A FRAMEWORK FOR ANALYZING DNN REPRESENTATION QUALITY: DYNAMICS OF INTERACTION GENERALIZABILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper addresses the core challenge in the field of symbolic interaction[1], i.e., how to define, quantify, and track the dynamics of generalizable and non-generalizable interactions encoded by a DNN throughout the training process. Specifically, this work builds upon the recent theoretical achievement in explainable AI (Ren et al., 2024), which proves that the detailed inference patterns of DNNs can be strictly rewritten as a small number of AND-OR interaction patterns. Based on this, we propose an efficient method to quantify the generalizability of each interaction and discover distinct three-phase dynamics of the generalizability of interactions during training. In particular, the early phase of training typically removes noisy and non-generalizable interactions and learns simple and generalizable interactions. The second and the third phases tend to re-learn increasingly complex interactions that are harder to generalize. Experimental results show that the dynamics of eliminating and re-learning non-generalizable interactions are temporally aligned with the dynamics of the training-testing loss gap.

1 INTRODUCTION

Understanding the quality of representations learned by DNNs remains an important problem in AI. Extensive studies (Foret et al., 2021; Li et al., 2018; Xiao et al., 2020) have explored the generalization and robustness of DNNs from different perspectives.

In this paper, rather than developing a new method for improving performance, we take preliminary steps toward a fundamental scientific question: *Do DNNs with different architectures exhibit similar dynamics in the evolution of their representation quality during the training process, and can these dynamics be characterized in terms of symbolic patterns?*

Unlike prior works that treat the representation quality of a DNN as a monolithic whole, we pursue a new research perspective: we decompose the overall representation quality of a DNN into the generalizability of compositional inference patterns encoded by the DNN (see Figure 1). In this way, the overall representation quality of the model can be interpreted through changes in the generalizability of inference patterns.

Background. The emerging direction of interaction-based explanation (Liu et al., 2023; Ren et al., 2023b;c; Zhou et al., 2024) has provided an initial answer to the above question (see Section 4 for a survey).[1] These studies have found that a small set of interaction patterns can sufficiently and comprehensively represent the intricate patterns used by a DNN for inference.

> As shown in Figure 1, an interaction can be viewed as a pattern between input variables that makes a numerical contribution to the model's output. For example, an LLM can automatically learn phrase-like patterns over word sets such as $S = \{$*earth, revolves, sun, in, a*$\}$. Each pattern can be formulated as an interaction. The interaction w.r.t. S contributes a positive effect $I_T^{\text{and}} = 0.39$, increasing the confidence of generating the token "year." Similarly, for DNNs on image classification, an interaction over different patches usually represents a visual pattern.

[1]The theory system spans more than 20 papers; representative studies are summarized in Section 4.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

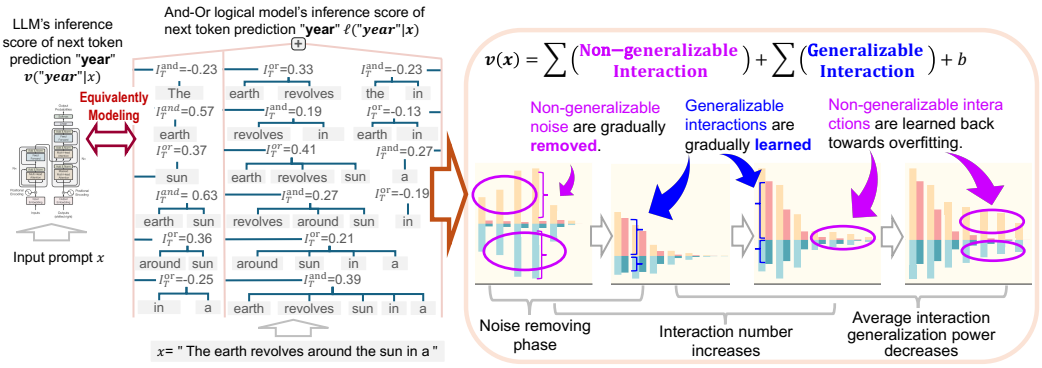

Figure 1: (Left) It has been proven that a small set of AND-OR interaction patterns is sufficient to mathematically represent all intricate inference patterns used by a DNN for inference. (Right) For each DNN, the representation quality can be explained by distinct three-phase dynamics of the generalizability of interactions encoded by it. **Additional AND-OR logical models that characterize predictions of the LLMs on different prompts are provided in Figure 7 in Appendix F.**

A variety of prior studies (Liu et al., 2023; Ren et al., 2023b;c; Zhou et al., 2024; Cheng et al., 2024) have established the faithfulness of interactions through theoretical analysis and empirical validation. Please see Section 2.1 for details.

Our research. As mentioned in the above theory, a DNN's prediction can be decomposed into the quantitative contributions of a compact collection of interaction patterns (see Figure 1). Therefore, we propose a method to quantify the generalizability of each interaction pattern. Extensive empirical experiments show that *changes in the generalizability of intrinsic interactions during training can well explain the changes of the overall representation quality of DNNs*.

For computational feasibility[2], we define the generalizability of each interaction pattern as the transferability of this interaction to another DNN trained for the same task. This definition is inspired by CCA-based analysis (Hotelling, 1936; Raghu et al., 2017; Morcos et al., 2018), which considers transferable features shared by various models as more reliable features. Therefore, we train a *baseline DNN* on held-out unseen samples[3]. Thus, interactions that can transfer to the baseline DNN are considered generalizable to these unseen samples. This is because the transferable interaction is also learned by the baseline DNN from such samples as a primitive inference pattern.

The quantification of generalizable interactions allows us to **uncover the distinctive three-phase dynamics of generalizability of interactions through the entire training process** (see Figure 3):

• Phase 1: Given a fully initialized DNN, in the early epochs of training, a large number of non-generalizable interactions are removed from the DNN, while a few generalizable interactions are gradually learned. These generalizable interactions are often simple and involve a small number of input variables. This may serve as supporting evidence that the DNN's generalization power increases substantially during the early stage of training.

• Phase 2: The DNN continues to learn more interactions, but the newly learned interactions exhibit increasing complexity (*i.e.*, learning interactions between more input variables). These more complex interactions often have poorer generalizability. Thus, although the testing loss of the DNN continues to decrease, it tends to saturate due to the decreasing generalizability of the newly learned interactions.

• Phase 3: The DNN continues to learn additional interactions, but these newly learned interactions are often complex and difficult to transfer to unseen data. Very few of these late-stage learned interactions represent simple, broadly generalizable patterns. Therefore, we can consider this phase

[2]Most existing definitions of generalizability are computationally intractable when used to evaluate the representational quality of a single inference pattern. For example, people cannot enumerate the occurrence of each specific interaction pattern (a phrase pattern) encoded by an LLM across massive input sentences.

[3]For implementation, considering the limited number of testing samples in datasets, the held-out samples are obtained by randomly splitting the original training set into two equal halves; the target DNN is trained on one half of the training samples. The other half of training samples can be regarded as unseen testing samples.

to be closely related to the overfitting of the DNN. During this phase, the training-testing loss gap continues to widen. Furthermore, the testing loss may even increase.

Besides the temporal alignment between interaction generalizability and the training-testing loss gap, the proposed interaction generalizability metric also exhibits significant correlations with other DNN performance metrics. (1) We find that subtracting the effects of non-generalizable interactions significantly reduces the training-testing loss gap. (2) Moreover, follow-up studies in Appendix J further show that penalizing non-generalizable interactions during training improves DNN performance.

2 METHODOLOGY

2.1 PRELIMINARIES: AND-OR INTERACTIONS

For a sample $\mathbf{x} = [x_1, \ldots, x_n]^T$, we index its n input variables by $N = \{1, \ldots, n\}$. We write the scalar prediction output of the DNN as $v(\mathbf{x}) \in \mathbb{R}$. Although different choices of the prediction output $v(\mathbf{x})$ are possible,[4] we adopt the standard formulation widely used in (Li & Zhang, 2023b; Liu et al., 2023) to let $v(\mathbf{x})$ be the classification confidence score for the ground-truth label $\mathbf{y}^*$:

$$v(\mathbf{x}) = f(p(\mathbf{y}^* \mid \mathbf{x})) = \log \frac{p(\mathbf{y}^* \mid \mathbf{x})}{1 - p(\mathbf{y}^* \mid \mathbf{x})}. \tag{1}$$

Universal matching property of interactions guarantees the fidelity of the explanation. Chen et al. (2024) have trained a logical model $\ell(\cdot)$ that encodes a set of AND-OR interaction patterns, and **they have theoretically proven that the DNN's outputs $v(\cdot)$ over the 2^n masked[5] variants of the input $\mathbf{x}$ can be closely replicated using the below logical model $\ell(\cdot)$.** *Please see a **video demo** for symbolic interaction in the supplementary material. Furthermore, please see Figure 7 in Appendix F for more logical models that explain LLMs.*

$$\forall \mathbf{x}' \in \mathcal{Q}, \quad |\ell(\mathbf{x}') - v(\mathbf{x}')| < \epsilon,$$
$$\ell(\mathbf{x}') = \sum_{T \in \Omega_{\text{and}}} \underbrace{I_T^{\text{and}} \cdot \delta_{\text{and}}\Big(\substack{\mathbf{x}' \text{ triggers AND relation} \\ \text{between input variables in } T}\Big)}_{\text{an AND interaction between input variables in } T} + \sum_{T \in \Omega_{\text{or}}} \underbrace{I_T^{\text{or}} \cdot \delta_{\text{or}}\Big(\substack{\mathbf{x}' \text{ triggers OR relation} \\ \text{between input variables in } T}\Big)}_{\text{an OR interaction between input variables in } T} + b, \tag{2}$$

where the small scalar ϵ guarantees that the logical model $\ell(\cdot)$ provides a close approximation to the DNN output function $v(\cdot)$. $\mathcal{Q} = \{\mathbf{x}_S \mid S \subseteq N\}$ represents the collection of 2^n masked[5] variants, in which $\mathbf{x}_S$ denotes a sample obtained by masking the input variables belonging to $N \setminus S$. The scalar weights I_T^{and}, I_T^{or} and b are estimated using the procedure of Chen et al. (2024). We use Ω_{and} to collect the AND interaction patterns and Ω_{or} to collect the OR interaction patterns. Please refer to Appendix A for the details of ϵ and the justification of the fidelity of using the logical model $\ell(\cdot)$ to explain the DNN output function $v(\cdot)$.

Each AND interaction is characterized by the binary function $\delta_{\text{and}}\Big(\substack{\mathbf{x}' \text{ triggers AND relation} \\ \text{between input variables in } T}\Big) \in \{0, 1\}$. The function takes the value 1 when none of the variables in T are masked in $\mathbf{x}'$ and takes 0 otherwise. Likewise, *each OR interaction* is characterized by the binary function $\delta_{\text{or}}\Big(\substack{\mathbf{x}' \text{ triggers OR relation} \\ \text{between input variables in } T}\Big) \in \{0, 1\}$. It evaluates to 1 when any variable in T remains not masked in $\mathbf{x}'$ and to 0 otherwise.

> For example, in Figure 1, given an input prompt *"the earth revolves around the sun in a"*, the LLM leverages automatically learned phrases to compute the inference score of the target token *year*, even though these phrases may not necessarily align with human intuition. Each phrase pattern can be mathematically formulated as an interaction. *E.g.*, the phrase comprising *"earth"* and *"sun"* constitutes an AND interaction, which boosts the inference score of *"year"* by 0.63. In contrast,

[4] E.g., $v(\mathbf{x})$ can be set to the feature dimension associated with the ground-truth category before the softmax operation.

[5] To mask an input variable, we replace it by a baseline value. For continuous input variables, this reference is their empirical mean across samples. For NLP models, tokens are suppressed using the reference embedding adopted in (Cheng et al., 2024; Ren et al., 2024).

> OR interactions capture a different type of phrase pattern. *E.g.*, either *"earth"* or *"revolves"* alone provides a weak indication of astronomy, and the OR interaction between them contributes an additional 0.33 numerical effect to the score for *"year"*.

> For image classification, the interactions between different image regions typically represent visual patterns captured by the DNN. In bird species classification, given an image of a flamingo, the DNN leverages an automatically learned AND interaction between visual patterns $S = \{\textit{red features}, \textit{long beak}\}$. This AND interaction increases the confidence in predicting the *"Flamingo"* class.

Sparsity of interactions. Notably, Ren et al. (2024) have demonstrated that DNNs typically[6] encode only a small number of AND-OR interactions. The number of interactions is constrained by theory and typically ranges from 50 to 150 interactions per input sample in practice. Please see Appendix C for more details. *Details of the fidelity of $\ell(\cdot)$ and the sparsity of interactions are provided in Appendix A and Appendix C, respectively.*

Regarding AND-OR interactions as basic units of inference. The interactions constituting $\ell(\cdot)$ are expressive enough to accurately approximate $v(\cdot)$ over the exponentially sized space of 2^n variants of the input $\mathbf{x}$. Consequently, these interactions are widely used (Li & Zhang, 2023b; Ren et al., 2024; Zhou et al., 2024) as basic units to explain the inference logic of DNNs.

Efficient computation and industrial applications of interactions. The computational cost of extracting interactions is high. To address this, many studies (Kang et al., 2024; 2025; Butler et al., 2025) have proposed different methods to speed up the computation. For example, Kang et al. (2024; 2025) have proposed efficient methods to approximate interaction-based explanations. Li & Zhang (2023b); Ren et al. (2023b) have have developed various methods to lower the computational overhead of computing interactions. Please see Appendix B for technical details of extracting interactions.

All these achievements have enabled interactions to be widely used to diagnose DNNs for various industrial applications (e.g., SymTrust AI (2026)). Please see Appendix B for more details. Moreover, in this paper, our analysis primarily requires the distribution of interactions rather than enumerating all tokens or regions. Therefore, it is sufficient to follow Ren et al. (2023b) to compute interactions on a small set of important input variables with high attribution scores in practice.

2.2 Quantifying the generalizability of interactions

2.2.1 Definition and quantification

The theoretical system built upon the universal matching property and the sparsity property ensures that a DNN's output can be faithfully decomposed into a linear combination of AND-OR interaction effects (see Figure 1). Therefore, the interaction generalizability provides an interpretable metric for analyzing the DNN's generalization power.

Previous definition of the generalizability of interactions. We first revisit the definition of interaction generalizability in (Zhou et al., 2024). Given an AND interaction S extracted from input $\mathbf{x}$ with $S \in \Omega_{\text{and}}$, we consider S generalizable if it frequently occurs in unseen samples and consistently affects the classification of a category. This criterion extends analogously to OR interactions.

Efficient quantification of the generalizability of interactions. However, the above definition does not provide an efficient way to quantify the generalizability of interactions. For example, in natural language processing tasks, it would require an exhaustive search for specific interactions across many unseen samples, leading to a prohibitive computational cost.

Therefore, we propose an approximate yet efficient method for quantifying the generalizability of each individual interaction. We evaluate whether individual interactions encoded by the DNN v can transfer to a baseline DNN v^{base} that is trained on held-out unseen samples. Because all interactions in v^{base} are learned from held-out unseen samples, all interactions that are transferable to v^{base} can be viewed as primitive inference patterns present in those held-out unseen samples[3]. This definition is inspired by CCA-based analysis (Hotelling, 1936; Raghu et al., 2017; Morcos et al., 2018), which

[6] Ren et al. (2024) have demonstrated that the sparsity of interactions can be ensured through three common conditions for smooth inferences in different DNNs on randomly masked samples. Please refer to Appendix C for more details.

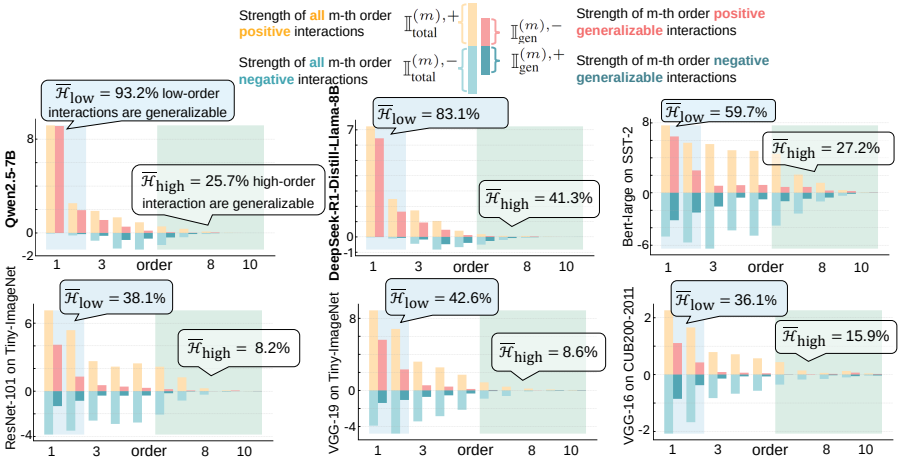

Figure 2: Comparison of generalizability between low-order interactions and high-order interactions. High-order interactions generally exhibit weaker generalizability than low-order interactions. Considering up to 10th order, we report the ratio of generalizable interactions of low orders as $\overline{\mathcal{H}}_{\text{low}} = \sum_{m=1}^{2}(|\mathbb{I}_{\text{gen}}^{(m),+}| + |\mathbb{I}_{\text{gen}}^{(m),-}|)/\sum_{m=1}^{2}(|\mathbb{I}_{\text{total}}^{(m),+}| + |\mathbb{I}_{\text{total}}^{(m),-}|)$ and the ratio of generalizable interactions of high orders as $\overline{\mathcal{H}}_{\text{high}} = \sum_{m=6}^{10}(|\mathbb{I}_{\text{gen}}^{(m),+}| + |\mathbb{I}_{\text{gen}}^{(m),-}|)/\sum_{m=6}^{10}(|\mathbb{I}_{\text{total}}^{(m),+}| + |\mathbb{I}_{\text{total}}^{(m),-}|)$.

considers inference patterns shared by various models to be more reliable. Please see Appendix D for the ablation study and Appendix N for justification.

To elucidate this approach, let us consider an AND interaction S encoded[7] by DNN v (*i.e.*, $S \in \Omega_{\text{and}}$). To this end, the interaction S is deemed to represent a generalizable pattern within the held-out data[3] if (1) the baseline DNN v^{base} concurrently encodes the AND interaction S (*i.e.*, $S \in \Omega_{\text{and}}^{v^{\text{base}}}$) and (2) the AND interaction S exerts a consistent directional influence on the classification of $\mathbf{x}$ (*i.e.*, $I_{S,v^{\text{base}}}^{\text{and}} \cdot I_S^{\text{and}} > 0$). The same principle applies to OR interactions. Accordingly, the generalizability of an AND/OR interaction is assessed by the following metrics:

$$\chi_{S,v^{\text{base}}}^{\text{and}} = \mathbf{1}(S \in \Omega_{\text{and}}^{v^{\text{base}}}) \cdot \mathbf{1}(I_S^{\text{and}} \cdot I_{S,v^{\text{base}}}^{\text{and}} > 0), \quad \chi_{S,v^{\text{base}}}^{\text{or}} = \mathbf{1}(S \in \Omega_{\text{or}}^{v^{\text{base}}}) \cdot \mathbf{1}(I_S^{\text{or}} \cdot I_{S,v^{\text{base}}}^{\text{or}} > 0) \tag{3}$$

where $\chi_{S,v^{\text{base}}}^{\text{and}}, \chi_{S,v^{\text{base}}}^{\text{or}} \in \{0, 1\}$, and $\mathbf{1}(\cdot)$ denotes the binary indicator function, which returns 1 if the condition is satisfied, and returns 0 otherwise.

2.2.2 Verifying previous findings on interaction generalizability

The metrics $\chi_{S,v^{\text{base}}}^{\text{and}}$ and $\chi_{S,v^{\text{base}}}^{\text{or}}$ allow us to rigorously verify the empirical finding in symbolic interaction[1] (Zhou et al., 2024): low-order interactions generally exhibit stronger generalizability compared to high-order interactions, which was observed by Zhou et al. (2024) in experiments.[8] Specifically, for an interaction S, its order means the cardinality of the set of input variables involved in this interaction, *i.e.*, $\text{Ord}_S = |S|$. Although the claim that "high-order interactions generalize poorly" has been widely cited by Chen et al. (2024); Cheng et al. (2024), it still requires verification at the level of individual interactions.

Specifically, we introduce four aggregate metrics over the extracted interactions[7] to assess the generalizability of interactions for each order m: (1) $\mathbb{I}_{\text{total}}^{(m),+}$ sums the positive effects of all m-th-order interactions; (2) $\mathbb{I}_{\text{total}}^{(m),-}$ sums the negative effects of all m-th-order interactions; (3) $\mathbb{I}_{\text{gen}}^{(m),+}$ sums the positive effects of generalizable interactions retained by the metric χ_S^{type}; and (4) $\mathbb{I}_{\text{gen}}^{(m),-}$ analogously sums the retained negative effects of generalizable interactions. Formally, we have

[7] Please refer to Appendix B for the technical details and pseudocode of computation of interactions.

[8] Evidence from two experiments substantiates this claim: (1) It is observed that when more label noise is introduced, overfitted DNNs tend to model more high-order interactions compared to standard DNNs trained on clean data. (2) It is demonstrated that high-order interactions typically show worse robustness to input noise.

$$\mathbb{I}^{(m),+}_{\text{total}} = \sum_{\substack{\text{type}\in\\ \{\text{and, or}\}}} \sum_{\substack{S\in\Omega_{\text{type}}:\\ |S|=m}} \max(I^{\text{type}}_S, 0), \quad \mathbb{I}^{(m),-}_{\text{total}} = \sum_{\substack{\text{type}\in\\ \{\text{and, or}\}}} \sum_{\substack{S\in\Omega_{\text{type}}:\\ |S|=m}} \min(I^{\text{type}}_S, 0),$$
$$\mathbb{I}^{(m),+}_{\text{gen}} = \sum_{\substack{\text{type}\in\\ \{\text{and, or}\}}} \sum_{\substack{S\in\Omega_{\text{type}}:\\ |S|=m}} \max(I^{\text{type}}_S \cdot \chi^{\text{type}}_S, 0), \quad \mathbb{I}^{(m),-}_{\text{gen}} = \sum_{\substack{\text{type}\in\\ \{\text{and, or}\}}} \sum_{\substack{S\in\Omega_{\text{type}}:\\ |S|=m}} \min(I^{\text{type}}_S \cdot \chi^{\text{type}}_S, 0). \tag{4}$$

To this end, we conducted experiments to analyze the strength of generalizable and non-generalizable interactions in comparison across different orders. We trained VGG-19 and ResNet-101 on the Tiny-ImageNet dataset, and VGG-16 on the CUB-200-2011 dataset. We trained Bert-large on the SST-2 dataset. Please refer to Appendix E for experimental details. Additionally, we evaluated the Qwen2.5-7B model and the DeepSeek-R1-Distill-Llama-8B model on the SQuAD dataset; please refer to Appendix O for more details. Figure 2 shows that high-order interactions consistently exhibit lower generalizability than low-order interactions across different cases.

3 EXPERIMENTS

3.1 THREE-PHASE DYNAMICS OF THE GENERALIZABILITY OF INTERACTIONS

In this section, we show that the evolution of the distributions of all interactions and generalizable interactions (see Equation (4)) can effectively explain changes in the representational quality of the DNN during normal training and overfitting. Specifically, we define the following two metrics to better characterize the dynamics of interaction generalizability.

(1) Metric 1: The metric $\overline{\mathcal{H}}$ represents the average generalizability of the interactions.

$$\overline{\mathcal{H}} = \sum_m \left(|\mathbb{I}^{(m),+}_{\text{gen}}| + |\mathbb{I}^{(m),-}_{\text{gen}}| \right) / \sum_m \left(|\mathbb{I}^{(m),+}_{\text{total}}| + |\mathbb{I}^{(m),-}_{\text{total}}| \right). \tag{5}$$

(2) Metric 2: The number of interactions $\overline{\mathcal{N}}$ is measured as the number of top-ranked interactions that account for 80.0% of the total interaction strength. Specifically, $\overline{\mathcal{N}}$ is defined as

$$\overline{\mathcal{N}} = \overline{\mathcal{N}}_{\text{and}} + \overline{\mathcal{N}}_{\text{or}}, \quad \text{with}$$
$$\overline{\mathcal{N}}_{\text{and}} = \min_{\Omega\subseteq 2^N} |\Omega| \ \text{ s.t. } \sum_{S\in\Omega} |I^{\text{and}}_S| \geq 0.8 \cdot \sum_{S\subseteq N} |I^{\text{and}}_S|, \ \ \overline{\mathcal{N}}_{\text{or}} = \min_{\Omega\subseteq 2^N} |\Omega| \ \text{ s.t. } \sum_{S\in\Omega} |I^{\text{or}}_S| \geq 0.8 \cdot \sum_{S\subseteq N} |I^{\text{or}}_S|, \tag{6}$$

where $2^N = \{S \subseteq N\}$, and $\overline{\mathcal{N}}_{\text{and}}$ denotes the number of AND interactions, $\overline{\mathcal{N}}_{\text{or}}$ denotes the number of OR interactions.

Three-phase dynamics. Figure 3 shows that for most DNNs, the distribution of interactions follows three-phase dynamics in terms of the generalizability $\overline{\mathcal{H}}$ and the interaction number $\overline{\mathcal{N}}$:

Finding 1: *During the training of all DNNs, the number of interactions always exhibits* ***a regularity of first decreasing and then increasing****.*

Finding 2: *The average generalizability of interactions always demonstrates* ***a regularity of first increasing and then decreasing****.*

Finding 3: *The three-phase dynamics of interactions are temporally* ***aligned with and well explain the curve of the training-testing loss gap****. Detailed dynamics are as follows.*

• **Before Training**: Given a DNN with fully initialized parameters, although interactions extracted from the initialized DNN represent random noise patterns, statistically, the interactions exhibit a specific spindle-shaped distribution (see Figure 3 at timepoint A). (1) The model primarily encodes middle-order interactions, while very high-order and very low-order interactions are rarely modeled. (2) Furthermore, the effects of the positive interactions and the negative interactions often tend to cancel out. This phenomenon has been demonstrated in the previous study (Cheng et al., 2025).

At this stage, we observe that most interactions cannot transfer to the baseline DNN. This corroborates that interactions encoded in an initialized DNN are noise patterns.

• **Phase 1**: *In the early epochs of training, the average generalizability of interactions $\overline{\mathcal{H}}$ usually increases, and the number of interactions $\overline{\mathcal{N}}$ usually decreases* (see Figure 3 from timepoint A to

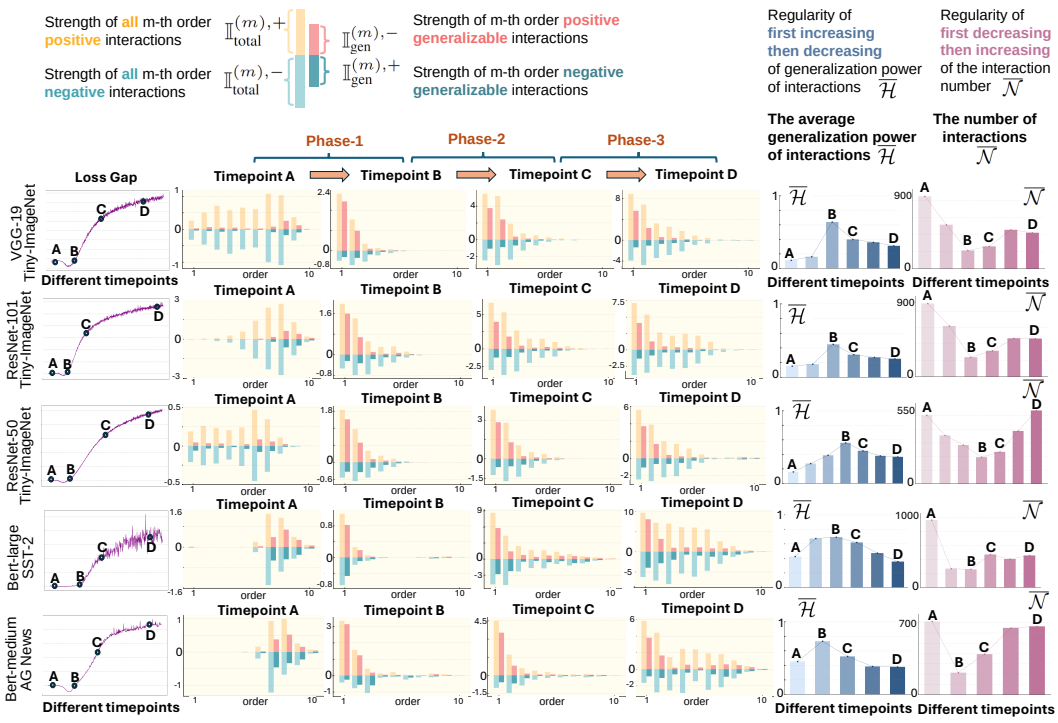

Figure 3: We visualize the change of interactions through the entire training process and track the change of the average generalizability of interactions $\overline{\mathcal{H}}$ and the number of interactions $\overline{\mathcal{N}}$. The dynamics of quantity and the average generalizability of interactions explain the change of the loss gap. Please see Appendix E for the training loss and the testing loss. Additionally, due to the high computational expense of training LLMs from scratch, Appendix P focuses on comparing interactions in an LLM before and after SFT; the results show that the SFT slightly removes some high-order interactions and boosts the generalizability of interactions.

timepoint B). The training primarily eliminates noise in the spindle-shaped distribution of interactions while simultaneously learning a few low-order interactions. During this phase, the DNN mainly learns these low-order interactions, which tend to generalize better to the baseline DNN as they capture simple, meaningful patterns. Meanwhile, high-order interactions initially modeled by the DNN are gradually eliminated.

According to Equation (2) in Section 2.1, the DNN's output function $v(\cdot)$ can be well predicted by the logical model $\ell(\cdot)$. Therefore, the classification confidence depends jointly on the number of interactions and the average generalizability of these interactions. Thus, the emergence of generalizable low-order interactions and the elimination of non-generalizable high-order interactions characterize the healthiest stage of training.

• **Phase 2**: *As training progresses, the average generalizability of interactions $\overline{\mathcal{H}}$ gradually decreases, and the number of interactions $\overline{\mathcal{N}}$ usually increases* (see Figure 3 from timepoint B to timepoint C). Both generalizable (mainly low-order) interactions and non-generalizable (mainly mid-to-high-order) interactions are learned by the DNN simultaneously. Since the initial noise interactions have been removed after the first phase, in the second phase the DNN progressively learns interactions of increasing order. Notably, we find that high-order interactions are generally less likely to be generalized to unseen data than lower-order interactions according to Section 2.2.2. Thus, two phenomena co-occur in the second phase: (1) the DNN continues to learn additional interactions; (2) the generalizability of newly learned interactions steadily decreases.

Consequently, the second phase faces a dilemma. Although the continuous learning of new interactions provides meaningful features for classification, the diminishing generalizability of these interactions introduces a risk of overfitting. This observation is further corroborated by the training curves: The rate of decrease of the training loss outpaces that of the testing loss, resulting in a

widening gap that indicates the potential risk of model overfitting to the training data. This phenomenon may arise because non-generalizable interactions contribute exclusively to the classification of training samples and fail to aid in the classification of samples unseen by the target DNN.

• **Phase 3**: *During the final training stage, DNNs primarily acquire non-generalizable (mainly mid-to-high-order) interactions, while learning very few generalizable (predominantly low-order) interactions* (see Figure 3 after timepoint C). This is because low-order interactions have already been largely captured during the preceding two phases. Meanwhile, these emergent high-order interactions typically demonstrate poor generalization to unseen data.

The dynamics of generalizable/non-generalizable interactions during training provide a concrete lens for understanding the overfitting process of a DNN. The low generalizability of the newly learned high-order interactions in Phases 2 and 3 indicates that these high-order interactions serve only the training samples. In contrast, in Phase 1, the DNN mainly learns low-order generalizable interactions while removing non-generalizable high-order interactions. Thus, interactions provide a decomposable metric for interpreting DNN generalization.

Additionally, a quantitative method for dividing the three phases is introduced in Appendix H.

3.2 Influence of non-generalizable interactions on the classification loss

Beyond the three-phase dynamics of interactions' generalizability, this subsection examines the correlation between non-generalizable interactions and DNN overfitting. The basic idea is implemented by (1) subtracting the effects of non-generalizable interactions from the prediction score of the DNN and (2) testing whether this subtraction reduces the training-testing loss gap.

Specifically, for each training or test sample $\mathbf{x}$, we define non-generalizable interactions, according to Equation (3), as those that do not transfer to the baseline DNN trained on the held-out unseen samples[3] (there is no overlap between training and testing sets). Therefore, our task is to examine whether subtracting the effects of the non-generalizable interactions identified using such third-party data can reduce the training-testing loss gap.

According to Equation (2) in Section 2.1, the DNN's output function $v(\cdot)$ can be well predicted by the logical model $\ell(\cdot)$. Thus we rewrite the scalar output of the DNN in Equation (2) as follows:

$$\begin{aligned} v(\mathbf{x}) \approx \ell(\mathbf{x}), \text{ with } \ell(\mathbf{x}) = & \sum_{\text{type}\in\{\text{and, or}\}} \sum_{S\in\Omega_{\text{type}}} I_S^{\text{type}} \cdot \chi_S^{\text{type}} \cdot \delta_{\text{type}}\big(\substack{\mathbf{x} \text{ triggers type relation} \\ \text{between input variables in } S}\big) \quad \textit{*generalizable interactions} \\ & + \sum_{\text{type}\in\{\text{and, or}\}} \sum_{S\in\Omega_{\text{type}}} I_S^{\text{type}} \cdot (1-\chi_S^{\text{type}}) \cdot \delta_{\text{type}}\big(\substack{\mathbf{x} \text{ triggers type relation} \\ \text{between input variables in } S}\big) + b \quad \textit{*non-generalizable interactions.} \end{aligned} \tag{7}$$

In this way, we can obtain the residual prediction score $v'(\mathbf{x})$ after the effects of all non-generalizable interactions have been subtracted. The predicted probability assigned to the correct class $\mathbf{y}^*$ after subtracting the effects of non-generalizable interactions can be derived from Equation (1) as follows:

$$p_{\text{sub}}(\mathbf{y}^*|\mathbf{x}) = f^{-1}\big(v'(\mathbf{x})\big) \quad \text{s.t.} \quad v'(\mathbf{x}) = v(\mathbf{x}) - \sum\nolimits_{\text{type}\in\{\text{and,or}\}} \sum\nolimits_{S\in\Omega_{\text{type}}} I_S^{\text{type}}\big(1-\chi_S^{\text{type}}\big) \tag{8}$$

Using the revised probability $p_{\text{sub}}(\mathbf{y}^*|\mathbf{x})$, we can compute the revised cross-entropy loss after subtracting the effects of non-generalizable interactions as $L_{\text{sub}} = -\sum_{\mathbf{x}} \log p_{\text{sub}}(\mathbf{y}^*|\mathbf{x})$.

We conducted experiments to train VGG-19, ResNet-50, and ResNet-101 on the Tiny-ImageNet dataset, Bert-large and Bert-medium on the SST-2 dataset, and Bert-large on the AG News dataset. We computed the original cross-entropy loss using the original classification probability $p(\mathbf{y}^*|\mathbf{x})$, as well as a modified cross-entropy loss using the revised probability $p_{\text{sub}}(\mathbf{y}^*|\mathbf{x})$. Please see Appendix I for more details. As shown in Figure 4, subtracting the effects of non-generalizable interactions significantly reduced the training-testing loss gap. This subtraction primarily increased the training loss while significantly decreasing, or only marginally affecting, the testing loss. These results suggested that the non-generalizable interactions whose effects were subtracted were closely correlated with DNN overfitting, as they helped fit training samples without generalizing to unseen data.

4 Related Work: Theory System of Symbolic Interaction

Symbolic explanation: a seemingly impossible task. Studies (Deng et al., 2022; Li & Zhang, 2023b; Ren et al., 2023b) show DNN inference can be reformulated via symbolic interactions between inputs.



Figure 4: Change in cross-entropy loss after subtracting the effects of non-generalizable interactions from the network output $v'(\mathbf{x})$. Subtracting these effects significantly increases the training loss, which restores the training loss that was inappropriately depressed by overfitted representations. In comparison, the testing loss either decreases significantly or changes only marginally.

Though counterintuitive, Ren et al. (2023a) proved a logical model based on such interactions can approximate DNN outputs on masked inputs. Further, Ren et al. (2024) found DNNs encode only a limited set of interactions under common conditions[6]. Complementing this, Chen et al. (2024) developed a method to extract generalizable interaction patterns consistent across models. These works form a mathematical basis for interpreting DNNs via variable interactions.

Characterizing DNN representational capacity. Interaction theory also serves as a tool to analyze DNN representation quality. Ren et al. (2021) showed adversarial attacks target complex interactions, with robustness decaying exponentially as interaction complexity grows. Ren et al. (2023c) proved Bayesian networks (Pearl, 1985) inherently struggle to model complex interactions. Deng et al. (2022) identified that DNNs capture very simple or highly complex interactions but falter with intermediate ones. Zhang et al. (2021) found dropout improves generalization by modulating interactions. Zhou et al. (2024) verified simpler interactions generalize better. Liu et al. (2023) gave theoretical evidence that DNNs find complex interactions harder to learn.

However, regarding the core issue of the generalizability of interactions, previous studies in symbolic interaction all failed to quantify the generalizability of a specific interaction. To this end, we propose to use a baseline DNN to identify generalizable interactions, which breaks through the computational efficiency bottleneck. In this way, we discover the distinctive three-phase dynamics in how interaction generalizability evolves during training. Our discovery shows that the interaction can be used as a decomposable metric for interpreting DNN generalization.

5 Conclusion

In this paper, we focus on the core issues in the field of symbolic interaction, *i.e.*, how to quantify the generalizability of each individual interaction in a DNN and how to utilize the evolution of the generalizability of interactions to diagnose the training process of a DNN. Specifically, we propose an efficient method to quantify the generalizability of each individual interaction learned by the DNN. Our analysis reveals the three-phase dynamics in the generalizability of interactions, which align well with the change of the DNN's training-testing loss gap. In the beginning, most non-generalizable interactions are eliminated, and the network primarily models simple, generalizable interactions. In the intermediate phase, the network begins to capture increasingly less generalizable interactions, and by the final phase, it predominantly learns entirely non-generalizable ones. Our experiments demonstrate that the dynamics of eliminating and re-learning non-generalizable interactions are temporally aligned with the dynamics of the training-testing loss gap.

Practical value. Our study suggests that subtracting the effects of non-generalizable interactions may improve the generalization power of a DNN. Follow-up studies in Appendix J have also shown that penalizing the strength of non-generalizable interactions $|I_S^{\text{NoneG}}|$ jointly with the minimization of the classification loss during the training process improves the performance of the DNN.

AI USE STATEMENT

AI Use Disclosure. In this paper, generative AI was employed only at the editing stage to refine sentence-level expression and improve the manuscript's clarity and fluency. No AI use contributed to the rest of the common research subtasks, such as generating synthetic data sets, cleaning and reformatting dataset, or helping develop theoretical models or conceptual frameworks or supporting qualitative and thematic data analysis. The authors independently conducted and validated the research and assessed each machine-assisted revision before incorporating it into the manuscript. The submitted manuscript remains entirely the responsibility of the authors, including any content developed with generative AI assistance.

REFERENCES

Landon Butler, Abhineet Agarwal, Justin Kang, Yigit Efe Erginbas, Bin Yu, and Kannan Ramchandran. Proxyspex: Inference-efficient interpretability via sparse feature interactions in llms. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 80441–80475. Curran Associates, Inc., 2025. doi: 10.52202/085713-2426. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/68dced1638d71d4d21598f79ae91262e-Paper-Conference.pdf.

Hua Cai, Shuang Zhao, Liang Zhang, Xuli Shen, Qing Xu, Weilin Shen, Zihao Wen, and Tianke Ban. Unilaw-R1: A large language model for legal reasoning with reinforcement learning and iterative inference. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 18117–18131, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.915. URL https://aclanthology.org/2025.emnlp-main.915/.

Lu Chen, Siyu Lou, Benhao Huang, and Quanshi Zhang. Defining and extracting generalizable interaction primitives from DNNs. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=OCqyFVFNeF.

Lei Cheng, Junpeng Zhang, Qihan Ren, and Quanshi Zhang. Revisiting Generalization Power of a DNN in Terms of Symbolic Interactions. *arXiv e-prints*, art. arXiv:2502.10162, February 2025. doi: 10.48550/arXiv.2502.10162.

Xu Cheng, Lei Cheng, Zhaoran Peng, Yang Xu, Tian Han, and Quanshi Zhang. Layerwise change of knowledge in neural networks. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=7zEoinErzQ.

Huiqi Deng, Qihan Ren, Hao Zhang, and Quanshi Zhang. Discovering and explaining the representation bottleneck of dnns. *International Conference on Learning Representations*, 2022.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=6Tm1mposlrM.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, and Zhibin Gou,

et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645 (8081):633—-638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016. URL https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html.

Harold Hotelling. *Relations Between Two Sets of Variates*, volume 28, pp. 321–377. [Oxford University Press, Biometrika Trust], 1936. URL http://www.jstor.org/stable/2333955.

Justin Singh Kang, Yigit Efe Erginbas, Landon Butler, Ramtin Pedarsani, and Kannan Ramchandran. Learning to understand: Identifying interactions via the möbius transform. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 46160–46202. Curran Associates, Inc., 2024. doi: 10.52202/079017-1467. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/520b379123d16e41f85472e766846486-Paper-Conference.pdf.

Justin Singh Kang, Landon Butler, Abhineet Agarwal, Yigit Efe Erginbas, Ramtin Pedarsani, Bin Yu, and Kannan Ramchandran. SPEX: Scaling feature interaction explanations for LLMs. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 28878–28903. PMLR, 2025. URL https://proceedings.mlr.press/v267/kang25a.html.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018.

Mingjie Li and Quanshi Zhang. Defining and quantifying and-or interactions for faithful and concise explanation of dnns. *arXiv preprint arXiv:2304.13312*, 2023a.

Mingjie Li and Quanshi Zhang. Does a neural network really encode symbolic concepts? In *International conference on machine learning*, pp. 20452–20469. PMLR, 2023b.

Dongrui Liu, Huiqi Deng, Xu Cheng, Qihan Ren, Kangrui Wang, and Quanshi Zhang. Towards the difficulty for a deep neural network to learn concepts of different complexities. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=mZ3hnyL9bS.

Ari Morcos, Maithra Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/a7a3d70c6d17a73140918996d03c014f-Paper.pdf.

J. Pearl. *Bayesian Networks: A Model of Self-activated Memory for Evidential Reasoning*. Report (University of California, Los Angeles. Computer Science Dept.). UCLA Computer Science Department, 1985. URL https://books.google.com.hk/books?id=1sfMOgAACAAJ.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/dc6a7e655d7e5840e66733e9ee67cc69-Paper.pdf.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods*

in Natural Language Processing, pp. 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL https://aclanthology.org/D16-1264/.

Jie Ren, Die Zhang, Yisen Wang, Lu Chen, Zhanpeng Zhou, Yiting Chen, Xu Cheng, Xin Wang, Meng Zhou, Jie Shi, and Quanshi Zhang. Towards a unified game-theoretic view of adversarial perturbations and robustness. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=fMaIxda5Y6K.

Jie Ren, Mingjie Li, Qirui Chen, Huiqi Deng, and Quanshi Zhang. Defining and quantifying the emergence of sparse concepts in dnns. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20280–20289, 2023a.

Jie Ren, Zhanpeng Zhou, Qirui Chen, and Quanshi Zhang. Can we faithfully represent absence states to compute shapley values on a dnn? *International Conference on Learning Representations*, 2023b.

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

Qihan Ren, Jiayang Gao, Wen Shen, and Quanshi Zhang. Where we have arrived in proving the emergence of sparse symbolic concepts in ai models. *International Conference on Learning Representations*, 2024.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1409.1556.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://aclanthology.org/D13-1170/.

Stanford CS231N. Tiny imagenet visual recognition challenge. https://cs231n.stanford.edu/tiny-imagenet-200.zip, 2015. Standard public Tiny-ImageNet-200 release.

SymTrust AI. SymTrust AI, 2026. URL https://symtrustai.com/en.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*, 2019. URL https://arxiv.org/abs/1908.08962.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446/.

Lechao Xiao, Jeffrey Pennington, and Samuel Schoenholz. Disentangling trainability and generalization in deep neural networks. In *International Conference on Machine Learning*, pp. 10462–10472. PMLR, 2020.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, and Jianxin Yang, et al. Qwen2.5 technical report, 2024. URL https://arxiv.org/abs/2412.15115.

Hao Zhang, Sen Li, Yinchao Ma, Mingjie Li, Yichen Xie, and Quanshi Zhang. Interpreting and boosting dropout from a game-theoretic view. *International Conference on Learning Representations*, 2021.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/250cf8b51c773f3f8dc8b4be867a9a02-Paper.pdf.

Huilin Zhou, Hao Zhang, Huiqi Deng, Dongrui Liu, Wen Shen, Shih-Han Chan, and Quanshi Zhang. Explaining generalization power of a dnn using interactive concepts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17105–17113, 2024.

APPENDIX

A JUSTIFICATION OF THE FIDELITY OF THE LOGICAL MODEL

A.1 THEORETICAL JUSTIFICATION

The fidelity requirement is fulfilled by the universal matching property established in Theorem A.1. This theorem demonstrates that a logical model with specific settings of I_T^{and} and I_T^{or} is able to accurately predict all outputs of the DNN across an exponential number of 2^n masked input states. The comprehensively massive sample set is then constructed as $\mathcal{Q} = \{\mathbf{x}_S \mid S \subseteq N\}$, which contains all 2^n masked instances of the input sample $\mathbf{x}$. $\mathbf{x}_S$ represents a masked input $\mathbf{x}$, in which input variables in $N \setminus S$ are masked[9].

Theorem A.1 (**universal matching property**, proven by (Chen et al., 2024))**.** *Given a DNN v and an input sample $\mathbf{x}$, let us set the scalar weights as $\forall T \subseteq N$, $I_T^{and} = \sum_{L\subseteq T}(-1)^{|T|-|L|} o_L^{and}$, $I_T^{or} = -\sum_{L\subseteq T}(-1)^{|T|-|L|} o_{N\setminus L}^{or}$ where $o_L^{and} = 0.5 \cdot v(\mathbf{x}_L) + \gamma_L$ and $o_L^{or} = 0.5 \cdot v(\mathbf{x}_L) - \gamma_L$, and set $b = v(\emptyset)$ (the network output when all input variables in $\mathbf{x}$ are masked.) $\{\gamma_L\}$ are a set of learnable parameters. The logical model can then accurately predict the network outputs as follows, regardless of how $\mathbf{x}$ is randomly masked:*

$$\forall \mathbf{x}' \in \mathcal{Q}, \quad \ell(\mathbf{x}') = v(\mathbf{x}'). \tag{9}$$

A.2 SETTING OF ϵ

According to (Chen et al., 2024; Ren et al., 2024; Li & Zhang, 2023a), the presence of noise ϵ in the model output $v(\mathbf{x})$ can significantly impede the accurate computation of interactions. Following the approach outlined in (Li & Zhang, 2023a) to address this challenge, we remove the noise ϵ from the output $v(\mathbf{x})$ to enable the extraction of AND-OR interactions. More specifically, the decomposition $v(\mathbf{x}_S) = o_S^{\text{and}} + o_S^{\text{or}} + \epsilon_S$ is reformulated such that $o_S^{\text{and}} = 0.5 \cdot (v(\mathbf{x}_S) - \epsilon_S) + \gamma_S$ and $o_S^{\text{or}} = 0.5 \cdot (v(\mathbf{x}_S) - \epsilon_S) - \gamma_S$. Consequently, the parameters $\{\gamma_S\}$ and $\{\epsilon_S\}$ are optimized within a LASSO-like loss framework. The values of $\{\epsilon_S\}$ were constrained to $[-\rho, \rho]$ where $\rho = 0.01 \cdot |v(\mathbf{x}) - v(\mathbf{x}_\emptyset)|$.

B COMPUTATION OF INTERACTIONS

B.1 SETTING OF INTERACTION COMPUTATION

Theoretically, our method is a generic algorithm applicable to DNNs with an arbitrary number of input variables n. For an input sample $\mathbf{x}$ with n distinct input variables, the total number of subsets

[9]The masking of an input variable means substituting this variable with a baseline value. We set the baseline value of an input variable as the average value of this variable across different input samples. For NLP models, we use a specific embedding in (Cheng et al., 2024; Ren et al., 2024) to mask input tokens. [5]

to evaluate scales as 2^n, rendering exhaustive computation intractable when each pixel (in vision) or each token (in language) is treated as an individual variable. To mitigate this, we adopt the strategies of prior works (Li & Zhang, 2023b; Liu et al., 2023) by restricting the interaction analysis to exactly 10 variables, as detailed below. Crucially, these strategies are designed such that the distribution of interactions remains similar, regardless of the value of n chosen for the computation.

Image task. We conduct experiments on image classification using convolutional networks (ResNet-50, ResNet-101, VGG-19). From a chosen intermediate feature map of each model (the first ReLU layer), we partition the spatial dimensions into a 5×5 grid and sample 10 patches to form the set N. We set the baseline value of each input variable to its empirical mean across the dataset. For any subset $S \subseteq N$, a masked input $\mathbf{x}_S$ is generated by replacing all patches in $N \setminus S$ with b, thereby enabling efficient estimation of interaction strengths.

Natural language task. We can group multiple tokens into phrases, or entire sentences and paragraphs for lengthy prompts, treating each group as a discrete input variable. For each input text, stop words, punctuation, and non-alphanumeric symbols are removed. In this way, we then randomly select 10 input variables with semantic meanings to constitute N. To construct each masked sample, we follow the masking embedding scheme of (Cheng et al., 2024; Ren et al., 2024): for BERT models, every token in $N \setminus S$ is substituted with the pre-trained [MASK] embedding vector (ID=103), yielding $\mathbf{x}_S$ and facilitating computation of higher-order token interactions.

Algorithm 1 Extract Interactions

Input: Trained DNN $v(\cdot)$; input sample $\mathbf{x}$ (features indexed $\{1, \ldots, n\}$; let $N = \{1, \ldots, n\}$ be the set of these indices).
Constants assumed: number of top interactions k.
Output: Interaction weights $I_S^{\text{and}}, I_S^{\text{or}}$ for all $S \subseteq N$; bias b; sets of top-k interactions $\Omega_{\text{and}}, \Omega_{\text{or}}$.
for each $S \subseteq N$ **do**
  Generate masked input $\mathbf{x}_S$ and the corresponding model output $v(\mathbf{x}_S)$.
end for
for each $S \subseteq N$ **do**
  Determine $I_S^{\text{and}}, I_S^{\text{or}}$ by solving $\min_{\{\epsilon_S\, \gamma_S\}} \sum_{S \subseteq N} (|I_S^{\text{and}}| + |I_S^{\text{or}}|)$.
end for
Determine the bias term b as the network's output for the empty set of features: $b \leftarrow v(\emptyset)$.
for each $S \subseteq N$ **do**
  Filter $|I_S^{\text{and}}|$ and $|I_S^{\text{or}}|$ to identify the most significant k interactions to construct Ω_{and} and Ω_{or}.
end for
return $I_S^{\text{and}}, I_S^{\text{or}}$, bias b, and sets $\Omega_{\text{and}}, \Omega_{\text{or}}$.

B.2 Pseudocode for computation of interactions

Building upon the framework established in (Chen et al., 2024; Ren et al., 2023b), the parameters $\{\gamma_L\}$ are optimized by minimizing a LASSO-regularized loss $\min_{\{\gamma_L\}} \|\mathbf{I}_{\text{and}}\|_1 + \|\mathbf{I}_{\text{or}}\|_1$ to obtain the sparsest interaction patterns, where we vectorize interactions as $\mathbf{I}_{\text{and}} = \left[I_{T_1}^{\text{and}}, \ldots, I_{T_{2^n}}^{\text{and}}\right]^T$, $\mathbf{I}_{\text{or}} = \left[I_{T_1}^{\text{or}}, \ldots, I_{T_{2^n}}^{\text{or}}\right]^T \in \mathbb{R}^{2^n}$.

Additionally, drawing upon insights from (Chen et al., 2024; Ren et al., 2024; Li & Zhang, 2023a), the presence of noise ϵ in the model output $v(\mathbf{x})$ can significantly impede the accurate computation of interactions. Following the approach outlined in (Li & Zhang, 2023a) to address this challenge, we remove the noise ϵ from the output $v(\mathbf{x})$ to enable the extraction of AND-OR interactions. More specifically, the decomposition $v(\mathbf{x}_S) = o_S^{\text{and}} + o_S^{\text{or}} + \epsilon_S$ is reformulated such that $o_S^{\text{and}} = 0.5 \cdot (v(\mathbf{x}_S) - \epsilon_S) + \gamma_S$ and $o_S^{\text{or}} = 0.5 \cdot (v(\mathbf{x}_S) - \epsilon_S) - \gamma_S$. Consequently, the parameters $\{\gamma_S\}$ and $\{\epsilon_S\}$ are optimized within a LASSO-like loss framework. The values of $\{\epsilon_S\}$ were constrained to $[-\rho, \rho]$ where $\rho = 0.01 \cdot |v(\mathbf{x}) - v(\mathbf{x}_\emptyset)|$. This process is designed to enforce sparsity and mitigate the influence of noise on the model output, as detailed in pseudocode 1.

C THE SPARSITY PROPERTY OF INTERACTIONS

In the field of symbolic interaction, the sparsity is defined as the state that almost all interactions have negligible values with only very few interactions having salient values. Particularly, (Ren et al., 2024) have further proven that under three common conditions, DNNs can encode only a small number ($\mathcal{O}\left(n^p/\tau\right) \ll 2^{n+1}$) of AND-OR interactions with salient effects, while all other interactions have almost zero effects. Specifically, τ is a tiny threshold for saliency of interactions, and empirically, $p \in [0.9, 1.2]$. The three common conditions are as follows:

Condition 1: finite-Order interactions

The network's output, denoted by a function v, can be fully characterized by interactions involving at most M input variables. This implies that any interaction effects $I(S)$ for subsets of input variables $S \subseteq \{1, \ldots, n\}$ where the size of the subset $|S| > M$ are zero: $I(S) = 0 \quad$ for all S such that $|S| > M$. Equivalently, in a Taylor expansion of the function v, all terms corresponding to mixed partial derivatives of order $M+1$ or higher vanish. Specifically, for any point $b \in \mathbb{R}^n$ and any non-negative integers $\kappa_1, \ldots, \kappa_n$ such that $\sum_{i=1}^n \kappa_i \geq M+1$, we have: $\left.\frac{\partial^{\kappa_1+\cdots+\kappa_n} v}{\partial x_1^{\kappa_1} \ldots \partial x_n^{\kappa_n}}\right|_{x=b} = 0$. Such a constraint limits model complexity to lower-order effects, aligning with observations in some large models where very high-order interaction strengths are minimal, thereby directly supporting the premise of sparse high-order interactions.

Condition 2: monotonicity of average network output

Let $v(x_S)$ be the network's output when only the features in subset S are revealed (and others are masked or set to a baseline $x_\varnothing$). Define the average increase in network output when exactly m features are revealed as: $\bar{u}(m) = \mathbb{E}_{S\subseteq\{1,\ldots,n\}:|S|=m}\left[v(x_S) - v(x_\varnothing)\right]$. This condition requires that the average network output is monotonically non-decreasing with the number of revealed features. That is, for any $m' \leq m$: $\bar{u}(m') \leq \bar{u}(m)$. Average monotonic behavior suggests feature interactions have underlying simplicity. Models dominated by many complex high-order interactions would be unlikely show such stability. This predictable response pattern indicates sparse, lower-order interactions drive outputs rather than numerous erratic high-order effects, indirectly showing the interaction structure isn't arbitrarily complex.

Condition 3: Polynomial lower bound on average output under masking

There exists a positive constant $p > 0$ such that for any number of revealed features m' and m with $m' \leq m$, the average network output $\bar{u}(m')$ (as defined under Condition 2) is lower-bounded as follows: $\bar{u}(m') \geq \left(\frac{m'}{m}\right)^p \bar{u}(m)$. The polynomial bound on output degradation when masking features suggests an interaction structure without dense high-order terms. If complex high-order interactions were critical, feature removal would cause more drastic drops. This supports the sparsity hypothesis for impactful high-order interactions.

D ABLATION STUDY TOWARDS THE NUMBER OF BASELINE DNNS

When we delve deeper into the problem of interaction generalizability, we observe that there is no clear boundary between generalizable and non-generalizable interactions. Consequently, the quantification of interaction generalizability is caught in the following dilemma.

(1) On one hand, we can only guarantee that all interactions modeled by a baseline DNN are learned from the held-out samples. However, we cannot ensure that the baseline DNN models all possible interactions. This issue necessitates training multiple baseline DNNs to cover as many different generalizable interactions as possible.

(2) On the other hand, given a group of baseline DNNs $v_1^{\text{base}}, v_2^{\text{base}}, \ldots, v_K^{\text{base}}$, each generating a set of AND-OR interactions $\Omega^k_{\text{and/or}}$, determining the generalizability of a target interaction becomes a fuzzy problem. If an interaction can only be generalized to one baseline DNN out of all K baseline DNNs, whether this interaction is truly a generalizable one remains an open problem.

Therefore, we conducted ablation experiments to illustrate the generalizable interactions that were identified by using different numbers of baseline DNNs. Given k baseline DNNs, a generalizable

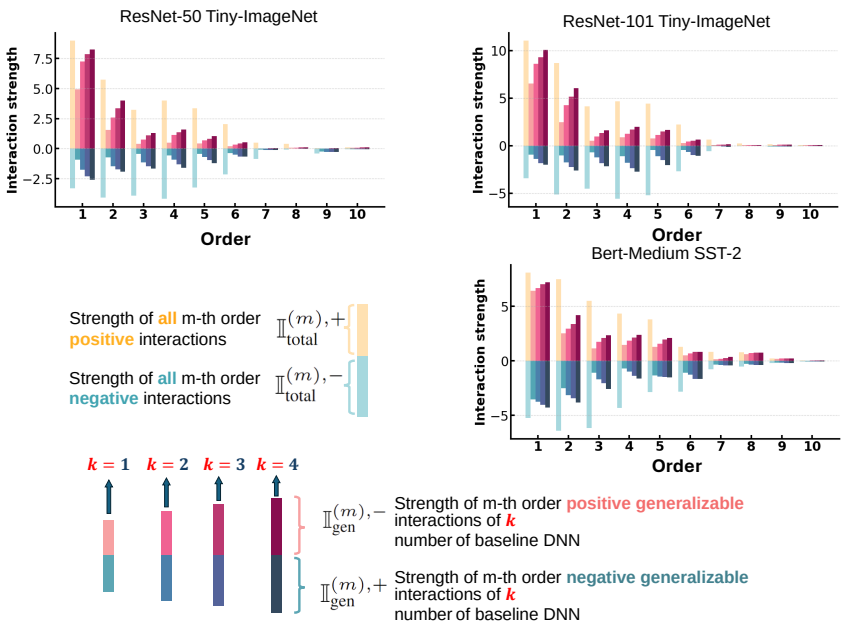

Figure 5: Ablation results of distribution of generalizable interactions with varying numbers of baseline DNNs. The distributions of interactions across different orders show a positive relationship wherein the strength of interactions increases proportionally with increasing values of k.

AND/OR interaction S was defined by its ability to generalize to any one baseline DNN, as follows.

$$\hat{\chi}_S^{\text{type}}(k) = \mathbf{1}\underbrace{\left(\chi_{S,v_1^{\text{base}}}^{\text{type}} = 1 \vee \chi_{S,v_2^{\text{base}}}^{\text{type}} = 1 \vee \cdots \vee \chi_{S,v_k^{\text{base}}}^{\text{type}} = 1\right)}_{S \text{ generalized to any one of the } \textit{baseline DNNs}} \in \{0,1\} \tag{10}$$

where type $\in$ {and, or} corresponds to AND/OR interactions, and $\mathbf{1}(\cdot)$ is a binary indicator function that outputs 1 if the condition is satisfied and 0 otherwise. Figure 5 shows the distribution of generalizable interactions and that of non-generalizable interactions under different numbers of baseline DNNs for different models. As evident from the figure, the proportion of generalizable interactions increases consistently with the number of baseline DNNs across all interaction orders. This observation aligns with our theoretical understanding that employing more baseline DNNs enhances the detection capability for generalizable interactions. These results empirically validate our approach of using multiple baseline DNNs to comprehensively capture the spectrum of generalizable interactions, thereby addressing the dilemma outlined earlier. Notably, it is evident that selecting $k = 2$ for our experiments offers a favorable balance between effectiveness and computational efficiency.

E Experimental Details

E.1 Architectures and Datasets

In our comprehensive experimental framework, we employed a diverse range of deep learning architectures across multiple benchmark datasets to systematically analyze the strength of generalizable and non-generalizable interactions across varying interaction orders. For computer vision tasks, we utilized several convolutional neural network architectures: VGG-19 and ResNet-101/Resnet-50 were trained on the Tiny-ImageNet dataset, which contains 200 classes with 500 training images per class, downsampled to 64x64 pixels. Additionally, we trained VGG-16 on the CUB-200-2011 dataset (Caltech-UCSD Birds), which consists of 11,788 images across 200 bird species. For natural language processing tasks, we leveraged transformer-based models, specifically BERT-large (24 layers, 1024 hidden dimensions) and BERT-medium (8 layers, 512 hidden dimensions), which were trained on the Stanford Sentiment Treebank (SST-2) dataset containing binary sentiment classifications of

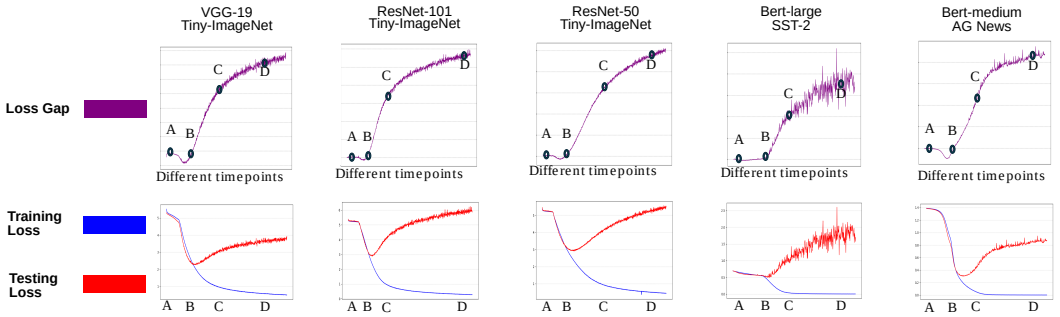

Figure 6: Training loss and testing loss for models.

movie reviews. Furthermore, we extended our NLP experiments by training both BERT-large and BERT-medium variants on the AG News dataset, which comprises news articles categorized into four classes.

For the training image preprocessing, the CUB-200-2011 dataset underwent bounding box processing to focus on the bird regions within each image, eliminating background noise that could potentially affect classification performance. The training images were then processed using data augmentation techniques including random resized cropping to 224x224 pixels, random horizontal flipping, tensor conversion, and normalization using ImageNet statistics. Similarly, Tiny-ImageNet images were preprocessed using data augmentation techniques, ensuring consistent input dimensions across all vision models despite the original differences in image resolutions between datasets.

To facilitate the training of both an original DNN and a corresponding baseline DNN, we randomly sampled and split the original training datasets to create two balanced subsets with equivalent data distributions. These equally distributed training subsets were used separately to train the original and baseline DNNs with different initialized parameters. We generally train the models following standard settings commonly used in practice. This maintained a controlled experimental environment and ensured that any observed differences between models could be attributed to our specific experimental manipulations rather than to variations in training data distribution.

E.2 Training loss and testing loss

We report the training and testing loss for the models used in Section 3.3 as shown in Figure 6.

F And-or graph

In this section, we provide additional AND-OR logical models to explain the outputs of DeepSeek-R1-Distill-Llama-8B and Qwen2.5-7B across different prompts. The results detailed in Figure 7 show that the AND-OR logical model can explain LLMs' outputs on any randomly occluded samples.

G Limitations

Although we have observed that different DNNs consistently share similar dynamics of interaction generalizability, the underlying mathematical principles and analytical formulations remain to be further explored. Moreover, owing to computational constraints, training an LLM from scratch for analytical purposes is infeasible.

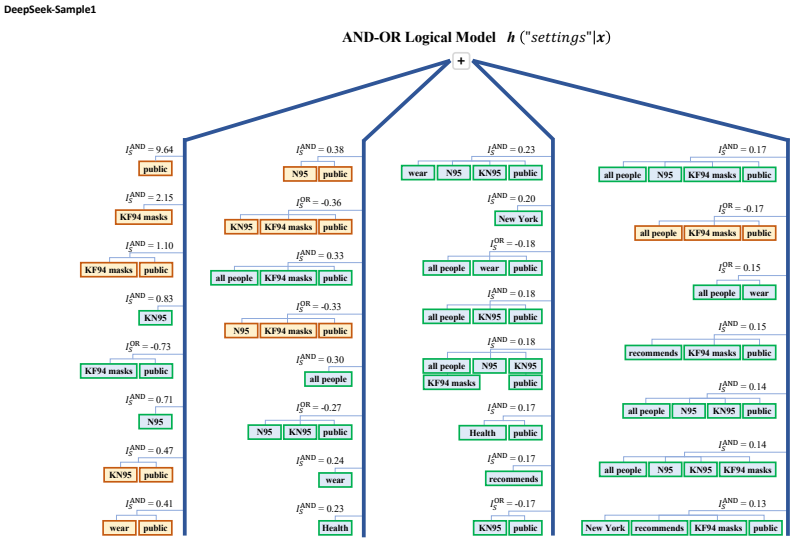

Figure 7: The AND-OR logical model successfully explains model outputs on arbitrarily occluded samples. The used distinct prompt-target pairs are: (1) Sample 1: "New York Department of Health recommends that all people should wear N95, KN95, or KF94 masks in all public" → "settings". Input variables: "New York", "Health", "recommends", "all people", "wear", "N95", "KN95", "KF94 masks", "public", "Health"; (2) Sample 2: "On June 11, 2018, OpenAI researchers and engineers published a paper introducing the first generative pre-trained" → "transformer". Input variables: "June 11, 2018,", "OpenAI", "researchers", "engineers", "published", "paper", "introducing","first", "generative", "pre-trained".

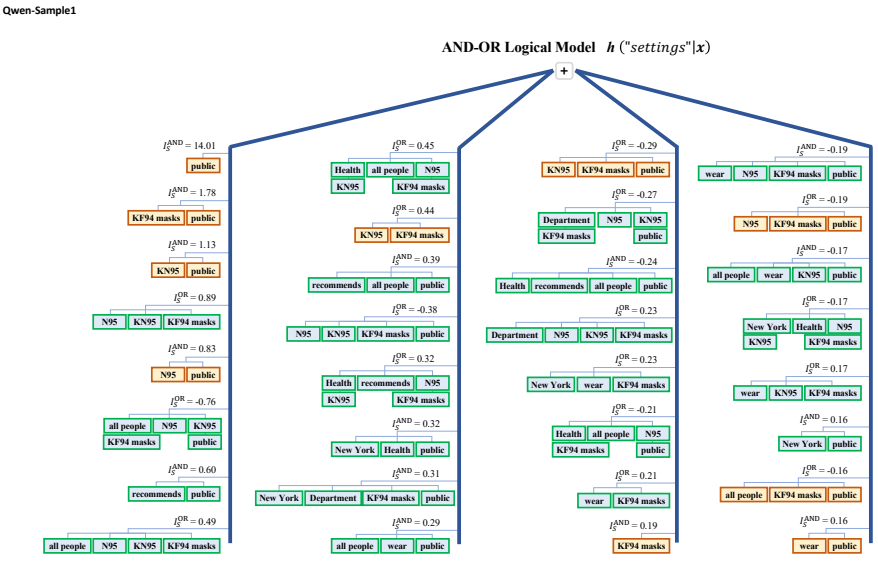

Figure 7: The AND-OR logical model successfully explains model outputs on arbitrarily occluded samples (continued).

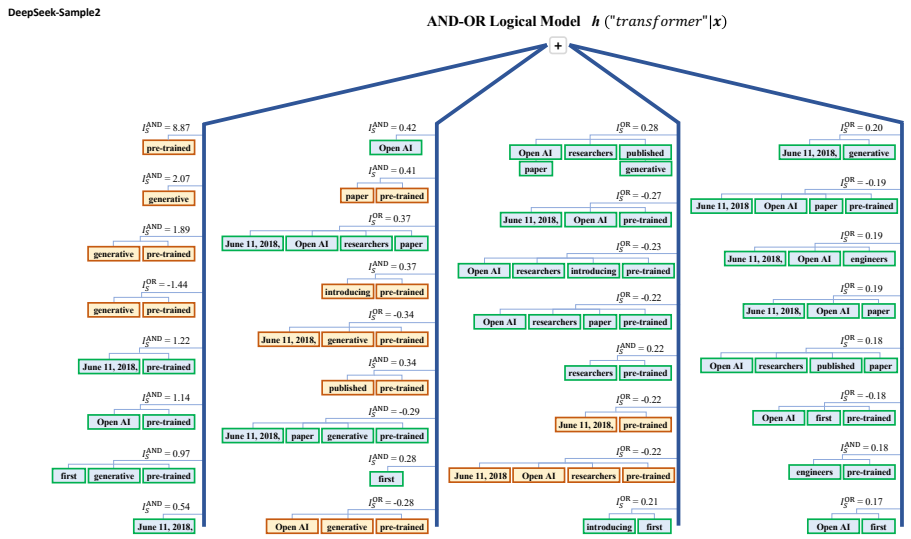

Figure 7: The AND-OR logical model successfully explains model outputs on arbitrarily occluded samples (continued).

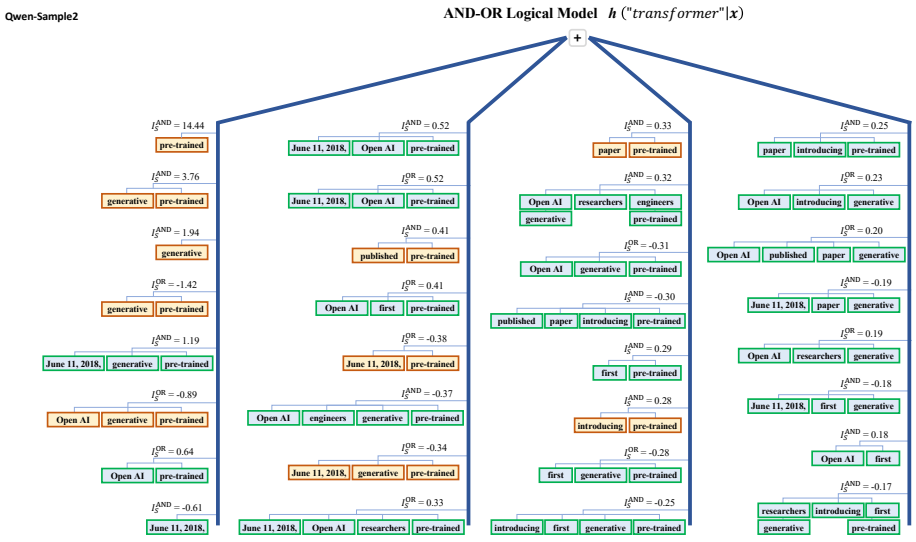

Figure 7: The AND-OR logical model successfully explains model outputs on arbitrarily occluded samples (continued).

H DEFINE THE THREE-PHASE MORE QUANTITATIVELY

The three phases are:

- **Phase 1,** $[t_0, t_1]$**:** before $\overline{\mathcal{H}}(t)$ first reaches its maximum;
- **Phase 2,** (t_1, t_2)**:** $\delta_g(t) > \delta_n(t)$;
- **Phase 3,** $[t_2, t_3]$**:** $\delta_n(t) \geq \delta_g(t)$.

Here, $\overline{\mathcal{H}}(t)$ denotes the average generalizability of interactions at time point t. The quantities $\delta_g(t)$ and $\delta_n(t)$ denote, respectively, the generalizable and non-generalizable interaction strengths acquired between adjacent time points $t-1$ and t.

The phase boundaries t_1 and t_2 are defined as

$$t_1 = \min\left(\arg\max_t \overline{\mathcal{H}}(t)\right), \tag{11}$$

and

$$t_2 = \min\left\{t \,|\, t > t_1,\ \delta_n(t) \geq \delta_g(t)\right\}. \tag{12}$$

For an AND/OR interaction S extracted at time point t, $I_S(t)$ denotes the effect of interaction S,

$$a_S(t) = |I_S(t)| \tag{13}$$

denotes its absolute effect strength, and $\mathcal{G}_S(t) \in \{0, 1\}$ indicates whether interaction S is generalizable.

The acquired generalizable and non-generalizable interaction strengths are computed as

$$\delta_g(t) = \sum_S \mathcal{G}_S(t) \max\left\{a_S(t) - a_S(t-1),\ 0\right\}, \tag{14}$$

and

$$\delta_n(t) = \sum_S \left(1 - \mathcal{G}_S(t)\right) \max\left\{a_S(t) - a_S(t-1),\ 0\right\}. \tag{15}$$

I EXPERIMENT ON SUBTRACTING THE EFFECTS OF NON-GENERALIZABLE INTERACTIONS FROM THE MODEL OUTPUT

I.1 SAMPLE SELECTION FOR LOSS COMPUTATION

The loss comparisons in Section 3.2, encompassing both standard and modified loss calculations after subtracting the effects of non-generalizable interactions, were derived from an average over 100 samples per dataset. For the Tiny-ImageNet dataset, these 100 samples were obtained by first randomly selecting 10 classes, and then randomly drawing 10 image instances from each chosen class. This sample selection process was applied independently to both the training and testing sets to gather the respective samples for loss calculation. For SST-2 and AG News datasets, 100 samples were randomly drawn from within all available classes, aiming for a proportional representation or a balanced distribution across classes (e.g., attempting to sample $m = 100/C$ instances from each of the C classes, with necessary adjustments made for class imbalances or varying class sizes).

I.2 FUTURE WORK AND POTENTIAL DIRECTIONS

The demonstration that non-generalizable interactions are a key driver of overfitting, as discussed in Section 3.2, suggests several compelling research avenues. Future efforts could focus on developing novel training paradigms, including targeted regularization, interaction-aware network pruning, or refined early stopping criteria, all designed to proactively prevent or penalize the formation of these detrimental interactions. Another promising direction involves moving beyond direct effect subtraction, exploring methods to dynamically modulate or down-weight the influence of identified non-generalizable interactions during training or inference, potentially guided by a deeper understanding of their emergence.

J EXPERIMENT OF PENALIZING NON-GENERALIZABLE INTERACTIONS IN DNNS

To bridge the theoretical discussion in our paper with practical implementation, we conducted a follow-up experiment. The core idea was to leverage a dedicated held-out set to identify and subsequently penalize non-generalizable interactions and classification loss during the end-to-end training process.

Our approach consisted of two key steps. First, to **quantify the generalizability of interactions**, we trained a baseline Deep Neural Network (DNN) on a separate held-out set. This model allowed us to quantify which of the learned feature interactions were non-generalizable. Second, to **design the penalty term**, we designed a specialized loss function based on the quantification from the previous step. During the main model's training, this function directly penalized the absolute values of the interactions identified as non-generalizable, denoted as $|I_S^{\text{NoneG}}|$, along with minimization of the classification loss.

We evaluated this approach on the CIFAR-10 dataset using a VGG-11 architecture, and the results were encouraging. The method led to a clear improvement in the quality of learned interactions: The proportion of generalizable interactions, as measured by $\overline{\mathcal{H}}$ defined in Section 3.1, increased significantly from 28% to 39%. Correspondingly, the proportion of non-generalizable interactions decreased from 72% to 61%.

Regarding the impact on model accuracy, the proposed method consistently improves performance across different architectures and datasets. For the VGG-16 model on the Tiny-ImageNet dataset, test accuracy increased by **2.5% (46.3% to 48.8%)**. Similarly, for the BERT-Tiny model on the SST-2 sentiment classification task, accuracy improved by **4.4% (72.2% to 76.6%)**. The algorithm also demonstrates significant gains in few-shot learning scenarios. For instance, in an object recognition task with only 200 training samples, our method achieved a **6.0% (76.6% to 82.6%)** improvement in classification accuracy using a ResNet-18 model.

In summary, this experiment demonstrated that our proposed method effectively reduced the learning of non-generalizable interactions and improved the neural network's performance, showing particular promise in situations with limited training data.

K COMPUTATIONAL RESOURCES AND PROCEDURAL DURATIONS

The baseline deep neural networks (DNNs) were trained on a dedicated server equipped with four NVIDIA GeForce RTX 3080 Ti GPUs and powered by Intel(R) Xeon(R) Gold 6146 CPU. For the large language model (LLM) experiments, we utilized a server with NVIDIA GeForce RTX 4090 GPUs to accommodate the increased memory and computational demands.

The model training durations varied depending on the specific experiment and model complexity:

- **Ablation study:** The training for ablation studies, involving up to five baseline DNNs, represented the most time-consuming part. Due to the necessity of training these models sequentially or on separate GPUs to ensure resource isolation, the cumulative training time for these experiments was approximately 50 hours.
- **Small to medium models:** For other experiments involving small to medium-sized models such as ResNet-50, individual training runs were typically completed within 12 hours per model.
- **Larger models:** Training larger models, including ResNet-101 and BERT-large, generally concluded within 30 hours per model.

Regarding specific experimental procedures:

- **Interaction extraction:** The process of extracting each individual interaction took approximately 60 seconds.
- **Generalizable interaction assessment:** The evaluation of interaction generalizability was comparatively swift. Assessing the generalizability for a set of 100 samples was typically completed within a range of 8 to 14 seconds.

These resources and timeframes allowed for comprehensive experimentation and validation of the proposed methods.

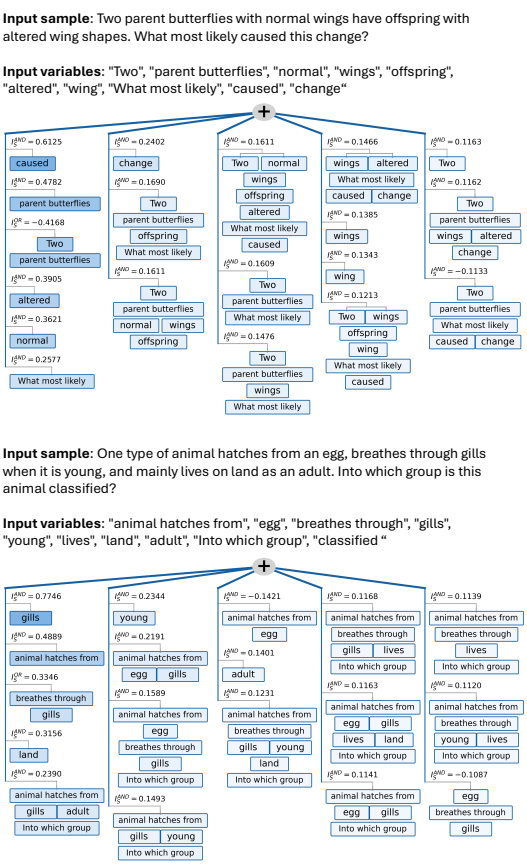

Figure 8: Examples of input variable selection on NLP data.



Figure 9: Examples of input variable selection on image data.

L EXAMPLES OF INPUT VARIABLE SELECTION

We provide examples of input variable selection on image data in Figure 9, where the input variables are highlighted with green boxes. And we provide examples of input variable selection on natural language data in Figure 8.

M REDEFINITION OF THE GENERALIZABILITY METRIC

In this section, we provide the redefinition of the generalizability metric for interactions in Equation (3). Instead of the binary metrics, we use a continuous metric within the range $[0, 1]$ to quantify the generalizability of interactions between the target model v and the baseline model v^{base}.

For each interaction S (an AND interaction or an OR interaction), the generalizability is redefined as the similarity between the interaction effect in the target model and that in the baseline model as follows:

$$\mathcal{G}'^{\text{and}}_{S,v^{\text{base}}} = \mathbf{1}\left(I^{\text{and}}_S \cdot I^{\text{and}}_{S,v^{\text{base}}} > 0\right) \cdot \frac{\min\left(\left|I^{\text{and}}_S\right|, \left|I^{\text{and}}_{S,v^{\text{base}}}\right|\right)}{\max\left(\left|I^{\text{and}}_S\right|, \left|I^{\text{and}}_{S,v^{\text{base}}}\right|\right)}, \tag{16}$$

$$\mathcal{G}'^{\text{or}}_{S,v^{\text{base}}} = \mathbf{1}\left(I^{\text{or}}_S \cdot I^{\text{or}}_{S,v^{\text{base}}} > 0\right) \cdot \frac{\min\left(\left|I^{\text{or}}_S\right|, \left|I^{\text{or}}_{S,v^{\text{base}}}\right|\right)}{\max\left(\left|I^{\text{or}}_S\right|, \left|I^{\text{or}}_{S,v^{\text{base}}}\right|\right)}. \tag{17}$$

Here, $\mathbf{1}(\cdot)$ is an indicator function, which returns 1 if the condition holds, and returns 0 otherwise. The ratio term $\frac{\min(|I_S|,|I_{S,v^{\text{base}}}|)}{\max(|I_S|,|I_{S,v^{\text{base}}}|)} \in [0,1]$ measures the similarity of the interaction effects encoded by the two models, which reflects generalizability of this interaction. Only when the same interaction S (or OR interaction S) is simultaneously extracted by both DNNs and exhibits exactly the same effect, we consider it to be 100.0% generalizable.

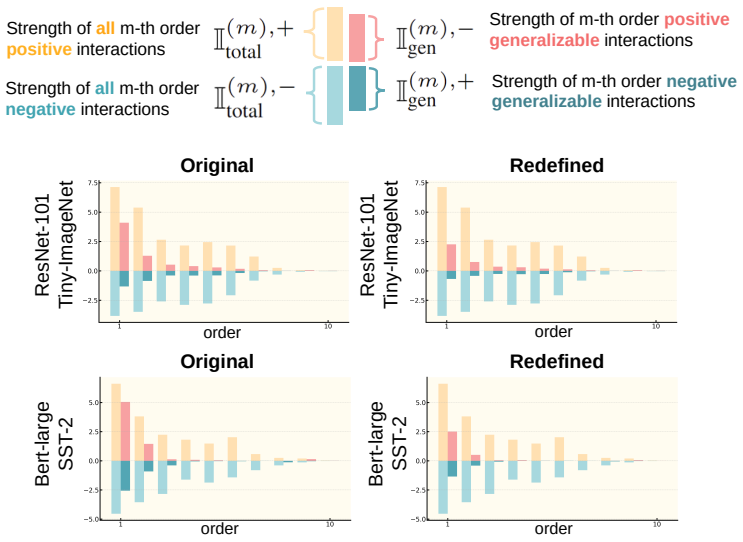

Figure 10: Analysis of generalizability across interaction orders. (Left) The distribution of interactions calculated with the original metric (χ) titled with "**Origin**". (Right) The distribution of interactions calculated with the redefined metric (χ') titled with "**Redefined**".

Normalization. Considering that the outputs of different DNNs often exhibit different ranges of perturbation, it is necessary to normalize the interaction effects to ensure fair comparison. We normalize I^{and}_S and I^{or}_S as: $I^{\text{and}}_S \leftarrow \frac{I^{\text{and}}_S}{\mathbb{E}_{\mathbf{x}}[|v(N)-v(\emptyset)|]}$, and $I^{\text{or}}_S \leftarrow \frac{I^{\text{or}}_S}{\mathbb{E}_{\mathbf{x}}[|v(N)-v(\emptyset)|]}$, where $\mathbb{E}_{\mathbf{x}}[|v(N)-v(\emptyset)|]$ represents the expected output variation of the model.

Using this redefined metric $\mathcal{G}'^{\text{and}}_{S,v^{\text{base}}}$ and $\mathcal{G}'^{\text{or}}_{S,v^{\text{base}}}$, we analyzed the distribution of generalizable interactions across different interaction orders on ResNet-101 trained with Tiny-ImageNet and Bert-large trained with SST-2. As shown in Figure 10, the new results aligned closely with results based on the old metric. Because the new metric is more stringent, the strength of generalizable interactions is reduced. However, the distribution of generalizable interactions over different orders based on the new metric is still similar to that based on the old metric.

N Evaluation of quantification of interaction generalizability

We conducted an experiment to explicitly evaluate the held-out-sample-based quantification of interaction generalizability. Specifically, we uniformly partitioned the available training data into three disjoint subsets (A, B, and C) to perform cross-validation. We trained the target DNN on dataset

A, and trained two baseline DNNs on datasets B and C, respectively. Here, dataset B and dataset C served as two independent held-out splits for the target DNN trained on dataset A. Then, given each input sample, we followed Equation (4) in Section 2.2 in the manuscript to measure the distribution (i.e., $\mathbb{I}^{(m),+}_{\text{gen}_\text{B}}$ and $\mathbb{I}^{(m),-}_{\text{gen}_\text{B}}$) of interactions that could generalize to held-out split B and the distribution (i.e., $\mathbb{I}^{(m),+}_{\text{gen}_\text{C}}$ and $\mathbb{I}^{(m),-}_{\text{gen}_\text{C}}$) of interactions that can generalize to held-out split C.[10]

Table 1: Jaccard Similarity (J) of generalizable interaction distributions between held-out splits B and C.

Model and Dataset	**Jaccard Similarity (J)**
VGG-16 on Tiny-ImageNet	85.3%
BERT-medium on SST-2	87.5%

This evaluation was conducted to test whether the interactions that could generalize to one held-out split could also generalize to the other held-out split. If feasible, the faithfulness of our method was validated. Specifically, we computed the Jaccard similarity between the two distributions as follows:

$$J = \frac{\sum_{m=1}^{n} \Big[\min\Big(\Big|\mathbb{I}^{(m),+}_{\text{gen}_\text{B}}\Big|, \Big|\mathbb{I}^{(m),+}_{\text{gen}_\text{C}}\Big| \Big) + \min\Big(\Big|\mathbb{I}^{(m),-}_{\text{gen}_\text{B}}\Big|, \Big|\mathbb{I}^{(m),-}_{\text{gen}_\text{C}}\Big| \Big) \Big]}{\sum_{m=1}^{n} \Big[\max\Big(\Big|\mathbb{I}^{(m),+}_{\text{gen}_\text{B}}\Big|, \Big|\mathbb{I}^{(m),+}_{\text{gen}_\text{C}}\Big| \Big) + \max\Big(\Big|\mathbb{I}^{(m),-}_{\text{gen}_\text{B}}\Big|, \Big|\mathbb{I}^{(m),-}_{\text{gen}_\text{C}}\Big| \Big) \Big]} \tag{18}$$

The experimental results in Table 1 showed that the DNNs exhibited high Jaccard similarity between the generalizable interaction distributions identified from the two held-out splits. This supported the faithfulness of our method of identifying generalizable interactions.

O EXPERIMENT ON TRANSFORMER-BASED LLMS

Extensively, we have conducted an experiment of analyzing the distribution of generalizable interactions on several open-source LLMs based on the transformer architecture, including Qwen2.5-7B and DeepSeek-R1-Distill-Llama-8B, which are well-known large language models and have been widely applied.

Notably, we acknowledge the significant challenge of retraining a foundational model from scratch to serve as a baseline. Therefore, considering that different LLMs released by various organizations/companies are typically trained on different and massive datasets, we provisionally utilized one LLM as the baseline DNN to evaluate another.

Specifically, we selected the Qwen2.5-7B model as the baseline model and the DeepSeek-R1-Distill-Llama-8B model as the target model. We used data from the SQuAD dataset and followed Equation (4) from Section 2.2 in the manuscript to measure the distribution of interactions that can generalize to the baseline DNN.

As illustrated in Figure 11, the new experimental results suggested that the two large models tended to encode low-order, simple generalizable interactions. This indicated that these well-trained LLMs might focus on learning simple patterns during training, and such simple interactions could potentially be more transferable between models. A systematic investigation of transferability across a larger set of LLMs and datasets is left to future work.

[10] Specifically, extending the definition in Equation (4), we utilized $\mathbb{I}^{(m),+}_{\text{gen}_\text{B}}$ to denote the aggregate effect strength of all generalizable interactions with positive effect extracted from the baseline DNN trained on dataset B, $\mathbb{I}^{(m),-}_{\text{gen}_\text{B}}$ to denote the aggregate effect strength of all generalizable interactions with negative effect extracted from the baseline DNN trained on dataset B, $\mathbb{I}^{(m),+}_{\text{gen}_\text{C}}$ to denote the aggregate effect strength of all generalizable interactions with positive effect extracted from the baseline DNN trained on dataset C and $\mathbb{I}^{(m),-}_{\text{gen}_\text{C}}$ to denote the aggregate effect strength of all generalizable interactions with negative effect extracted from the baseline DNN trained on dataset C.

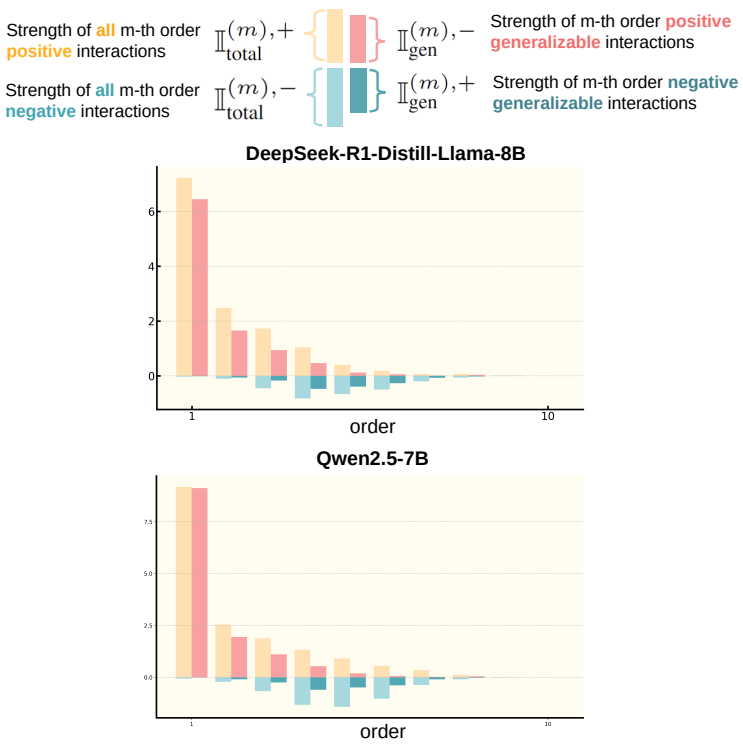

Figure 11: Distribution of interactions measured between Qwen2.5-7B (baseline) and DeepSeek-R1-Distill-Llama-8B (target) using SQuAD data following Equation 4 in Section 2.2. The plot illustrates that the two LLMs tend to encode low-order, simple interactions that are more transferable.

P Analyzing LLM fine-tuning using our method

For the SFT process, we used the Unilaw-R1-Data (Cai et al., 2025) to fine-tune the Qwen2.5-7B-Instruct model, which was originally trained on 18T tokens of data. Experimental results showed that during the SFT process, low-order interactions increased, while high-order interactions decreased. The proportion of generalizable interactions changed from 45.2% to 52.3%. The current experimental results showed that the SFT process was still within the first phase. Figure 12 shows the distribution of interactions before and after the SFT process. We will continue to finetune the LLM and keep reporting latest distribution of interactions to characterize the dynamics in the second and third phases in future. The above experiments show that our method remains applicable to analyzing LLM training.

Q Licenses and Terms of Existing Assets

Table 2 summarizes the sources, versions, licenses, and terms of use for the datasets, model architectures, and pretrained models used in this paper. We used the official or standard public versions of these assets and followed their corresponding citation and usage requirements.

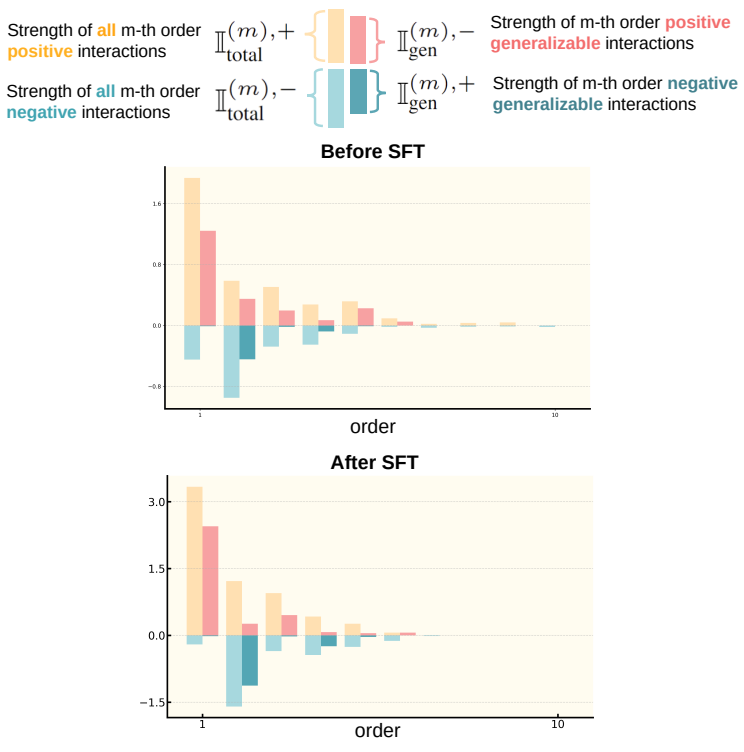

Figure 12: Distribution of interactions before and after the SFT process.

Table 2: Sources, versions, licenses, and terms of use for existing assets used in this paper.

Asset	**Source / Citation / Version**	**License / Terms**	**URL**
Tiny-ImageNet	Stanford CS231N; derived from ImageNet (Deng et al., 2009; Stanford CS231N, 2015). We used the standard Tiny-ImageNet-200 public release.	Subject to ImageNet terms; used for non-commercial academic research and educational purposes.	`https://cs231n.stanford.edu/tiny-imagenet-200.zip`
CUB-200-2011	Caltech-UCSD Birds dataset (Wah et al., 2011). We used the official CUB-200-2011 release.	For non-commercial research purposes only; original image copyrights remain with their owners.	`https://www.vision.caltech.edu/datasets/cub_200_2011/`
SST-2	Stanford Sentiment Treebank (Socher et al., 2013); GLUE SST-2 split (Wang et al., 2018). We used the official/standard binary classification split.	No separate explicit dataset license was found on the official SST/GLUE public pages. The ACL Anthology license applies to the paper, not necessarily to the dataset. We used the official/standard SST-2 split only for academic research experiments.	`https://nlp.stanford.edu/sentiment/`; `https://gluebenchmark.com/tasks`
AG News	AG News topic classification benchmark constructed by Zhang et al. (Zhang et al., 2015) from the AG news corpus. We used the standard 4-class topic classification split.	The Hugging Face dataset card lists the license as unknown. The original AG corpus page permits research/non-commercial use and does not authorize redistribution under a different name.	`http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html`; `https://huggingface.co/datasets/fancyzhx/ag_news`

Continued on next page.

Table 2: Sources, versions, licenses, and terms of use for existing assets used in this paper (continued).

Asset	Source / Citation / Version	License / Terms	URL
SQuAD	Stanford Question Answering Dataset (Rajpurkar et al., 2016). We used the official public SQuAD release used in standard QA experiments.	CC BY-SA 4.0.	https://rajpurkar.github.io/SQuAD-explorer/
ResNet	Residual Networks (He et al., 2016). We used standard ResNet-50 and ResNet-101 architectures.	Architecture from the original paper; torchvision implementation is BSD-3-Clause licensed if used.	https://arxiv.org/abs/1512.03385; https://github.com/pytorch/vision
VGG	Very Deep Convolutional Networks (Simonyan & Zisserman, 2015). We used standard VGG-16 and VGG-19 architectures.	Architecture from the original paper; torchvision implementation is BSD-3-Clause licensed if used.	https://arxiv.org/abs/1409.1556; https://github.com/pytorch/vision
BERT	BERT (Devlin et al., 2019); compact BERT models (Turc et al., 2019). We used standard BERT-large and BERT-medium style models in experiments.	Official Google BERT code and released checkpoints are under Apache License 2.0.	https://github.com/google-research/bert
Qwen2.5	Qwen2.5 official model family (Yang et al., 2024). We used the 7B official releases.	Apache License 2.0 for the official Hugging Face releases.	https://huggingface.co/Qwen/Qwen2.5-7B; https://huggingface.co/Qwen/Qwen2.5-7B-Instruct
DeepSeek-R1-Distill-Llama-8B	DeepSeek-R1 distilled model (Guo et al., 2025). We used the official DeepSeek-R1-Distill-Llama-8B release.	The official DeepSeek-R1 release is under the MIT License; the Distill-Llama variant is also subject to the Llama 3.1 license terms.	https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-8B; https://huggingface.co/deepseek-ai/DeepSeek-R1
Unilaw-R1-Data	Unilaw-R1-Data (Cai et al., 2025). We used the official repository release.	The associated ACL Anthology paper is licensed under CC BY 4.0. No explicit dataset license or redistribution terms were found in the official GitHub repository; we used the released data only for academic research experiments.	https://github.com/Hanscal/Unilaw-R1/tree/main/dataset; https://aclanthology.org/2025.emnlp-main.915/