the reward of each individual player in a cooperative game. Recently, the Shapley value has been applied to explain the attribution/contribution/importance of each input variable in a DNN. Specifically, we regard a deep neural network as a game with n players, $N = \{1, \cdots, n\}$. Then, each input variable (e.g., an input pixel or a word) corresponds to a player. The scalar output of a DNN is referred to as the total reward won by all players in the game. In this way, the attribution problem "how to fairly attribute the output of a DNN to each input variable" can be considered as "how to fairly distribute the total reward of a game to each player." The Shapley value $\phi(i)$ of the input variable i is defined, as follows:

$$\phi(i) = \sum_{S \subseteq N \setminus \{i\}} c(S)[v(S \cup \{i\}) - v(S)] \tag{1}$$

where $c(S) = |S|!(n - |S|+1)!/n!$. Here, $v(S \cup \{i\}) - v(S)$ is referred to as the additional award of the input variable i when it cooperates with input variables in the subset S. Correspondingly, the Shapley value $\phi(i)$ defines the average additional award of the input variable i over different sets of contextual variables S. In the computation of the Shapley value, $v(S)$ is the output of the DNN when input variables in S are present, and input variables in $N \setminus S$ are replaced by the baseline value. A widely-used setting of the baseline value is the average value of the variable over different samples[27].

Reference [28] has proven that the Shapley value satisfies the following four axioms. Therefore, the Shapley value can be considered as a fair approach to assigning the total reward gained by all players to each individual player.

Linearity axiom. If the game u and the game v are combined into a new game w, i.e., $\forall S \subseteq N$, $w(S) = u(S) + v(S)$, then for each player i in N, the Shapely value computed in the game u and that computed in the game v can be also combined into the Shapley value computed in the new game w, i.e., $\forall i \in N, \phi_w(i) = \phi_u(i) + \phi_v(i)$.

Dummy axiom. If a player $i \in N$ forms a coalition with any subset of players in $S \subseteq N \setminus \{i\}$, but the coalition cannot bring any additional reward, i.e., $\forall S \subseteq N \setminus \{i\}$, $v(S \cup \{i\}) = v(S) + v(\{i\})$, then the player i can be regarded as a dummy player. The Shapley value $\phi(i)$ of a dummy player satisfies $\phi(i) = v(\{i\}) - v(\emptyset)$.

Symmetry axiom. Let us consider two players i, j, if for any subset of players $S \subseteq N \setminus \{i, j\}$, the player i collaborates with players in S in the same way as how the player j collaborates with players in S, i.e., $\forall S \subseteq N \setminus \{i, j\}$, $v(S \cup \{i\}) = v(S \cup \{j\})$, then the player i and the player j have the same Shapley value, i.e., $\forall S \subseteq N \setminus \{i, j\}$, $\phi(i) = \phi(j)$.

Efficiency axiom. The reward assigned to each player adds up to exactly the total reward. i.e., $v(N) - v(\emptyset) = \sum_{i \in N} \phi(i)$.

2.2 Game-theoretic interactions

Interactions between input variables have been widely investigated. Several classical interaction metrics were proposed in game theory, such as the Harsanyi dividend interaction[25] and the Shapley interaction index[26]. We propose two extended interaction metrics, e.g., the multi-order interaction metric[14], and the multivariate interaction metric[13]. Then, we propose a method to fast approximate the overall significance of all types of multivariate interactions among a set of input variables[13].

Besides, in this section, we will also discuss game-theoretic axioms of different interaction metrics, which ensures the trustworthiness of these interaction metrics.

2.2.1 Harsanyi dividend

Let us consider a cooperative game with n players, $N = \{1, \cdots, n\}$. We can consider the output of a neural network as the total reward in a game, and consider input variables as players. Harsanyi dividend[25] $I(S)$ is proposed to quantify the interaction between a set of players $S \subseteq N$, which influences the network output $v(L)$ as compositional causal factors.

$$I(S) = \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L) \tag{2}$$

where $v(L)$ is referred to as the output of the DNN when

we mask input variables in $N \setminus L$ by replacing their values with their baseline values. The Harsanyi dividend naturally guarantees that interaction utilities of all subsets of N can fit the exact model output $v(N)$, as follows:

$$v(N) = \sum_{S \subseteq N} I(S). \tag{3}$$

As shown above, Harsanyi[25] has ensured that the Harsanyi dividend satisfies the efficiency property. To ensure that the Harsanyi dividend is a trustworthy approach to quantifying interaction utilities between a subset of input variables, we prove the following properties in [17].

Efficiency property (proven by [25]). Interaction utilities of all subsets of N can fit the exact model output $v(N)$, i.e., $v(N) = \sum_{S \subseteq N} I(S)$.

Linearity property. If game u and game v can be combined into a new game w, i.e., for $\forall S \subseteq N$, $w(S) = u(S) + v(S)$, then the Harsanyi dividend of game u and game v can also be combined into the Harsanyi dividend of the new game w, i.e., $I_w(S) = I_u(S) + I_v(S)$.

Dummy property. If the co-appearance of the player $i \in N$ and any subset of players $S \subseteq N \setminus \{i\}$ does not have any interaction utilities, i.e., $\forall S \subseteq N \setminus \{i\}$, $v(S \cup \{i\}) = v(S) + v(\{i\})$, then the player i is considered as a dummy player. A dummy input variable participates the game without interacting with other input variables, i.e., $\forall S \subseteq N \setminus \{i\},\ I(S \cup \{i\}) = 0$.

Symmetry property. Consider two players i, j, if for any subset of players $S \subseteq N \setminus \{i, j\}$, the way how the player i collaborates with players in S is the same as the way how the player j collaborates with players in S, i.e., if $\forall S \subseteq N \setminus \{i, j\},\ v(S \cup \{i\}) = v(S \cup \{j\})$, then, $\forall S \subseteq N \setminus \{i, j\},\ I(S \cup \{i\}) = I(S \cup \{j\})$.

Anonymity property. If a random permutation π is added to N, then $\forall S \subseteq N$, $I_v(S) = I_{\pi v}(\pi S)$ is always guaranteed, where the new set of players πS is defined as $\pi S = \{\pi(i), i \in S\}$, the new game πv is defined as $(\pi v)(\pi S) = v(S)$. This suggests that permutation does not change the Harsanyi dividend.

Recursive property. The Harsanyi dividend can be calculated in a recursive manner. For $\forall i \in N, S \subseteq N \setminus \{i\}$, the Harsanyi dividend of $S \cup \{i\}$ can be calculated as the difference between the Harsanyi dividend of S when setting the presence of the player i as a constant background and the vanilla Harsanyi dividend of S that considers the absence of the player i as the constant background, i.e., $\forall i \in N,\ S \subseteq N \setminus \{i\},\ I(S \cup \{i\}) = I(S|i$ is consistently present$) - I(S)$, where $I(S|i$ is consistently present$) = \sum_{L \subseteq S} (-1)^{|S|-|L|} v(L \cup \{i\})$.

Interaction distribution property. This property describes how an interaction function[29] distributes interactions. An interaction function v_T parameterized by a context T is defined as follows, i.e., for $\forall S \subseteq N$, if context T is a subset of S, i.e., $T \subseteq S$, then $v_T(S) = c$; if not, $v_T(S) = 0$. Then, the interaction for an interaction function v_T can be computed as $I(T) = c$, and $\forall S \neq T,\ I(S) = 0$.

Connections with the Shapley value. Harsanyi[25] has proven that the Harsanyi dividend connects strongly with the Shapley value. Specifically, the Shapley value of the input variable i can be computed by adding up Harsanyi dividend interactions when the input variable i cooperates with different subsets of input variables. In particular, when the input variable i collaborates with each subset of input variables $S \subseteq N \setminus \{i\}$, such a collaboration makes an interaction utility, which is quantified by the Harsanyi dividend interaction $I(S \cup \{i\})$. The Shapley value considers that each input variable participating in the collaboration contributes to the Harsanyi dividend interaction equally. Therefore, the input variable i will be allocated $1/(|S|+1)$ proportion of the specific Harsanyi dividend interaction $I(S \cup \{i\})$ as a numerical component of its Shapley value. i.e.,

$$\phi(i) = \sum_{S \subseteq N \setminus \{i\}} \frac{1}{|S|+1} \cdot I(S \cup \{i\}). \tag{4}$$

2.2.2 Shapley interaction index

Grabisch and Roubens[26] have proposed the Shapely interaction index $I^{Shap}(S)$. Let us consider a cooperative game v where n players participate in the game, $N = \{1, \cdots, n\}$. In order to use the Shapley interaction index to explain the DNN, we can consider the DNN as the game, consider the model output as the reward in the game, and regard input variables as players. The Shapley interaction index is defined to measure the overall utility of interactions between players in the subset $S \subseteq N$, as follows:

$$I^{Shap}(S) = \sum_{T \subseteq N \setminus S} \frac{t!(n-t-s)!}{(n-s+1)!} \Delta_S v(T) \tag{5}$$

where $\Delta_S v(T) = \sum_{L \subseteq S} (-1)^{s-l} v(L \cup T)$ quantifies interaction utilities between variables in the subset S given the contextual variable subset $T \subseteq N \setminus S$. The Shapley interaction index $I^{Shap}(S)$ measures the average interaction utility between input variables in S under different contextual variable subsets.

Based on the definition, Grabisch and Roubens[26] have proven that the Shapely interaction index satisfies the following properties.

Linearity property. If we merge the game u and the game v into a new game w, i.e., for $\forall S \subseteq N$, $w(S) = u(S) + v(S)$, then, for any subset of players $S \subseteq N$, the interaction utilities between players in S in game u and those in game v can also be merged into interaction utilities between players in S in the new game w, i.e., $I_w^{Shap}(S) = I_u^{Shap}(S) + I_v^{Shap}(S)$.

Dummy property. Let us assume that a player $i \in N$ cooperates with a subset of players $S \subseteq N \setminus \{i\}$,

Table 1 Notation used in this paper

Symbols	Descriptions
$\boldsymbol{x}$	The input sample ($\boldsymbol{x} = [x_1, x_2, \cdots, x_n]^{\mathrm{T}}$)
$\boldsymbol{\delta}$	The adversarial perturbation ($\boldsymbol{\delta} = [\delta_1, \delta_2, \cdots, \delta_n]^{\mathrm{T}}$)
$\tilde{\boldsymbol{x}}$	The adversarial sample
y	The ground truth label of the input sample
Ω	The set of all input samples ($\boldsymbol{x} \in \Omega$)
N	The input variable set ($N = \{1, 2, 3, \cdots, n\}$)
S, T	The input variable subset ($S \subseteq N,\ T \subseteq N$)
i	An single input variable i
x_i	The value of the input variable i
b_i	The baseline value of the input variable i
a_i	The attribution score of the input variable i
δ_i	The adversarial perturbation on the input variable i
u, v, w, h	A pre-trained deep neural network model
W	The network parameter of a DNN
$\triangle W$	The change of the network parameter W
π_S	The degree vector of the Taylor expansion ($\pi_S = [\kappa_1, \kappa_2, \cdots \kappa_n]^{\mathrm{T}}$)
$\phi(i)$	The Shapley value of the input variable i
$I(S)$	The Harsanyi dividend of a subset of input variables S
$I^{Shap}(S)$	The Shapley interaction index of a subset of input variables S
$I^{(m)}(i, j)$	The multi-order interaction between the input variables i and j
$B([S])$	The multivariate interaction between input variables in S
$T([S])$	The overall multivariate interaction strength between input variables in S
$I(S, \pi_S)$	The Taylor interaction between input variables in S based on π_S
$B_{\max}([S])$	Overall strength of positive interactions
$B_{\min}([S])$	Overall strength of negative interactions
B^+	The sum of absolute value of all positive elementary interaction components
B^-	The sum of absolute value of all negative elementary interaction components
$\ell_{\text{interaction}}$	The loss to penalize the strength of game-theoretic interactions
ℓ_{IR}	The interaction-reduced loss
$D^{(m)}$	The disentanglement metric to describe the discriminative power of m-order interactions
$J^{(m)}$	The relative strength of m-order interactions
$\triangle u_c(r_1, r_2)$	Interactions of $[0, r_2 n]$-th orders for the inference of category c
$\triangle W^{(m)}(i, j)$	The component of $\triangle W$ w.r.t the gradient of the interaction $I^{(m)}(i, j)$ to the network parameter W
$L^+(r_1, r_2)$	The loss function to encourage the DNN to use interactions of the orders within the range $[0, r_2 n]$
$L^-(r_1, r_2)$	The loss function to penalize the DNN to use interactions of the orders within the range $[0, r_2 n]$

and the co-appearance of the player i and any subset of players S does not have any interaction effects, i.e., $\forall S \subseteq N \setminus \{i\},\ v(S \cup \{i\}) = v(S) + v(\{i\})$. Then, the player i can be regarded as a dummy player. A dummy play-

er joins the game without interacting with other players in the game, i.e., $\forall S \subseteq N \setminus \{i\},\ I^{Shap}(S \cup \{i\}) = 0$.

Symmetry property. Let us consider two players i and j, if for any subset of players $S \subseteq N$, the player i collaborates with players in S in the same way as how the player j collaborates with players in S, i.e., $i, j \in N$, $\forall S \subseteq N \setminus \{i, j\},\ v(S \cup \{i\}) = v(S \cup \{j\})$, then $\forall S \subseteq N \setminus \{i, j\},\ I^{Shap}(S \cup \{i\}) = I^{Shap}(S \cup \{j\})$.

Recursive property 1. The Shapley interaction index $I^{Shap}(S)$ is equal to the difference between rewards gained by S and the sum of rewards gained by each subset L of S. Specifically, for $\forall S \subseteq N, |S| > 1$, we have

$$I^{Shap}(S|N) = I^{Shap}([S]|N_{[S]}) - \sum_{L \subsetneq S, L \neq \emptyset} I^{Shap}(L|N_L). \tag{6}$$

Here, $I^{Shap}(S|N)$ denotes the Shapley interaction index of S over the set N. Besides, in the computation of $I^{Shap}([S]|N_{[S]})$, we consider the subset S as a singleton player $[S]$, rather than a set of multiple players. In this way, $I^{Shap}([S]|N_{[S]})$ represents the Shapley interaction index of the singleton player $[S]$ over the set $N_S \overset{\text{def}}{=} N \backslash S \cup \{[S]\}$. Similarly, $I^{Shap}(L|N_L)$ denotes the Shapley interaction index of multiple players in L over the set $N_L \overset{\text{def}}{=} N \backslash S \cup L$, where players in $S \backslash L$ do not participate in the game.

Recursive property 2. The interaction utility between players in $S \cup \{i\}$, i.e., $I^{Shap}(S \cup \{i\})$, can be computed as "the interaction utility between players in S when player i always participates the game" minus "the interaction utility between palyers in S when player i is always absent," i.e., $\forall i \in S,\ I^{Shap}(S \cup \{i\}) = I^{Shap}(S|\text{with the presence of } i) - I^{Shap}(S|\text{with the absence of } i)$.

Connections with the Harsanyi dividend

We have proven that the Shapley interaction index connects with the Harsanyi dividend[17]. Specifically, in the computation of the Shapley interaction index $I^{Shap}(S)$, we regarded the set S as a singleton player $[S]$. When the singleton player $[S]$ cooperates with each different subset of players $T \subseteq N \backslash S$, such a collaboration creates an interaction utility, which is given as the Harsanyi dividend interaction $I(T \cup [S])$. It is considered that each player participating in the collaboration contributes equally to the specific Harsanyi dividend interaction. Therefore, the singleton player $[S]$, as one of the players, gets $1/(|T|+1)$ proportion of the specific Harsanyi dividend interaction as a component of the interaction utility $I^{Shap}(S)$. In this way, the Shapley interaction index $I^{Shap}(S)$ can be considered as a weighted sum of the Harsanyi dividend interaction over different subsets of players $T \subseteq N \backslash S$, i.e.,

$$I^{Shap}(S) = \sum_{T \subseteq N \backslash S} \frac{1}{|T|+1} I([S] \cup T). \tag{7}$$

Connections to multivariate interactions

Previous studies have proposed several interaction metrics to quantify interactions between multiple players, but these interaction metrics usually have significant computational cost. Therefore, we proposed a simplified multivariate interaction between a set of players in S, which is denoted by $B([S])$[13]. Specifically, $B([S])$ is defined as the additional award when players in the subset S interact with each other w.r.t. when players in S work individually. Here, we use the Shapley value to quantify the reward gained by a set of players, i.e.,

$$B([S]) = \phi([S]|N_{[S]}) - \sum_{i \in S} \phi(i|N_i). \tag{8}$$

In the computation of the Shapley value $\phi([S]|N_{[S]})$ of the subset S, we regard S as a singleton player $[S]$. Then, we consider that the game is played by players in $N_{[S]} \overset{\text{def}}{=} N \backslash S \cup \{[S]\}$, instead of N. Similarly, in the computation of the Shapley value $\phi(i|N_i)$ of the player i, we consider the game is played by players in $N_i \overset{\text{def}}{=} N \backslash S \cup \{i\}$. In other words, players in $S \setminus \{i\}$ do not participate in the game.

The multivariate interaction $B([S])$ can be used to explain interactions between input words encoded by NLP models, such as bidirectional encoder representations from Transformers (BERT), long short-term memory (LSTM), and Transformer. Specifically, we adopt $B([S])$ to quantify interaction utilities between two adjoining words (or phrases), and then we represent adjoining words (or phrases) with significant interaction strength using a tree structure.

Castro et al.[30] have proposed a method to fast approximate the Shapley value based on sampling, so the multivariate interaction $B[S]$ can be calculated efficiently. We prove that $B([S])$ can be considered as the sum of the Shapley interaction index of all possible subsets $L \subseteq S$, where each subset L contains at least two variables, i.e.,

$$B([S]) = \sum_{L \subseteq S, |L| > 1} I^{Shap}(L|N_L). \tag{9}$$

2.2.3 Multivariate interaction strength

The positive and negative interactions contained in the multivariate interaction $B[S]$ may neutralize each other, so the multivariate interaction $B[S]$ can not precisely reflect the interaction strength. Therefore, we propose a new metric to measure the overall strength of positive and negative interactions[13]. To this end, we design $B_{\max}([S])$ and $B_{\min}([S])$ to approximate the strength of all positive interactions within players in $S' \subseteq S$ and the strength of all negative interactions within players in $S' \subseteq S$, respectively. Specifically, we divide all input variables in S into K coalitions, $\Omega = [C_1, \cdots, C_K]$, which satisfy $\cup_{i=1}^{K} C_i = S,\ \forall 1 \leq i < j \leq K,\ C_i \cap C_j = \emptyset$. Given a specific partition of players Ω, we can compute overall interaction effects under such a partition as $B([S]) = \phi([S]|N_{[S]}) -$

$\sum_{i\in S}\phi(i|N_i)$. To approximate the strength of all positive interactions, $B_{\max}([S])$ is referred to as the maximum interaction utility over all potential partitions, as follows:

$$B_{\max}([S]) = \max_{\Omega}\sum_{C\in\Omega}\phi([C]|N_{[C]}) - \sum_{i\in S}\phi(i|N_i)$$

where $\phi([C]|N_{[C]})$ measures the reward gained by $[C]$ when we consider players in C as a singleton player $[C]$ over the set of players $N\backslash S\cup\{[C]\}$, and $\phi(i|N_i)$ measures the reward gained by the player i among all players in the set $N\backslash S\cup\{i\}$. Similarly, to approximate the strength of all negative interactions, we define $B_{\min}([S])$, as follows:

$$B_{\min}([S]) = \min_{\Omega}\sum_{C\in\Omega}\phi([C]|N_{[C]}) - \sum_{i\in S}\phi(i|N_i).$$

In this way, we further define the multivariate interaction strength $T([S]) = B_{\max}([S]) - B_{\min}([S])$ to measure the overall strength of both positive and negative interaction utilities,

$$T([S]) = \max_{\Omega}\sum_{C\in\Omega}\phi([C]|N_{[C]}) - \min_{\Omega}\sum_{C\in\Omega}\phi([C]|N_{[C]}). \quad (10)$$

Connections between $T([S])$ and $B([S])$

We demonstrate that the multivariate interaction strength $T([S])$ connects with the multivariate interaction $B([S])$. Based on (9), we accordingly define $B([S]) = \sum_{L\subseteq S,|L|>1}|I(L)|$ as the sum of $|I(L)|$, where $I(L) = I^{Shap}(L|N_L)$. In this way, $B([S])$ can be considered as equivalent to $B^+ - B^-$, where $B^+ = \sum_{L\subseteq S,|L|>1,I(L)\geq 0} I(L)$ represents the utility of all positive interactions, and $B^- = \sum_{L\subseteq S,|L|>1,I(L)<0} I(L)$ represents the utility of all negative interactions. Both $B_{\max}([S])$ and B^+ represent positive interaction utilities, and both $B_{\min}([S])$ and B^- represent negative interaction utilities. Thus, the multivariate interaction strength $T([S]) = B_{\max}([S]) - B_{\min}([S])$ has strong connections with the multivariate interaction $B([S]) = B^+ - B^-$.

In the computation of the multivariate interaction strength $T([S])$, we need to take all possible partitions Ω into consideration, which results in an unaffordable computational cost. To this end, we have proposed an approximation method to efficiently compute $T([S])$ based on sampling in [13].

2.2.4 Multi-order interactions

In order to quantify the interaction utilities of different complexities, we propose the multi-order interaction in [14]. Specifically, let us consider a game v where n players participate in the game. We can consider a trained DNN as a game, and regard n variables of the input sample as n players, $N = \{1,\cdots,n\}$. The multi-order interaction $I^{(m)}(i,j)$ between the input variable i and the input variable j is defined, as follows:

$$I^{(m)}(i,j) = \mathrm{E}_{S\subseteq N\backslash\{i,j\},|S|=m}\Delta v(i,j,S) \quad (11)$$

where $\Delta v(i,j,S) = v(S\cup\{i,j\}) - v(S\cup\{i\}) - v(S\cup\{j\}) + v(S)$. The m-order interaction $I^{(m)}(i,j)$ quantifies the average utility of the interaction between two variables i and j when variables i and j cooperate with different sets of m contextual variables.

In order to verify the trustworthiness of the multi-order interaction, we have proven that the multi-order interaction $I^{(m)}(i,j)$ satisfies the following properties in [22].

Linearity property. If a game u and a game v can be combined into a new game w, i.e., for $\forall S\subseteq N$, $w(S) = u(S) + v(S)$, then, multi-order interactions of the game u and game v can be also combined into the multi-order interaction of the new game w, i.e., $I_w^{(m)}(i,j) = I_u^{(m)}(i,j) + I_v^{(m)}(i,j)$.

Nullity property. If the collaboration between an input variable i and any subset of variables cannot bring any additional reward, i.e., $\forall S\subseteq N\setminus\{i\}$, $v(S\cup\{i\}) = v(S) + v(\{i\})$, then, the variable i is termed a dummy variable. The multi-order interaction between the dummy variable i and any other variables equals to zero, i.e., $\forall j\in N\setminus\{i\}$, $\forall m$, $I^{(m)}(i,j) = 0$.

Commutativity property. The interaction utilities between the variable i and the variable j are equal to the interaction utilities of the variable j and the variable i, i.e., $\forall i,\ j\in N$, $I^{(m)}(i,j) = I^{(m)}(j,i)$.

Symmetry property. Assume two variables i and j are equivalent in the sense that i and j have the same interactions with other variables, i.e., $v(S\cup\{i\}) = v(S\cup\{j\})$. Then, for any variable $k\in N$, we have $\forall m, I^{(m)}(i,k) = I^{(m)}(j,k)$.

Efficiency property (The details about the proof can be found in [22]). The multi-order interaction of different orders between different pairs of input variables can fit the exact model output score of a DNN.

$$v(N) - v(\emptyset) = \sum_{i\in N}\mu_i + \sum_{i,j\in N,i\neq j}\sum_{m=0}^{n-2} w^{(m)}I^{(m)}(i,j) \quad (12)$$

where $\mu_i = v(\{i\}) - v(\emptyset)$ represents the independent utility of the variable i, and $w^{(m)} = (n-1-m)/[n(n-1)]$.

Connection with the pairwise interaction between two variables

Furthermore, we show that the multi-order interaction $I^{(m)}(i,j)$ is trustworthy by proving that the multi-order interaction strongly connects the pairwise interaction between two variables in [14]. Specifically, the classical the pairwise Shapley interaction index $I^{Shap}(i,j)$ between two input variables i and j can be decomposed into the multi-order interactions.

$$I^{Shap}(i,j) = \frac{1}{n-1}\sum_{m=0}^{n-2} I^{(m)}(i,j).$$

3 Unifying attribution explanations

Attribution methods present a typical direction of explaining DNNs, which infer the attribution, importance, or saliency of each input variable to the output score of a DNN. However, attribution methods cannot directly explain either the concept representation of a DNN or the performance of a DNN.

In recent years, many attribution methods have been proposed based on different heuristics, but there is no solid theoretical foundation to summarize the essential mechanism shared by different attribution methods. Therefore, to improve the trustworthiness of attribution methods, we reformulate different attribution methods in a unified Taylor-interaction framework, and such a unified framework allows us to fairly compare common and distinctive mechanisms of different attribution methods. Furthermore, based on the Taylor-interaction framework, we establish three principles to evaluate the fairness of different attribution methods.

3.1 Taylor interactions

We propose the Taylor interaction[31] to measure the collaborative relationship between different input variables of a DNN, which enables us to explain the output score of a DNN from a new perspective.

Specifically, based on the Taylor interaction, the output score of a DNN can be fully explained as two typical effects caused by input variables. The first effect is that an individual input variable directly affects the output score without collaborating with other input variables. Such an effect is termed as the independent effect. The other effect is that different input variables collaborate with each other, thereby making a certain numerical effect on the output score. Such an effect is termed the interaction effect. For example, for the inference of a face, the nose region can increase the confidence of recognition without depending on other image regions, which corresponds to an independent effect. In addition, the image regions of eyes, nose, and mouth collaborate with each other and increase the confidence of recognition, which corresponds to an interaction effect. Accordingly, the output score can be explained as the sum of independent effects and interaction effects.

Let us further mathematically formulate how to explain the output score of a DNN as the sum of independent effects and interaction effects, as follows. We quantify these two effects by the proposed Taylor interaction, and prove that the output score $v(N)$ can be represented as the sum of two effects.

$$v(N) = v(\emptyset) + \sum_{j\in N} \phi(j) + \sum_{S\subseteq N, |S|>1} \sum_{\pi_S} I(S, \pi_S) \quad (13)$$

where $\phi(j)$ denotes the independent effect of the input variable j. In addition, $I(S, \pi_S)$ denotes the interaction effect of the input variable subset S with a degree vector π_S, where $\pi_S = [\kappa_1, \cdots, \kappa_n]$ satisfying $\forall i \in N,\ \kappa_i \in N$ and $S = \{i|\kappa_i \neq 0\}$. Specifically, given a DNN f,

$$I(S, \pi_S) = C(S, \pi_S) \bigtriangledown f(S, \pi_S) \prod_{i\in S} (x_i - b_i)^{\kappa_i}$$

where x_i and b_i denote the variable value and the baseline value of the i-th input variable, respectively. In addition, the coefficient term $C(S, \pi_S) = \frac{1}{n!}\binom{\kappa_1+\cdots+\kappa_n}{\kappa_1,\cdots,\kappa_n}$ and $\bigtriangledown f(S, \pi_S) = \frac{\partial^{\kappa_1+\cdots+\kappa_n} f(\boldsymbol{b})}{\partial^{\kappa_1} x_1 \cdots\cdots \partial^{\kappa_n} x_n}$ denotes the partial derivative of the DNN f at the baseline point $\boldsymbol{b} = [b_1, \cdots, b_n]$ given the degree vector π_S. Deng et al.[31] have proven that, the sum of $I(S, \pi_S)$ over all possible degree vectors π_S equals to the Harsanyi dividend interaction.

$$I(S) = \sum_{\pi_S} I(S, \pi_S). \quad (14)$$

Hence, we can consider that the interaction effect $I(S, \pi_S)$ with a degree vector π_S can explain the basic element of the Harsanyi dividend interaction. Furthermore, (7) have shown the connection between the Harsanyi dividend and the Shapley interaction index. Thus, combining (14) and (7), we can also represent the Shapley interaction index as

$$I^{Shap}(S) = \sum_{T\subseteq N\setminus S} \frac{1}{|T|+1} \sum_{\pi_{[S]\cup T}\in\Omega_{[S]\cup T}} I([S]\cup T, \pi_{[S]\cup T}).$$

3.2 Unifying and evaluating attribution explanations

Crucially, we discover and prove that the Taylor interaction can explain fourteen attribution methods[5, 32–39]. Hence, we consider that the Taylor interaction reveals the essential mechanism shared by different attribution methods. Specifically, the goal of an attribution method is to infer the contribution a_i of an input variable i to the output. Then, according to the above explanation of the output, we discover that the contribution of each input variable to the output can also be explained by independent effects and interaction effects in (15). Beyond this, we further prove that the attribution estimated by each attribution method can also be reformulated as a weighted sum of these two effects.

$$a_i = \sum_{j\in N} w_{i,j}\phi(j) + \sum_{S\subseteq N, |S|>1} \sum_{\pi_S} w_{i,\{S,\pi_S\}} I(S, \pi_S) \quad (15)$$

where $w_{i,j}$ denotes the ratio of the independent effect $\phi(j)$ of the variable j that is assigned with the contribution of the variable i. $w_{i,\{S,\pi_S\}}$ denotes the ratio of the interaction effect $I(S, \pi_S)$ of the variable subset S that is assigned with the contribution of the variable i.

Based on the Taylor interaction, we discover that the

essential difference of the mechanisms of fourteen attribution methods mainly lies in the strategy of an attribution method assigning interaction effects $I(S, \pi_S)$. For example, the Shapley value[24] evenly distributes interaction effects $I(S, \pi_S)$ to each variable in S, i.e., for $\forall i \in S$, $w_{i,\{S,\pi_S\}} = 1/|S|$. In comparison, Occlusion-1[40] distributes the whole interaction effect to each variable $i \in S$, which means that for $\forall i \in S$, $w_{i,\{S,\pi_S\}} = 1$.

Each attribution method's distinctive way of assigning interaction effects and independent effects determines its distinctive attribution scores. This motivates us to compare these attribution methods. Therefore, we establish three principles for the fairness of attribution methods.

1) For each input variable $j \in N$, its independent effect $\phi(j)$ should all be attributed (to input variables), i.e., $\sum_{i\in N} w_{i,j} = 1$. Similarly, for each input variable subset $S \subseteq N$ and each degree vector π_S, its interaction effect $I(S, \pi_S)$ should all be attributed (to input variables), i.e., for $\sum_{i\in N} w_{i,\{S,\pi_S\}} = 1$.

2) The independent effect of the input variable i should be attributed only to the variable i, i.e., $w_{i,i} = 1$ and for $\forall j \neq i$, $w_{i,j} = 0$.

3) Each interaction effect $I(S, \pi_S)$ between input variables in subset S should be and only be attributed to input variables in the subset S, instead of input variables out of S, i.e., $\sum_{i\in S} w_{i,\{S,\pi_S\}} = 1$, and for $\forall i \notin S, w_{i,\{S,\pi_S\}} = 0$.

We use the above three principles to evaluate fourteen attribution methods[5, 24, 32, 33, 40, 34–39]. For example, we find that Gradient $\times$ Input[32], Grad-CAM[34] and Occlusion-1[40] violate the first principle. LRP-$\alpha\beta$[33] and deep Taylor[38] violate the second and third principles. In comparison, the Shapley value[24], DeepLIFT[37] and integrated Gradients[36] satisfy all three principles, so they can be considered to generate fair attributions from the perspective of three principles.

4 Explaining concept representations

As mentioned above, explaining concept representations in DNNs can help us comprehensively understand the performance of DNNs. To this end, we use above game-theoretic interactions to explain the concept representation of DNNs from three perspectives. First, we decompose the DNN representation into a hierarchical interaction tree to explain the internal behaviors of the DNN[17]. Second, we define visual concepts and explain the essence of signal processing of visual concepts encoded in DNNs[20]. Third, we explore prototypical concepts that are encoded in the DNN[21].

4.1 Decomposing DNN representations into axiomatic and hierarchical And-Or graphs

Many explanations have been proposed to explain DNNs, but there is no guarantee for their objectiveness, which hurts the trustworthiness of these explanations. We have shown that explainer models learned by knowledge distillation cannot objectively reflect the underlying attention of the target DNN[17].

To this end, there are essentially two challenges for an objective explanation from the perspective of concept representations. First, there is not a standard definition for the objectiveness of explanations. Second, we need to precisely disentangle and explain all concepts encoded in the DNN, in order to examine the objectiveness of the explanation. We propose a possible solution to the above issues in [17].

Definition of the objectiveness of explanations. We first define the objectiveness of explanations as follows. For a target model v, an objective explainer model g is supposed to ensure that causal factors for inference in the explainer model g are exactly the same as causal factors in the target model v. Causal factors can be represented as Harsanyi dividend interactions between input variables $I(S)$ (in Section 2.2) encoded by the model. Therefore, we define the objectiveness of an explainer model as follows.

Definition 1 (Objectiveness of explanations). Given a specific input sample $\boldsymbol{x} \in \mathbf{R}^n$, an objective explainer model g is supposed to encode exactly the same Harsanyi dividend interactions as the target model v w.r.t. all cases.

$$\forall S \subseteq N, I_g(S) = I_v(S) \tag{16}$$

where $I_g(S)$ and $I_v(S)$ denote the Harsanyi dividend interaction of variables in S encoded in the explainer model g and the Harsanyi dividend interaction encoded in the target model v, respectively.

This definition ensures that the explainer model encodes exactly the same causal factors for inference as the target model.

Existence of the explanation satisfying the above objectiveness. We have proven that the Harsanyi dividend interaction is exactly an objective explanation that satisfied the above definition. Furthermore, based on (3), the model output $v(N)$ can be decomposed into the sum of Harsanyi dividend interactions $I(S)$ of all subsets S. Therefore, we consider (3) as an And-Sum representation of the model output, where each interaction $I(S)$ represents the AND relationship between variables in S. The SUM relationship refers to that all interactions $I(S)$ sum up to the model output $v(N)$. In this way, as Fig. 3 shows, we build up an And-Or graph (AOG) based on the And-Sum representation. Moreover, we propose three techniques to further simplify the AOG explanation.

4.2 Explaining visual concepts

Using visual concepts to explain DNNs is also a clas-

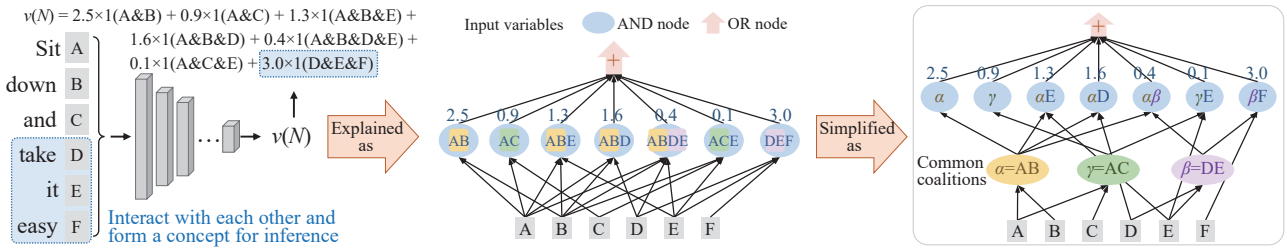


Fig. 3 We represent casual factors encoded in a DNN into an And-Or graph (AOG), and we further simplify the AOG explanation by extracting common coalitions among different casual factors[17].

sical direction of explainable AI. For example, the visualization of network features[1–3] aimed to illustrate visual concepts encoded by DNNs, and attribution methods[4–6] were related to discovering important visual concepts contained in the image. Although these methods could explain DNNs effectively, they still ignored the core issue that previous methods usually did not define and model visual concepts theoretically. Instead, these studies mainly analyzed how visual concepts were distributed in different layers in an experimental manner.

Hence, without theoretical definitions of visual concepts, it is difficult to build up a clear connection between the traditional taxonomy of visual concepts and the concept representation of a DNN. In other words, the traditional taxonomy of visual concepts mainly relies on human cognition, which cannot correctly reflect distinctive signal-processing behaviors of a DNN encoding different visual concepts. Specifically, people usually classify visual concepts into shapes with rich structural information and textures without clear structural information[41]. Such a cognitive classification of visual concepts into textures and shapes can be further subdivided into objects, parts, scenes, textures, materials and colors[42].

To this end, we provide a new way to categorize visual concepts w.r.t. their complexities, which can better reflect how a DNN processes visual features to encode visual concepts[20]. Specifically, we represent visual concepts of different complexities using multi-order interactions (defined in Section 2.2). To this end, simple/local concepts usually correspond to low-order interactions, and complex/global concepts are often referred to as high-order interactions. Thus, the concept complexity enables us to analyze the concept representation of the DNN.

Using multi-order interactions to explain visual concepts. We discover that low-order interactions usually reflect common visual concepts without rich structural information. In comparison, middle/high-order interactions tend to encode concepts with relatively more complex structural (textural/shape) information.

Using multi-order interactions to explain textures and shapes. Furthermore, we analyze the difference between the behavior when the DNN models textural concepts and the behavior when the DNN models shape concepts. We discover that signal-processing behaviors of modeling textural concepts are much more flexible than those of modeling shape concepts from the perspective of multi-order interactions. This is because a textural concept can be encoded flexibly as either a large number of repeated simple and local textures, or a small number of middle-complexity textures whose appearances are memorized as specific contextual patterns by the DNN. In comparison, the modeling of shape concepts is not as flexible as the modeling of textural concepts, i.e., the shape concept is usually represented as a relatively stable distribution of interactions over different orders. Especially, if the shape concept is mainly represented by high-order interactions, then this shape concept may represent specific large-scale shapes or out-of-the-distribution samples.

4.3 Explaining prototypical concepts

Based on the analysis of visual concepts, a more interesting problem is to explore the prototypical concepts encoded in a DNN, which is also a typical direction in explainable AI.

To this end, we provide a method to model and disentangle prototypical concepts contained by an image, and further revise this image to make prototypical concepts in the image more salient[21]. Specifically, we find that the DNN usually represents an image as a mixture of concepts. Thus, we propose a hypothesis that prototypical concepts usually more strongly activate the DNN than non-prototypical concepts. Therefore, removing plenty of non-prototypical concepts from the image usually forces the DNN to pay more attention to prototypical concepts.

A major challenge of verifying the above hypothesis is how to mathematically define prototypical concepts and non-prototypical concepts encoded in a DNN. Fortunately, we have proven that the visual concepts encoded in a DNN can be represented by the Harsanyi dividend, which has been introduced in Section 2.2[17]. Specifically, considering an image consisting of n patches $N = \{1, 2, \cdots, n\}$, we define the visual concept as a subset of patches $S \subseteq N$ that have strong interaction utilities (or the Harsanyi dividend interaction) with each other. In this way, we have proven that feature representation can be represented as the sum of Harsanyi dividend interactions of different visual concepts, i.e., $feature = \sum_{S \subseteq N} I(S)$. We extend the vanilla Harsanyi dividend in-

to a vector $I(S)$, which measures the Harsanyi dividend with respect to the computation of a vectorized intermediate-layer feature of the DNN.

We consider visual concepts that have salient influence on feature representations as prototypical concept′s, i.e., the prototypical concept S usually has a relatively large interaction effect $\|I(S)\|_2$. Among a huge number of visual concepts, most visual concepts have little impact on feature representations, and these visual concepts are considered as non-prototypical concept′s, i.e., the non-prototypical concept S usually has negligible interaction effects $\|I(S)\|_2$.

In this way, to verify the hypothesis, we revise the original image by enhancing prototypical concepts and removing non-prototypical concepts. If such an image revision makes the image look more prototypical, then we believe that the hypothesis was somewhat trustworthy.

Fig. 4 shows that the revised image tends to be more prototypical than the original image, e.g., the tree in the image turns greener. Thus, we consider that the above hypothesis is successfully verified.

Moreover, in order to verify that prototypical concepts make an image easier to perceive, we increase or decrease the number of prototypical concepts m_2 to revise an image. Fig. 5 shows that, when the number of prototypical concepts m_2 is large, the revised image $\hat{x}$ tends to exhibit a more prototypical appearance. Such a phenomenon verifies the effectiveness of the proposed method to model and disentangle prototypical concepts.

5 Explaining the representation power of a DNN

The new perspective of concept representation enables us to explain the representation power of a DNN in a fine-grained manner. In other words, we can explain the representation power of a DNN by directly exploring properties of various concepts encoded by the DNN. More crucially, such an explanation enables us to unify different methods of boosting adversarial transferability, and helps us extract the common mechanism of these methods. To this end, we use game-theoretic interactions to explain DNNs from four perspectives, including adversarial transferability, adversarial robustness, generalization power, and representation bottleneck of DNNs.

5.1 Unifying studies of boosting adversarial transferability

In recent years, adversarial transferability has attracted much attention in the field of deep learning. Many approaches have been proposed to improve adversarial

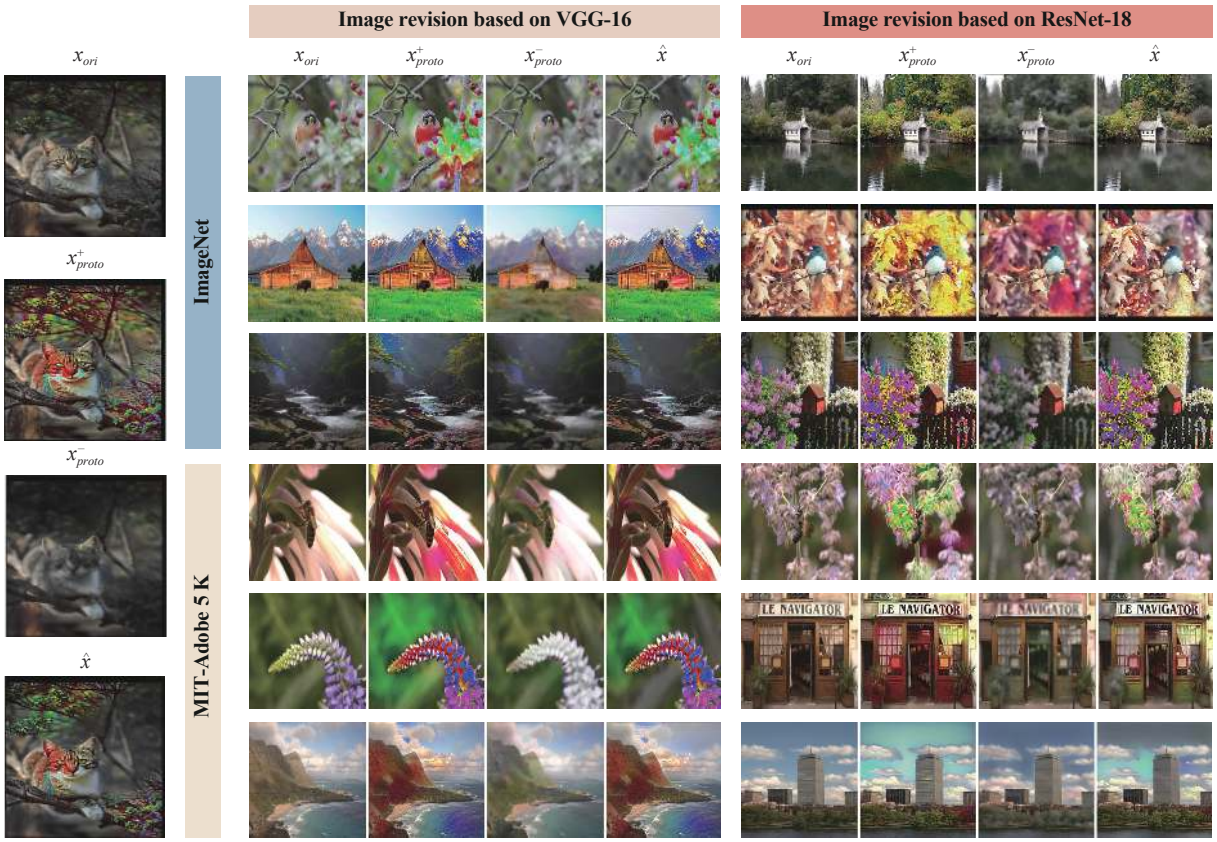


Fig. 4 We revise the original image x_{ori} by enhancing prototypical concepts and removing non-prototypical concepts, and get the revised image $\hat{x}$[21]. We revise images in the ImageNet dataset based on VGG-16 and ResNet-18, and images in the MIT-Adobe 5K dataset based on VGG-16 and ResNet-18, respectively.

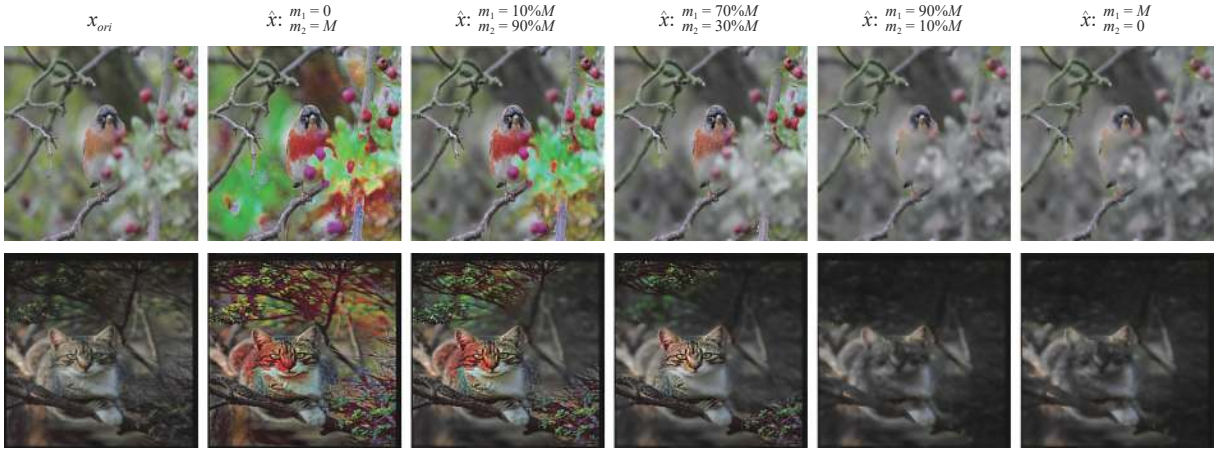


Fig. 5 Comparisons of images revised to contain different numbers of m_1 non-prototypical concepts and m_2 prototypical concepts[21]. The image $\hat{x}$ is revised by strengthening prototypical concepts and weakening non-prototypical concepts.

transferability. However, the intrinsic mechanism of these methods to improve the adversarial transferability remains unclear. It is of vital importance to theoretically explain common effects of previous transferability-boosting approaches, thereby guiding researchers to develop effective defense methods. Furthermore, the theoretical explanation may also help people discover theoretical flaws in previous methods and boost their performance. It is because some heuristic methods may not purely boost the adversarial transferability, but sometimes also make opposite effects.

Therefore, an essential explanation for adversarial transferability is supposed to reflect the common mechanism shared by different transferability-boosting methods. More crucially, such a theoretical explanation is also supposed to further guide us to refine existing heuristic methods and boost the transferability of adversarial perturbations.

To this end, we propose a unified explanation for the adversarial transferability from the perspective of game-theoretic interactions[23]. Specifically, adversarial examples are usually formulated as follows. Given a pre-trained DNN h and an input sample $\boldsymbol{x}$, let $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{\delta}$ denote the adversarial example, where $\boldsymbol{\delta}$ denotes the adversarial perturbation. The adversarial perturbation $\boldsymbol{\delta}$ is obtained as follows:

$$\boldsymbol{\delta} = \arg\max_{\boldsymbol{\delta}} \ell(h(\boldsymbol{x} + \boldsymbol{\delta}), y), \quad \text{s.t.} \quad \|\boldsymbol{\delta}\|_p \le \epsilon \tag{17}$$

where $h(\cdot) \in \mathbf{R}^{\mathbf{C}}$ denotes the DNN output before the softmax layer. ℓ denotes the loss for classification, and y denotes the ground-truth label of the input sample $\boldsymbol{x}$. ϵ is a constant to constrain the norm of the adversarial perturbation. Each dimension of the perturbation δ_i is regarded as a perturbation unit.

In this case, we can use $N = \{1, 2, \cdots, n\}$ to denote the set of all the n perturbation units in $\boldsymbol{\delta}$. The attacking utility of perturbation units in the subset $S \subseteq N$ is defined as $v(S) = \max_{y \neq y} h_y(\boldsymbol{x} + \boldsymbol{\delta}^{(S)}) - h_y(\boldsymbol{x} + \boldsymbol{\delta}^{(S)})$, where $h_y(\cdot)$ denotes the value of the y-th dimension of $h(\cdot)$. $\boldsymbol{\delta}^{(S)}$ denotes the perturbation when only perturbation units in S are perturbed, i.e., $\forall i \in S$, $\delta_i^{(S)} = \delta_i$; $\forall i \notin S$, $\delta_i^{(S)} = 0$. Therefore, the pairwise interaction between two perturbation units i, j can be represented as follows:

$$I^{Shap}(i,j) = \sum_{S \subseteq N \setminus \{i,j\}} \frac{s!(n-s-2)!}{(n-1)!} \Delta_{\{i,j\}} v(S) \tag{18}$$

where $s = |S|$ and $\Delta_{\{i,j\}} v(S) = v(S \cup \{i,j\}) - v(S \cup \{i\}) - v(S \cup \{j\}) + v(S)$.

Based on the above game-theoretic interaction $I^{Shap}(i,j)$ between two perturbation units i, j, we have found that the adversarial transferability and game-theoretic interactions are negatively correlated. First, we verify that perturbations generated from single-step adversarial attacking methods usually encode weaker game-theoretic interactions than those generated from multi-step adversarial attacking methods. Previous study[43] have found that perturbations generated from single-step adversarial attacking methods are more transferable than those generated from multi-step adversarial attacking methods. Therefore, we deduce that the adversarial transferability and game-theoretic interactions are negatively correlated. Experimental results in Fig. 6 have verified this conclusion.

Based on the above finding, we have proven a unified explanation for the adversarial transferability, i.e., reducing game-theoretic interaction is the common effect shared by various transferability-boosting methods. Specifically, we prove and verify that the following five previous transferability-boosting methods all reduce the game-theoretic interaction between perturbation units.

Momentum iterative attack (MI Attack).[44] Gradient momentum is added to the optimization process of adversarial perturbations.

Variance-reduced attack (VR attack).[45] During the attack, Gaussian noise is added to the input image to

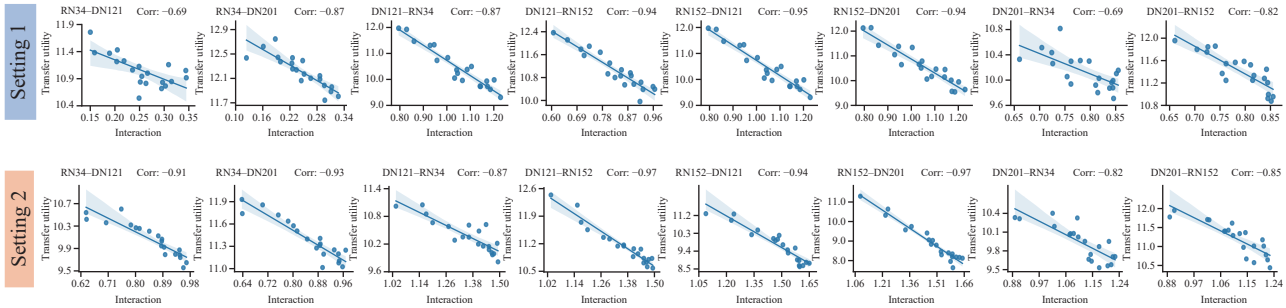


Fig. 6 We verify that the adversarial transferability and game-theoretic interactions are negatively correlated.[23]

smooth the gradient against the input image.

Skip gradient method attack (SGM attack).[46] For the residual block structure of the DNN, this method increases the gradient weight of the skip-connection branch in the back-propagation process.

Diversity input attack (DI attack).[43] During the attack, this method introduces random padding and resizing of the input image to increase the diversity.

Translation invariant attack (TI attack).[47] This method generates adversarial perturbations by convolving the gradient w.r.t. a set of translated versions of the original images during the attack.

Because we have found that many transferability-boosting methods all share the same mechanism of reducing the interaction, such a common mechanism guides us to propose a new loss function to boost the adversarial transferability. We propose the interaction-reduced loss (IR loss) to directly penalize the game-theoretic interaction between perturbation units $I^{Shap}(i,j)$ during the attack as follows:

$$\ell_{\text{IR}} = \text{E}_{i,j\in N, i\neq j} I^{Shap}(i,j). \tag{19}$$

During the attack, we generate and obtain adversarial perturbations by jointly minimizing the classification loss ℓ and the IR loss ℓ_{IR}. We term this method as the interaction-reduced attack (IR Attack). The IR Attack can also be combined with previous transferability-boosting methods to further boost their performance, which is termed the HybridIR Attack. As Table 2 shows, with the additional interaction-reduction (IR) loss, adversarial transferability is boosted on both the L_∞ attack and the L_2 attack.

In fact, our latest study has found that the decrease of the interaction is actually the common mechanism of a total of twelve transferability-boosting methods[44–46, 48–56]. Moreover, we can use this mechanism to further refine the algorithmic design of three previous methods[52, 53, 57].

5.2 Explaining adversarial robustness

Besides adversarial transferability, adversarial robustness is another important problem, whose mechanism has not been fully explained. There have been many methods to boost adversarial robustness of DNNs, but we still cannot understand the essence of adversarial examples and robustness. To this end, some studies[10, 58] proved mathematical bounds for robustness. Some studies[59, 60] explored the connection between adversarial robustness and network interpretability. However, these studies could not reveal the essential reason for robustness of DNNs, i.e., which types of features made the DNN robust and why the DNN learned such non-robust features.

To this end, we propose to explain and understand adversarial robustness from the perspective of game-theoretic interactions[22]. We can first explain effects of adversarial examples. We theoretically prove and experimentally verify that adversarial perturbations mainly affect high-order interactions between input variables. More specifically, given the input sample $\boldsymbol{x}$, let $I^{(m)}(i,j)$ denote the m-order interaction between two input variables i,j in the input sample $\boldsymbol{x}$. We compare multi-order interactions between variables in the original sample $\boldsymbol{x}$ and multi-order interactions between variables in the adversarial example $\tilde{\boldsymbol{x}}$. Fig. 7 shows that high-order interactions are more sensitive to perturbations and are easier to be affected by adversarial attacks than low-order interactions.

Furthermore, we find that adversarial training enhances the discrimination power of low-order interactions encoded in a DNN, thereby boosting the robustness of high-order interactions and the robustness of the model. Specifically, we find that low-order interactions in adversarially-trained DNNs have more attacking utilities than those in normally-trained DNNs. Meanwhile, high-order interactions in adversarially-trained DNNs are more robust than those in normally-trained DNNs. To explain this phenomenon, we define a metric $D^{(m)}$ to measure the discriminative power of m-order interactions, which is termed as the disentanglement metric. A large value of $D^{(m)}$ indicates that m-order interactions purely describe a specific category. Fig. 8 shows that in adversarially-trained DNNs, low-order interactions exhibit larger values of $D^{(m)}$ than low-order interactions in normally-trained DNNs. This demonstrates that adversarial training boosts the discriminative power of low-order interactions, which yields more robust high-order interactions.

The above explanation enables us to unify the mech-

Table 2 Transferability with and without the interaction loss[23]: The success rates of ℓ_∞ and ℓ_2 black-box attacks crafted on six source models, including AlexNet, VGG16, RN-34/152 and DN-121/201, against seven target models. Penalizing interactions between perturbation units boosted the transferability of adversarial perturbations.

Source	Method	VGG-16	RN152	DN-201	SE-154	IncV3	IncV4	IncResV2
AlexNet	PGD ℓ_∞	67.0±1.6	27.8±1.1	32.3±0.4	28.2±0.7	29.1±1.5	23.0±0.4	18.6±1.5
	PGD ℓ_∞+IR	**78.7±1.0**	**42.0±1.5**	**50.3±0.4**	**41.2±0.6**	**43.7±0.5**	**36.4±1.5**	**29.0±1.0**
DN-121	PGD ℓ_∞	68.6±1.1	63.6±3.2	86.9±1.5	46.1±1.5	37.3±1.6	37.1±2.1	28.9±2.8
	PGD ℓ_∞+IR	**85.0±0.3**	**84.8±0.4**	**95.1±0.2**	**70.3±1.7**	**61.1±2.5**	**62.1±2.0**	**53.5±0.3**
DN-201	PGD ℓ_∞	64.4±1.4	67.8±0.2	–	50.9±0.8	39.5±3.3	36.5±0.9	34.2±0.4
	PGD ℓ_∞+IR	**78.6±2.5**	**85.0±1.1**	–	**73.9±0.5**	**61.6±1.8**	**63.7±0.6**	**56.4±2.1**
RN-34	PGD ℓ_∞	65.4±2.9	59.2±2.7	63.5±3.3	33.1±2.9	27.4±3.6	23.9±1.7	21.1±1.1
	PGD ℓ_∞+IR	**84.0±0.5**	**84.7±2.3**	**88.5±0.9**	**64.4±1.6**	**56.9±3.1**	**59.3±4.3**	**49.2±1.1**
RN-152	PGD ℓ_∞	51.6±3.2	–	61.5±2.4	33.9±1.5	28.1±0.9	25.0±1.2	22.4±1.0
	PGD ℓ_∞+IR	**72.3±1.2**	–	**82.1±1.3**	**61.1±0.9**	**53.6±0.8**	**50.6±3.5**	**46.0±2.3**
VGG-16	PGD ℓ_∞	–	43.0±1.8	48.3±2.0	52.9±2.7	39.3±0.7	49.3±1.1	29.7±2.0
	PGD ℓ_∞+IR	–	**63.1±1.6**	**70.0±1.1**	**71.2±1.5**	**57.6±1.0**	**68.6±3.2**	**49.2±1.2**
AlexNet	PGD ℓ_2	85.1±1.5	58.9±1.0	60.2±2.1	55.1±1.5	56.0±3.7	49.6±3.4	44.6±3.3
	PGD ℓ_2+IR	**91.6±1.1**	**72.0±1.6**	**76.8±1.0**	**69.0±1.0**	**73.0±0.8**	**63.1±2.1**	**59.4±1.9**
DN-121	PGD ℓ_2	89.4±1.1	86.8±1.0	97.6±1.0	75.6±1.7	70.1±2.9	70.4±4.4	66.5±4.7
	PGD ℓ_2+IR	**94.2±0.1**	**93.3±0.8**	**97.7±0.3**	**87.8±0.7**	**84.5±0.7**	**84.2±0.1**	**82.4±0.1**
RN-34	PGD ℓ_2	88.2±1.4	86.2±0.4	89.6±1.3	66.9±1.1	64.2±2.9	60.0±1.9	55.2±1.8
	PGD ℓ_2+IR	**95.2±0.2**	**95.4±0.1**	**96.7±0.6**	**86.7±1.2**	**84.3±0.6**	**81.8±1.9**	**80.4±1.9**
VGG-16	PGD ℓ_2	–	76.7±0.9	82.3±2.9	83.5±1.9	77.5±3.6	82.1±2.2	69.4±2.1
	PGD ℓ_2+IR	–	**86.5±0.9**	**88.9±1.5**	**89.6±1.2**	**85.2±1.1**	**88.3±1.4**	**80.4±0.4**

anism of the following four adversarial defense methods.

The ML-LOO detection method. Yang et al.[61] proposed to detect adversarial examples from normal samples based on the attribution of inputs. We prove that this method actually utilizes the most sensitive components (high-order interactions) in DNNs to distinguish adversarial examples and normal samples.

The cutout training method. DeVries and Taylor[62] proposed to randomly mask a region of input images to enhance the robustness of the model. We prove that this method discards sensitive high-order interactions encoded by the DNN, thereby boosting the robustness of the DNN.

The rank-based method. Jere et al.[63] found that normal samples contain more low-rank features (w.r.t. the singular value decomposition), and adversarial examples are usually composed of high-rank features (w.r.t. the singular value decomposition). Specifically, given the SVD of the original image, low-rank features refer to images that are constructed by using the largest k singular values and their corresponding singular vectors. In comparison, high-rank features refer to images that are constructed by using the smallest k singular values and their corresponding singular vectors. We demonstrate that high-order interactions can better explain the difference between normal samples and adversarial examples.

The high recoverability of adversarial examples in adversarially-trained DNNs.[64] The adversarial recoverability refers to as the ability of minimizing the classification loss to invert an adversarial example to the original sample without being perturbed. We discover that adversarial examples generated on adversarially-trained DNNs exhibit higher recoverability than those generated on normally-trained DNNs. This can be explained by the phenomenon that adversarial training usually learns low-order interactions with much stronger discriminative power than high-order interactions. Low-order interactions are usually less affected by adversarial attacks and are easier to be recovered.

5.3 Explaining dropout and generalization

Besides the adversarial robustness, explaining and improving the generalization power of DNNs is still a considerable challenge for deep learning. The dropout operation can effectively improve the generalization power of DNNs. To explain the effectiveness of the dropout operation, some studies regarded the dropout operation as a

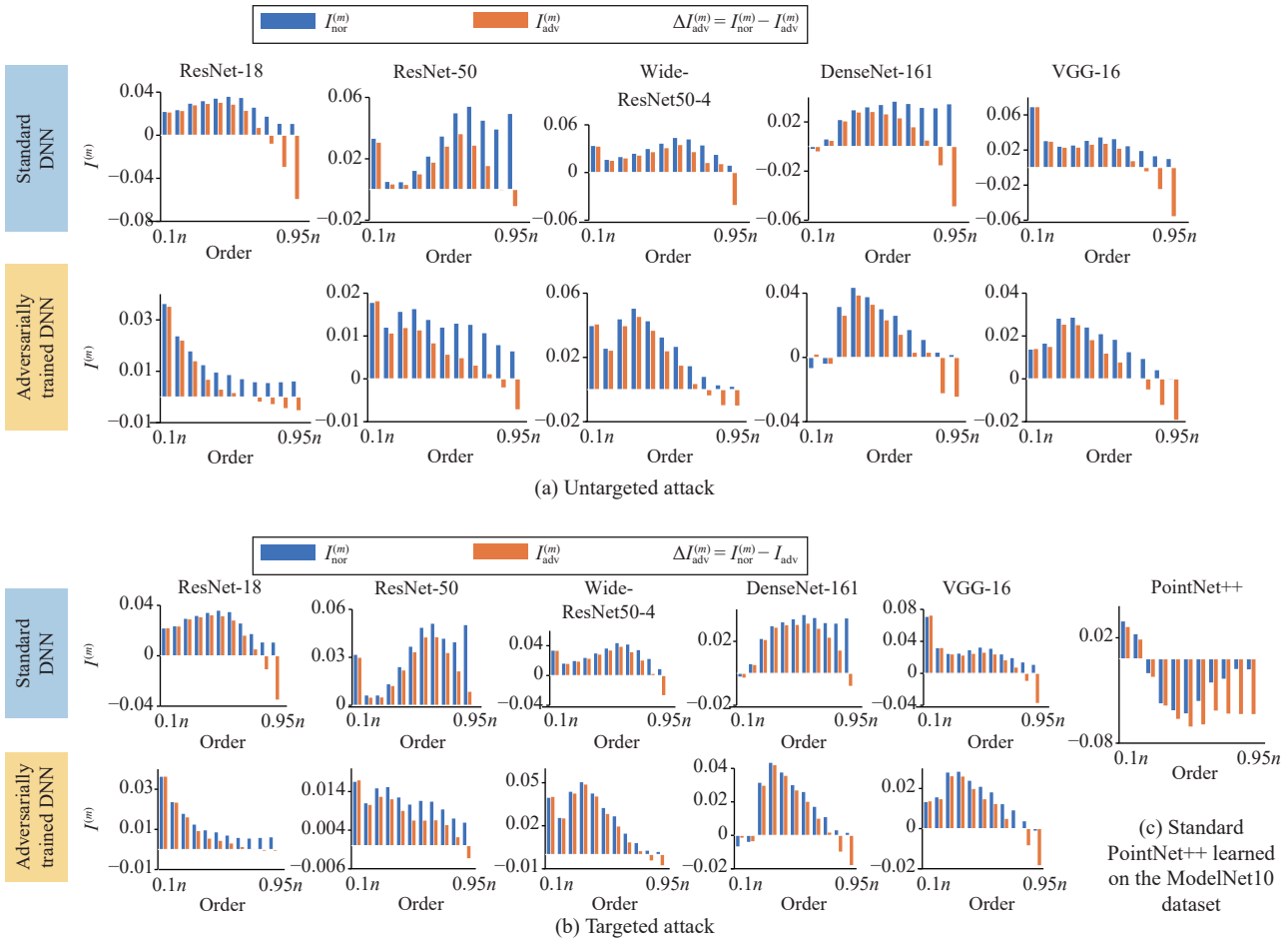


Fig. 7 Multi-order interactions encoded in normally-trained DNNs and adversarially-trained DNNs[22]. We find that high-order interactions are more sensitive to adversarial perturbations than low-order interactions.

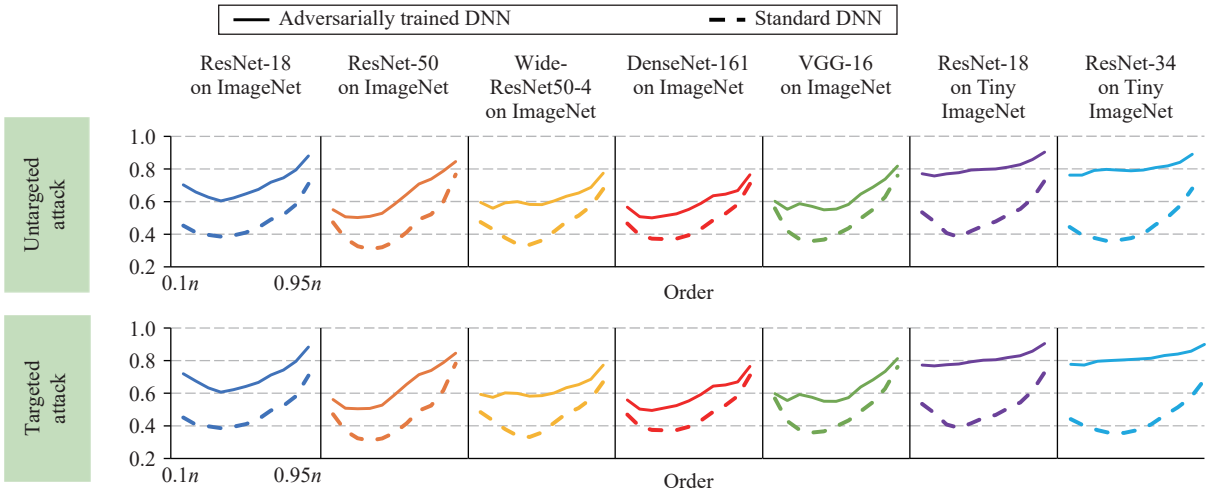


Fig. 8 Disentanglement metric $D^{(m)}$[22]. We find that low-order interactions in adversarially-trained DNNs exhibit larger values of $D^{(m)}$ than low-order interactions in normally-trained DNNs.

method of data augmentation[65], some studies considered that the dropout reduced the dependence between different feature units[66]. However, these studies only qualitatively analyzed how the dropout operation improved the generalization power of a DNN, but did not quantify the effect of the dropout operation.

In contrast to previous studies, we use multi-order interactions, which are defined in Section 2.2, to represent interaction utilities between different input variables. Next, we explore influences of the dropout operation on

interactions encoded in a DNN. We explain how the dropout operation improves the generalization power of a DNN by verifying the relationship between the generalization power of a DNN and interactions encoded in the DNN. We find a negative relationship between the interaction strength and the generalization power. Therefore, we propose a loss function to boost the effect of the dropout operation, thereby further improving the generalization power of a DNN[14].

Specifically, we verify that the dropout operation decreases the strength of interactions encoded in a DNN. Given an input sample with n input variables, $N = \{1, 2, \cdots, n\}$, some input variables collaborate to make impacts on the inference. The m-order interaction $I^{(m)}(i, j)$ measures the average interaction utility between input variables i, j under all possible context $S \subseteq N$ with m contextual variables. When the dropout operation randomly drops some input variables in N, the m-order interaction $I^{(m)}_{\text{dropout}}(i, j)$ is only computed based on different subsets of contexts S, which are composed of m input variables that are not dropped. In this way, the dropout operation makes the feature be computed based on much fewer contexts, thereby suppressing strength of interactions encoded in the feature.

Besides the above finding that the dropout operation decreases the interaction strength, we also find that the decrease of the interaction strength improves the generalization power of a DNN. In this way, we propose a new loss function to boost the effect of the dropout operation. We propose an interaction loss to directly penalize the strength of game-theoretic interactions to further improve the generalization power of a DNN. During the training process, we jointly minimize the classification loss and the interaction loss.

$$\ell_{\text{interaction}} = \mathrm{E}_{i,j\in N, i\neq j}\left[\left|\frac{1}{n-1}\sum_{m=0}^{n-2} I^{(m)}(i,j)\right|\right]. \quad (20)$$

We added the above interaction loss to the classification loss to directly control the significance of interactions modeled by the DNN. $Loss = \ell_{\text{classification}} + \lambda \times \ell_{\text{interaction}}$. We trained AlexNet, VGG-11, VGG-13 and VGG-16 on the CIFAR-10 dataset; RN-18, RN-34, VGG-16 and VGG-19 on the Tiny ImangeNet dataset; and VGG-13, VGG-16 and RN-18 on the Gender estimation dataset. As Table 3 shows, the interaction loss could effectively control the DNN from over-fitting to under-fitting. When the weight for the interaction loss was properly selected, the DNN trained using the interaction loss outperformed the DNN trained using the dropout.

5.4 Explaining the representation bottleneck

Besides the explanation of adversarial transferability, adversarial robustness and generalization power of DNNs, we also focus on another essential problem about the representation capacity of DNNs, i.e., common limitations in feature representations of different DNNs. Or more specifically, which types of concepts is a DNN unlikely to learn[19]? The exploration of common limitations of DNNs in feature representations is crucial to boost the representation capacity of DNNs. Various previous studies have proposed to explore common limitations of DNNs, e.g., theoretically maximum complexity, generalization power, adversarial robustness, and etc. In comparison, we firstly investigate the bottleneck of feature representations of a DNN.

Specifically, we use multi-order interactions to measure numerical utilities of interaction concepts between two input variables $i, j \in N$ at a certain complexity level, i.e., $I^{(m)}(i, j) \overset{\text{def}}{=} \mathrm{E}_{S\subseteq N\setminus\{i,j\},|S|=m}\Delta v(i, j, S)$, which are defined in Section 2.2. Here, each interaction $I^{(m)}(i, j)$ corresponds to an interaction concept, and m represents the complexity of the interaction concept, where $0 \leq m \leq n-2$. The efficiency property of multi-order interactions indicates that each interaction concept makes a compositional contribution to the output, i.e., $v(N) - v(\emptyset) = \sum_{i\in N}[v(\{i\}) - v(\emptyset)] + \sum_{i,j\in N, i\neq j}\sum_{m=0}^{n-2} w^{(m)} I^{(m)}(i, j)$, where $w^{(m)} = \frac{n-1-m}{n(n-1)}$. Therefore, we can take the numerical utility of an interaction concept, i.e., $I^{(m)}(i, j)$, as the underlying reason to explain the DNN.

In this way, we define the relative strength $J^{(m)}$ of the concept representation of the m-th order, as follows:

Table 3 Classification accuracy when the DNNs are controlled from over-fitting to under-fitting[14]

Gender estimation					Tiny ImageNet						CIFAR-10 dataset				
λ	RN-18	λ	VGG-13	VGG-16	λ	VGG-16	VGG-19	λ	RN-18†	RN-34†	λ	AlexNet†	VGG-11†	VGG-13†	VGG-16†
0.0	92.7	0.0	94.6	93.7	0.0	33.4	37.6	0.0	48.8	45.6	0.0	66.2	61.9	60.8	62.0
0.001	93.0	5.0	94.8	93.8	50.0	38.4	38.2	0.001	50.0	48.4	50.0	69.2	63.9	64.0	63.8
0.003	**93.1**	10.0	94.7	**94.6**	100.0	38.0	38.6	0.003	49.6	49.0	100.0	69.6	64.3	65.4	64.5
0.01	93.0	20.0	**94.9**	94.1	200.0	38.2	39.0	0.01	**52.2**	**49.6**	200.0	69.6	65.3	65.9	64.7
0.03	92.9	50.0	94.7	94.08	500.0	**42.8**	41.8	0.03	50.4	48.8	500.0	**70.0**	65.9	**66.2**	**64.9**
–	–	100.0	94.7	94.3	1 000.0	40.8	**45.2**	–	–	–	1 000.0	64.3	**66.3**	66.0	64.5
Dropout	92.1	Dropout	94.6	92.4	Dropout	36.8	32.6	Dropout	47.4	46.0	Dropout	67.5	60.9	60.9	63.0

$$J^{(m)} = \frac{E_{x\in\Omega}[E_{\{i,j\}}[|I^{(m)}(i,j|x)|]]}{E_m\left[E_{x\in\Omega}[E_{\{i,j\}}[|I^{(m)}(i,j|x)|]]\right]} \tag{21}$$

where Ω denotes the set of all input samples. The relative strength of the m-th order $J^{(m)}$ is averaged over all pairs of input variables in all input samples. The distribution of the relative strength $J^{(m)}$ measures the distribution of the complexity of interaction concepts encoded in DNNs.

We discover that there exists a common representation bottleneck for different DNNs in encoding interaction concepts, i.e., a DNN usually tends to encode both too complex interaction concepts and too simple interaction concepts, but it is difficult for a DNN to encode interaction concepts of intermediate complexity. Specifically, the relative strength $J^{(m)}$ of low-order interaction concepts and the relative strength of high-order interaction concepts are usually high. In comparison, the relative strength $J^{(m)}$ of middle-order interaction concepts is usually low ($m \approx 0.5n$).

Beyond above empirical discovery, we theoretically prove the mechanism for the representation bottleneck in Lemma 1 and Theorem1. Let W denote network parameters of a DNN. Then, the change ΔW of parameters represents the strength of training the DNN. In addition, we use L to denote the loss function and use η to denote the learning rate.

Lemma 1 (proven in [19]). The change ΔW of network parameters can be decomposed into the sum of gradients $\frac{\partial I^{(m)}(i,j)}{\partial W}$ of multi-order interactions.

$$\Delta W = \Delta W_U + \sum_{m=0}^{n-2}\sum_{i,j\in N, i\neq j}\Delta W^{(m)}(i,j)$$
$$\Delta W_U = -\eta\frac{\partial L}{\partial v(N)}\frac{\partial v(N)}{\partial U}\frac{\partial U}{\partial W}$$
$$\Delta W^{(m)}(i,j) = R^{(m)}\frac{\partial I^{(m)}(i,j)}{\partial W}$$

where $R^{(m)} = -\eta\frac{\partial L}{\partial v(N)}\frac{\partial v(N)}{\partial I^{(m)}(i,j)}$. Here, $\Delta W^{(m)}(i,j)$ represents the component of ΔW w.r.t. the gradient of the multi-order interaction $I^{(m)}(i,j)$. Besides, ΔW_U represents the component of learning independent effects U, where $U = v(\emptyset) + \sum_{i\in N}[v(\{i\}) - v(\emptyset)]$.

Theorem 1 (proven in [19]). Assume $\mathrm{E}_{i,j,S}[\frac{\partial\Delta v(i,j,S)}{\partial W}] = 0$. Let σ^2 denote the variance of $\frac{\partial\Delta v(i,j,S)}{\partial W}$. Then, for $\Delta W^{(m)}(i,j) = R^{(m)}\frac{\partial I^{(m)}(i,j)}{\partial W}$,

$$\mathrm{E}_{i,j}[\Delta W^{(m)}(i,j)] = 0, \text{ and}$$
$$var[\Delta W^{(m)}(i,j)] = \left(\eta\frac{\partial L}{\partial v(N)}w^{(m)}\right)^2\sigma^2/\binom{n-2}{m}$$

where $w^{(m)} = (n-1-m)/[n(n-1)]$, i.e, the learning strength $|\Delta W^{(m)}(i,j)|$ of the multi-order interaction $I^{(m)}(i,j)$ is proportional to $w^{(m)}/\sqrt{\binom{n-2}{m}}$.

Theorem 1 shows that the DNN is more likely to encode simple interactions (m is small) and complex interactions (m is large), but it is less likely to encode interactions of intermediate complexity (m approximates $0.5n$).

Based on the representation bottleneck, we further propose two losses to guide the learning of conceptual representation encoded in DNNs by learning interactions of specific orders. Specifically, we randomly sample a set of variables $S_2 \subseteq N$, and the network output $v(S_2)$ encodes interactions between input variables in S_2, where $v(S_2)$ is the output score when we keep variables in S_2 unchanged but mask variables in $N\setminus S_2$ by the baseline value. Then, we randomly sample a subset S_1 of the set S_2, i.e., $S_1 \subsetneq S_2$. Similarly, the network output $v(S_1)$ based on S_1 encodes interactions between input variables in S_1. Since S_1 is a subset of S_2, some interactions encoded in $v(S_1)$ and $v(S_2)$ overlap. We prove that using $v(S_2)$ to minus $\frac{|S_2|}{|S_1|}v(S_1)$ can cancel out these common interactions, thereby mainly maintaining interactions of some specific orders. Therefore, we accordingly define $\Delta u(r_1, r_2)$, as follows:

$$\Delta u(r_1,r_2) = \mathrm{E}_{S_1,S_2:\emptyset\subseteq S_1\subsetneq S_2\subseteq N}[v(S_2) - r_2/r_1 v(S_1)] \tag{22}$$

where $|S_1| = r_1 n$, $|S_2| = r_2 n$, and $0 \le r_1 < r_2 \le 1$. We prove that $\Delta u(r_1, r_2)$ mainly encodes interactions of $[0, r_2 n]$-th orders. In this way, we propose $L^+(r_1, r_2)$ and $L^-(r_1, r_2)$ losses, which encourage and penalize the DNN to use interactions of the orders within the range $[0, r_2 n]$ for inference, respectively.

$$\begin{aligned} L^+(r_1,r_2) = &-\frac{1}{|\Omega|}\sum_{x\in\Omega}\sum_{c=1}^{C}P(y^*=c|x)\times \\ &\log P(\hat{y}=c|\Delta u_c(r_1,r_2|x)) \\ L^-(r_1,r_2) = &\frac{1}{|\Omega|}\sum_{x\in\Omega}\sum_{c=1}^{C}P(\hat{y}=c|\Delta u_c(r_1,r_2|x))\times \\ &\log P(\hat{y}=c|\Delta u_c(r_1,r_2|x)). \end{aligned} \tag{23}$$

Here, given an input image x in the training set Ω, y^* and $\hat{y}$ denote the true label and the predicted label, respectively. $P(\hat{y} = c|\Delta u_c(r_1, r_2|x))$ denotes the probability of using $\Delta u_c(r_1, r_2|x)$ to classify x to the category c, where $\Delta u_c(r_1, r_2)$ follows the definition in (22) when $v(S)$ is set as the logit of the category c. As mentioned above, $\Delta u_c(r_1, r_2)$ represents interactions of $[0, r_2 n]$-th orders. In this way, $L^+(r_1, r_2)$ is computed as the cross entropy loss of the classification based on $\Delta u_c(r_1, r_2)$, which encourages the DNN to use interactions of $[0, r_2 n]$-th orders for inference. In contrast, $L^-(r_1, r_2)$ is set as the the minus entropy loss of the classification based on $\Delta u_c(r_1, r_2)$, which prevents the DNN from using interactions of $[0, r_2 n]$-th orders for inference.

Based on the above two losses, we train DNNs to mainly encode high-order interactions. We find that

DNNs, which mainly encode high-order interactions, are usually sensitive to the adversarial perturbation than normally-trained DNNs. These results verify that the adversarial attack mainly affects high-order interactions, which has been demonstrated in [22].

6 Conclusions

This paper mainly introduces our recent system of game-theoretic interactions, which unifies both the explanation for knowledge representations of a DNN and the explanation for the representation power of a DNN. We define the multi-order interaction and the multivariate interaction. Such interactions help us explain DNNs from novel perspectives, for example, quantifying knowledge concepts encoded by a DNN, extracting prototypical concepts, and explaining the representation bottleneck of DNNs. We can also use these interactions to explain and improve current explanation methods. For example, the interaction enables us to learn optimal baseline values for the Shapley value, and provides a unified perspective to compare fourteen different attribution methods. Finally, we prove that interactions encoded in a DNN directly determine the representation power of a DNN (e.g., generalization power, adversarial transferability, and adversarial robustness). Therefore, we can consider the game-theoretic interaction as a unified explanation, which successfully bridges the gap between "the explanation of knowledge concepts encoded in a DNN" and "the explanation of the representation capacity of a DNN."

Acknowledgements

This work was partially supported by National Nature Science Foundation of China (Nos. 62276165 and U19B2043), National Key R & D Program of China (No. 2021ZD0111602), Shanghai Natural Science Foundation, China (Nos. 21JC1403800 and 21ZR1434600).

Declarations of conflict of interest

The authors declared that they have no conflicts of interest to this work.

References

[1] A. Dosovitskiy, T. Brox. Inverting visual representations with convolutional networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, pp. 4829–4837, 2016. DOI: 10.1109/CVPR.2016.522.

[2] A. Mahendran, A. Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Boston, USA, pp. 5188–5196, 2015. DOI: 10.1109/CVPR.2015.7299155.

[3] K. Simonyan, A. Vedaldi, A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Proceedings of the 2nd International Conference on Learning Representations*, Banff, Canada, 2013. DOI: 10.48550/arXiv.1312.6034.

[4] M. T. Ribeiro, S. Singh, C. Guestrin. "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*, ACM, San Francisco, USA, pp. 1135–1144, 2016. DOI: 10.1145/2939672.2939778.

[5] S. M. Lundberg, S. I. Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 4768–4777, 2017.

[6] P. J. Kindermans, K. T. Schütt, M. Alber, K. R. Müller, D. Erhan, B. Kim, S. Dähne. Learning how to explain neural networks: Patternnet and patternattribution. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018. [Online], Available: https://dblp.org/rec/conf/iclr/KindermansSAMEK18.bib.

[7] S. Sabour, N. Frosst, G. E. Hinton. Dynamic routing between capsules. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 3859–3869, 2017.

[8] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, P. Abbeel. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, Barcelona, Spain, pp. 2180–2188, 2016.

[9] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, A. Lerchner. Beta-VAE: Learning basic visual concepts with a constrained variational framework. In *Proceedings of the 5th International Conference on Learning Representations*, Toulon, France, 2017.

[10] T. W. Weng, H. Zhang, P. Y. Chen, J. F. Yi, D. Su, Y. P. Gao, C. J. Hsieh, L. Daniel. Evaluating the robustness of neural networks: An extreme value theory approach. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018.

[11] S. Fort, P. K. Nowak, S. Jastrzebski, S. Narayanan. Stiffness: A new perspective on generalization in neural networks. [Online], Available: https://arxiv.org/abs/1901.09491, 2019.

[12] R. Novak, Y. Bahri, D. A. Abolafia, J. Pennington, J. Sohl-Dickstein. Sensitivity and generalization in neural networks: An empirical study. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018.

[13] H. Zhang, Y. C. Xie, L. J. Zheng, D. Zhang, Q. S. Zhang. Interpreting multivariate shapley interactions in dnns. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, vol. 35, pp. 10877–10886, 2021. DOI: 10.1609/aaai.v35i12.17299.

[14] H. Zhang, S. Li, Y. C. Ma, M. J. Li, Y. C. Xie, Q. S. Zhang. Interpreting and boosting dropout from a game-theoretic view. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[15] J. Ren, Z. P. Zhou, Q. R. Chen, Q. S. Zhang. Can we faithfully represent absence states to compute shapley values

on a DNN? [Online], Available: https://arxiv.org/abs/2105.10719, 2021.

[16] H. Q. Deng, N. Zou, M. N. Du, W. F. Chen, G. C. Feng, X. Hu. A unified taylor framework for revisiting attribution methods. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, vol. 35, pp. 11462–11469, 2021. DOI: 10.1609/aaai.v35i13.17365.

[17] J. Ren, M. J. Li, Q. R Chen, H. Q. Deng, Q. S. Zhang. Towards axiomatic, hierarchical, and symbolic explanation for deep models. [Online], Available: https://arxiv.org/abs/2111.06206, 2021.

[18] D. Zhang, H. Zhang, H. L. Zhou, X. Y. Bao, D. Huo, R. Z. Chen, X. Cheng, M. Y. Wu, Q. S. Zhang. Building interpretable interaction trees for deep NLP models. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, vol. 35, pp. 14328–14337, 2021. DOI: 10.1609/aaai.v35i16.17685.

[19] H. Q. Deng, Q. H. Ren, H. Zhang, J. Ren, Q. S. Zhang. Discovering and explaining the representation bottleneck of dnns. In *Proceedings of the 10th International Conference on Learning Representations*, 2021.

[20] X. Cheng, C. T. Chu, Y. Zheng, J. Ren, Q. S. Zhang. A game-theoretic taxonomy of visual concepts in DNNs. [Online], Available: https://arxiv.org/abs/2106.10938, 2021.

[21] X. Cheng, X. Wang, H. T. Xue, Z. Y. Liang, Q. S. Zhang. A hypothesis for the aesthetic appreciation in neural networks. [Online], Available: https://arxiv.org/abs/2108.02646, 2021.

[22] J. Ren, D. Zhang, Y. S. Wang, L. Chen, Z. P. Zhou, Y. T. Chen, X. Cheng, X. Wang, M. Zhou, J. Shi, Q. S. Zhang. A unified game-theoretic interpretation of adversarial robustness. [Online], Available: https://arxiv.org/abs/2111.03536, 2021.

[23] X. Wang, J. Ren, S. Y. Lin, Y. S. Wang, Q. S. Zhang. A unified approach to interpreting and boosting adversarial transferability. In *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[24] L. S. Shapley. A value for n-person games. *Contributions to the Theory of Games*, H. W. Kuhn, A. W. Tucker, Eds., Princeton, USA: Princeton University Press, pp. 307–317, 1953. DOI: 10.1515/9781400881970-018.

[25] J. C. Harsanyi. A simplified bargaining model for the n-person cooperative game. *International Economic Review*, vol. 4, no. 2, pp. 194–220, 1963. DOI: 10.2307/2525487.

[26] M. Grabisch, M. Roubens. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of Game Theory*, vol. 28, no. 4, pp. 547–565, 1999. DOI: 10.1007/s001820050125.

[27] P. Dabkowski, Y. Gal. Real time image saliency for black box classifiers. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 6970–6979, 2017.

[28] R. J. Weber. Probabilistic values for games. *The Shapley Value. Essays in Honor of Lloyd S. Shapley*, A. E. Roth, Ed., Cambridge, UK: Cambridge University Press, pp. 101–120, 1988. DOI: 10.1017/CBO9780511528446.008.

[29] M. Sundararajan, K. Dhamdhere, A. Agarwal. The shapley taylor interaction index. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 9259–9268, 2020.

[30] J. Castro, D. Gómez, J. Tejada. Polynomial calculation of the shapley value based on sampling. *Computers & Operations Research*, vol. 36, no. 5, pp. 1726–1730, 2009. DOI: 10.1016/j.cor.2008.04.004.

[31] H. Q. Deng, N. Zou, M. N. Du, W. F. Chen, G. C. Feng, X. Hu. A general taylor framework for unifying and revisiting attribution methods. [Online], Available: https://arxiv.org/abs/2105.13841, 2021.

[32] A. Shrikumar, P. Greenside, A. Shcherbina, A. Kundaje. Not just a black box: Learning important features through propagating activation differences. [Online], Available: https://arxiv.org/abs/1605.01713, 2016.

[33] S. Bach, A. Binder, G. Montavon, F. Klauschen, K. R. Müller, W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS One*, vol. 10, no. 7, Article number e0130140, 2015. DOI: 10.1371/journal.pone.0130140.

[34] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of IEEE International Conference on Computer Vision*, Venice, Italy, pp. 618–626, 2017. DOI: 10.1109/ICCV.2017.74.

[35] L. M. Zintgraf, T. S. Cohen, T. Adel, M. Welling. Visualizing deep neural network decisions: Prediction difference analysis. In *Proceedings of the 5th International Conference on Learning Representations*, Toulon, France, 2017.

[36] M. Sundararajan, A. Taly, Q. Q. Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia, pp. 3319–3328, 2017.

[37] A. Shrikumar, P. Greenside, A. Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia, pp. 3145–3153, 2017.

[38] G. Montavon, S. Lapuschkin, A. Binder, W. Samek, K. R. Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, vol. 65, pp. 211–222, 2017. DOI: 10.1016/j.patcog.2016.11.008.

[39] G. Erion, J. D. Janizek, P. Sturmfels, S. M. Lundberg, S. I. Lee. Improving performance of deep learning models with axiomatic attribution priors and expected gradients. *Nature Machine Intelligence*, vol. 3, no. 7, pp. 620–631, 2021. DOI: 10.1038/s42256-021-00343-w.

[40] M. D. Zeiler, R. Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the 13th European Conference on Computer Vision*, Springer, Zurich, Switzerland, pp. 818-833, 2014. DOI: 10.1007/978-3-319-10590-1_53.

[41] Q. S. Zhang, Y. N. Wu, S. C. Zhu. Interpretable convolutional neural networks. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 8827–8836, 2018. DOI: 10.1109/CVPR.2018.00920.

[42] D. Bau, B. L. Zhou, A. Khosla, A. Oliva, A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, pp. 3319–3327, 2017. DOI: 10.1109/CVPR.2017.354.

[43] C. H. Xie, Z. S. Zhang, Y. Y. Zhou, S. Bai, J. Y. Wang, Z.

Ren, A. L. Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Long Beach, USA, pp. 2725–2734, 2019. DOI: 10.1109/CVPR.2019.00284.

[44] Y. P. Dong, F. Z. Liao, T. Y. Pang, H. Su, J. Zhu, X. L. Hu, J. G. Li. Boosting adversarial attacks with momentum. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 9185–9193, 2018. DOI: 10.1109/CVPR.2018.00957.

[45] L. Wu, Z. X. Zhu, C. Tai, W. N. E. Understanding and enhancing the transferability of adversarial examples. [Online], Available: https://arxiv.org/abs/1802.09707, 2018.

[46] D. X. Wu, Y. S. Wang, S. T. Xia, J. Bailey, X. J. Ma. Skip connections matter: On the transferability of adversarial examples generated with resnets. In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020.

[47] Y. P. Dong, T. Y. Pang, H. Su, J. Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Long Beach, USA, pp. 4307–4316, 2019. DOI: 10.1109/CVPR.2019.00444.

[48] Y. W. Li, S. Bai, Y. Y. Zhou, C. H. Xie, Z. S. Zhang, A. Yuille. Learning transferable adversarial examples via ghost networks. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, New York, USA, vol. 34, pp. 11458–11465, 2020. DOI: 10.1609/aaai.v34i07.6810.

[49] N. Inkawhich, K. Liang, L. Carin, Y. R. Chen. Transferable perturbations of deep feature distributions. In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020.

[50] Q. Huang, I. Katsman, H. He, Z. Q. Gu, H. He, S. Belongie, S. N. Lim. Enhancing adversarial example transferability with an intermediate level attack. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Seoul, Republic of Korea, pp. 4732–4741, 2019. DOI: 10.1109/ICCV.2019.00483.

[51] L. L. Gao, Q. L. Zhang, J. K. Song, X. L. Liu, H. T. Shen. Patch-wise attack for fooling deep neural network. In *Proceedings of the 16th European Conference on Computer Vision*, Springer, Glasgow, UK, pp. 307–322, 2020. DOI: 10.1007/978-3-030-58604-1_19.

[52] Y. W. Guo, Q. Z. Li, H. Chen. Backpropagating linearly improves transferability of adversarial examples. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Vancouver, Canada, vol. 33, pp. 85–95, 2020.

[53] Y. Zhu, J. C. Sun, Z. G. Li. Rethinking adversarial transferability from a data distribution perspective. In *Proceedings of the 10th International Conference on Learning Representations*, 2022.

[54] Z. Y. Qin, Y. B. Fan, Y. Liu, L. Shen, Y. Zhang, J. Wang, B. Y. Wu. Boosting the transferability of adversarial attacks with reverse adversarial perturbation. [Online], Available: https://arxiv.org/abs/2210.05968, 2022. DOI: 10.48550/arXiv.2210.05968.

[55] Z. B. Wang, H. C. Guo, Z. F. Zhang, W. X. Liu, Z. Qin, K. Ren. Feature importance-aware transferable adversarial attacks. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Montreal, Canada, pp. 7619–7628, 2021. DOI: 10.1109/ICCV48922.2021.00754.

[56] J. M. Springer, M. Mitchell, G. T. Kenyon. Adversarial perturbations are not so weird: Entanglement of robust and non-robust features in neural network classifiers. [Online], Available: https://arxiv.org/abs/2102.05110, 2021.

[57] J. D. Lin, C. B. Song, K. He, L. W. Wang, J. E. Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020.

[58] A. Fawzi, O. Fawzi, P. Frossard. Analysis of classifiers′ robustness to adversarial perturbations. *Machine Learning*, vol. 107, no. 3, pp. 481–508, 2018. DOI: 10.1007/s10994-017-5663-3.

[59] A. Boopathy, S. J. Liu, G. Y. Zhang, C. Liu, P. Y. Chen, S. Y. Chang, L. Daniel. Proper network interpretability helps adversarial robustness in classification. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 1014–1023, 2020.

[60] A. Ignatiev, N. Narodytska, J. Marques-Silva. On relating explanations and adversarial examples. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Vancouver, Canada, pp. 15883–15893, 2019.

[61] P. Y. D. Yang, J. B. Chen, C. J. Hsieh, J. L. Wang, M. Jordan. ML-LOO: Detecting adversarial examples with feature attribution. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, New York, USA, vol. 34, pp. 6639–6647, 2020. DOI: 10.1609/aaai.v34i04.6140.

[62] T. DeVries, G. W. Taylor. Improved regularization of cs with cutout. [Online], Available: https://arxiv.org/abs/1708.04552, 2017.

[63] M. Jere, M. Kumar, F. Koushanfar. A singular value perspective on model robustness. [Online], Available: https://arxiv.org/abs/2012.03516, 2020.

[64] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu. Towards deep learning models resistant to adversarial attacks. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada, 2018.

[65] X. Bouthillier, K. Konda, P. Vincent, R. Memisevic. Dropout as data augmentation. [Online], Available: https://arxiv.org/abs/1506.08700, 2015.

[66] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

Huilin Zhou received the B. Sc. degree in mathematics from University of Electronic Science and Technology of China, China in 2019. She is currently a Ph. D. degree candidate in computer science and technology at Shanghai Jiao Tong University, China.

Her research interests include computer vision and machine learning.

E-mail: zhouhuilin116@sjtu.edu.cn
ORCID iD: 0000-0001-8834-4665

Jie Ren received the B. Sc. degree in computer science from Shanghai Jiao Tong University, China in 2020. She is currently a Ph. D. degree candidate in computer science and technology at Shanghai Jiao Tong University, China.

Her research interests include computer vision and machine learning.

E-mail: ariesrj@sjtu.edu.cn

Huiqi Deng received the B. Sc. degree in applied mathematics from Central China Normal University, China in 2015, the Ph. D. degree in applied mathematics from School of mathematics, Sun Yat-Sen (Zhongshan) University, China in 2020. She is a post-doctor at Shanghai Jiao Tong University, China.

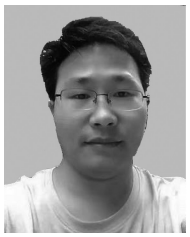
Her research interests include computer vision and machine learning.

E-mail: denghq7@sjtu.edu.cn

Xu Cheng received the B. Sc. degree in communication engineering at Xiamen University, China in 2018. She is currently a Ph. D. degree candidate in computer science and technology at Shanghai Jiao Tong University, China.

Her research interests include computer vision and machine learning.

E-mail: xcheng8@sjtu.edu.cn



Jinpeng Zhang received B. Sc. and M. Sc. degrees in material science & engineering from Huazhong University of Science & Technology (HUST), China in 2011 and 2014, respectively, the Ph. D. degree in pattern recognition and intelligent systems from National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences (NLPR, CASIA), China in 2019. He is with Intelligent Science & Technology Academy Limited of China Aerospace Science and Industry Corporation (CASIC), China.

His research interests include machine learning and computer vision.

E-mail: zhjphust@126.com

Quanshi Zhang received the Ph. D. degree from University of Tokyo, Japan in 2014. From 2014 to 2018, he was a postdoctoral researcher at University of California, USA. He is an associate professor at Shanghai Jiao Tong University, China. He won the ACM China Rising Star Award at ACM TURC 2021. He was the speaker of the tutorials on XAI at IJCAI 2020 and IJCAI 2021. He was the co-chairs of the workshops towards XAI in ICML 2021, AAAI 2019, and CVPR 2019. In particular, he has made influential research in explainable AI (XAI).

His research interests include machine learning and computer vision.

E-mail: zqs1022@sjtu.edu.cn (Corresponding author)
ORCID iD: 0000-0002-6108-2738

Machine Intelligence Research

Citation: H. Zhou, J. Ren, H. Deng, X. Cheng, J. Zhang, Q. Zhang. Interpretability of neural networks based on game–theoretic interactions. *Machine Inteligence Research*. https://doi.org/10.1007/s11633–023–1419–7

Articles may interest you