DO LLMS LEARN UNIVERSAL SYMBOLIC REPRESENTATIONS? AN EARLY EXPLORATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Exploring internal cognitive patterns of large language models (LLMs) has attracted growing research interest. In this paper, we use interaction patterns as an interpretable and verifiable metric to rigorously decompose the inference logic of an LLM into symbolic AND-OR interactions, with theoretical guarantees of explanatory completeness. We find that mainstream open-weight LLMs exhibit highly consistent symbolic interaction patterns despite substantial differences in parameter scale and training data. Compared with model-specific, non-shared interactions, these cross-model shared interaction patterns exhibit lower structural complexity and weaker positive-negative cancellation, making them more compelling as inference patterns. We further show that a large fraction of LLM prediction scores is attributable to these shared interaction patterns. Overall, our work validates the hypothesis that LLMs encode inherent symbolic patterns and provides a precise and verifiable decomposition of cognitive patterns that are universally shared across LLMs.

1 INTRODUCTION

Rather than developing a new learning algorithm, in this paper, we study a fundamental question about large language models (LLMs): whether independently trained LLMs can learn *shared symbolic inference patterns*. A growing body of work suggests that independently trained neural networks may converge toward similar internal representations, even when they differ in model architecture, training data, or modality (Bansal et al., 2021; Huang et al., 2025; Son et al., 2025). The Platonic Representation Hypothesis gives this trend a broader interpretation: increasingly capable models may be driven toward a shared statistical model of reality (Huh et al., 2024). In parallel, recent work, such as sparse autoencoder (Huben et al., 2024; Templeton et al., 2026) and J-space (Gurnee et al., 2026), has shown that these internal representations can be associated with verbalizable and human-interpretable patterns.

However, existing studies of cross-model shared representations are confined to coarse feature-space analysis and inherently limited by pervasive neuronal polysemanticity. Verifiable metrics are still lacking to rigorously extract explicit symbolic patterns with clear semantic boundaries and establish strict one-to-one correspondences of these shared patterns across LLMs. To move beyond such indirect evidence, we argue that a rigorous validation of shared symbolic patterns should satisfy the following three requirements, as shown in Figure 1.

• The inference patterns must have clear semantic boundaries. Besides, the explanation should come with a theoretical guarantee of completeness, *i.e.*, such inference patterns should faithfully reproduce the detailed inference logic of an LLM on a given sample within a bounded approximation error.

• The learning of different LLMs should converge to a set of similar inference patterns.

• Shared inference patterns should make a substantial contribution to the LLM's final predictions.

Thus, inspired by recent developments in interaction-based explanation (Ren et al., 2023; 2024a; Li & Zhang, 2023), we represent the complex inference logic underlying an LLM's prediction of a single token as a sparse set of interactions. As illustrated in Figure 1, each interaction captures a nonlinear relationship among a specific set of words/phrases encoded by the LLM, and it contributes a quantifiable effect to the prediction score. Moreover, the universal matching of interactions guarantees and completeness of interaction-based explanations (Ren et al., 2024b).

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.

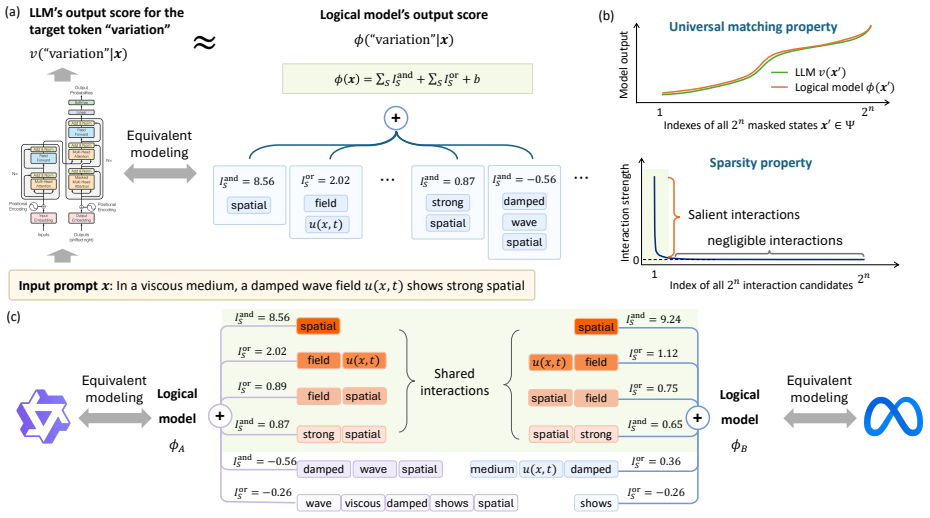

Figure 1: Interaction-based explanation satisfies three requirements. First, each interaction pattern between input variables (tokens/phrases) has a clear semantic boundary (a) and guaranteed explanation completeness, *i.e.*, the interaction-based logical model faithfully matches the LLM outputs over all 2^n masked input states (b). Second, different LLMs share a similar set of interactions (c). Third, the shared interactions make the major contribution to the models' final prediction (c).

> As illustrated in Figure 1, an interaction represents a phrase pattern automatically used by the LLM for inference. For example, the LLM encodes a phrase $S = \{\text{strong, spatial}\}$. The co-occurrence of the two words activates the interaction and contributes $I_S^{\text{and}} = 0.87$ to boost the prediction score of generating the target token "variation."

Thus, in this study, we extensively evaluate 16 classic open-weight LLMs[1] and find that *these independently trained models converge toward similar interaction patterns.*

(1) Inference logic of different independently trained LLMs usually converges to a similar set of interaction patterns. When given the same input prompt, different LLMs often rely on shared interaction patterns to compute the prediction score of the same target token, despite substantial differences in parameter scale and training data. In particular, shared interaction patterns in a neural network become increasingly prominent during the training process and contribute more strongly to high-confidence predictions. We refer to this tendency as *interaction convergence*.

(2) Shared interactions tend to exhibit more convincing patterns than non-shared interactions. Specifically, shared interactions are predominantly low-order, where the *order* of an interaction is defined as the number of input variables involved in the interaction. Prior works Liu et al. (2023); Zhou et al. (2024) suggest that lower-order interactions are more likely to capture reliable and generalizable inference patterns, whereas higher-order interactions tend to reflect memorized and sample-specific patterns.

(3) Shared interactions contribute the majority of the effective prediction score. We find that non-shared interactions often have substantial individual effects, but their positive and negative contributions strongly cancel each other. The net effect of non-shared interactions on the target-token prediction score is much smaller than that of shared interactions. This suggests that the effective prediction utility is primarily carried by the interactions shared across different LLMs, whereas much of the model-specific interaction strength is internally offset.

Taken together, our extensive empirical evidence provides a rigorous disentanglement of explicit symbolic interactions. The clear semantic boundaries and one-to-one cross-LLM correspondences of these interactions can be strictly verified.

[1]Our method can only be applied when the model parameters are publicly available.

2 INTERACTION-BASED EXPLANATION

2.1 PRELIMINARIES

Let us consider an LLM v and an input prompt $\mathbf{x} = [x_1, \ldots, x_n]^T$ with n input variables indexed by $N = \{1, \ldots, n\}$. Each input variable can be a token, a word, or a phrase. Given the prompt $\mathbf{x}$, the confidence score of generating the target token x_{n+1} is usually defined as

$$v(\mathbf{x}) := \log \frac{p(x_{n+1} \mid \mathbf{x})}{1 - p(x_{n+1} \mid \mathbf{x})}, \tag{1}$$

where $p(x_{n+1} \mid \mathbf{x})$ denotes the LLM's probability of generating the target token x_{n+1}.

AND-OR logical model. Interaction-based explanation provides a formal way to interpret the confidence score $v(\cdot)$ into a set of interactions. Each interaction corresponds to a phrase automatically used by the LLM. Specifically, Chen et al. (2024) have proved that the output score $v(\cdot)$ of a neural network can be decomposed into a set of interactions, each contributing an effect (I_S^{and} or I_S^{or}). That is, *we can construct a logical function with such interaction logic to accurately match the shape of the network function $v(\cdot)$ on exponentially many masked states[2] of the input.*

$$\begin{aligned} &\forall \mathbf{x}' \in \Psi, \qquad |\phi(\mathbf{x}') - v(\mathbf{x}')| < \epsilon, \\ &\phi(\mathbf{x}') = \underbrace{\sum_{T \in \Omega^{\text{and}}} I_T^{\text{and}} \cdot \mathbb{1}_{\text{and}}\left(\mathbf{x}' \text{triggers AND relation} \atop \text{between variables in } T\right)}_{\text{an AND interaction}} + \underbrace{\sum_{T \in \Omega^{\text{or}}} I_T^{\text{or}} \cdot \mathbb{1}_{\text{or}}\left(\mathbf{x}' \text{triggers OR relation} \atop \text{between variables in } T\right)}_{\text{an OR interaction}} + b. \end{aligned} \tag{2}$$

The above equation is referred to as the *universal-matching property*, where the tiny error bound ϵ ensures the fidelity of the interaction explanation. $\Psi = \{\mathbf{x}_S \mid S \subseteq N\}$ denotes the set of all 2^n masked states, where $\mathbf{x}_S$ represents a masked sample in which the variables in $N \setminus S$ are masked[2]. Ω^{and} and Ω^{or} denote the sets of AND and OR interactions, respectively. The corresponding interaction effects $I_T^{\text{and}}, I_T^{\text{or}} \in \mathbb{R}$ and the scalar bias b are learned following Chen et al. (2024) (Please see Appendix C for more details).

The universal-matching property has been demonstrated empirically (Li & Zhang, 2023; Liu et al., 2023; Zhou et al., 2024) and rigorously established in theory (Ren et al., 2024a), providing a formal guarantee of explanatory completeness.

The AND trigger function $\mathbb{1}_{\text{and}}(\cdot) \in \{0, 1\}$ is activated only when all variables in $T \subseteq N$ are present in $\mathbf{x}'$, capturing their synergistic effect. For example, in Figure 2(a), the AND interaction $S = \{\text{virtually}, \text{virginia}\}$ contributes a positive effect $I_S^{\text{and}} = 0.19$ to predicting the target token "min". The OR trigger function $\mathbb{1}_{\text{or}}(\cdot) \in \{0, 1\}$ is activated when at least one variable in T is present, capturing redundant effects among variables in T. For example, in Figure 2(b), the OR interaction $S = \{\text{fly}, \text{airspace}\}$ contributes a negative effect $I_S^{\text{or}} = -0.53$ to predicting "rules".

Sparsity of interactions. Ren et al. (2024a) have shown that the number of interactions extracted from a given input is usually small and theoretically bounded. In practice, empirical studies find that only 50 to 150 interactions are extracted from each input prompt. We provide additional empirical validation of the sparsity property in Appendix A.

2.2 INTERACTIONS SERVE AS PRIMITIVE INFERENCE PATTERNS

We now explain why interactions can be treated as primitive inference patterns of an LLM. An ideal primitive pattern should accurately explain the LLM's prediction score, be semantically interpretable, and be sparse. AND-OR interactions satisfy these properties from both theoretical and empirical perspectives.

[2]An input word is masked by replacing its token embedding(s) with a baseline vector $\mathbf{b}_{\text{base}} \in \mathbb{R}^d$, which represents a *no-information state* (Chen et al., 2024).

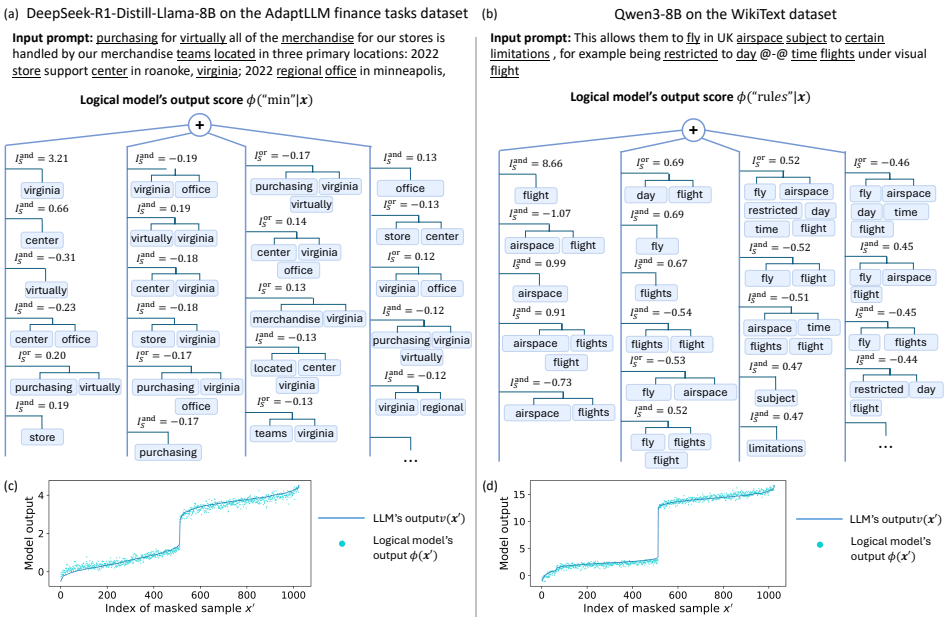

Figure 2: (a, b) Visualization of logical models extracted from LLMs. (c, d) The logical model $\phi(\mathbf{x}')$ closely matches the original LLM scores $v(\mathbf{x}')$ over 2^n masked states. Input variables are defined by following Chen et al. (2024)[5].

Functional interpretation. The logical model $\phi(\mathbf{x}')$ can be explained as follows. It decomposes the LLM prediction score on each masked state $\mathbf{x}'$ of the input into two parts $v^{\text{and}}(\mathbf{x}')$ and $v^{\text{or}}(\mathbf{x}')$, and interaction effects in $\phi(\mathbf{x}')$ can be derived by the Möbius transform (Rota, 1964).

$$\forall \mathbf{x}' \in \Psi, \quad v(\mathbf{x}') = v^{\text{and}}(\mathbf{x}') + v^{\text{or}}(\mathbf{x}'),$$
$$\text{with} \quad \{I_T^{\text{and}}\}_{T\subseteq N, T\neq\emptyset} = \text{Möbius}(\mathbf{v}^{\text{and}}), \quad \{I_T^{\text{or}}\}_{T\subseteq N, T\neq\emptyset} = -\text{Möbius}(\widehat{\mathbf{v}}^{\text{or}}), \tag{3}$$

where $\text{Möbius}(\mathbf{v}^{\text{and}})$ denotes the Möbius transform applied to the output values of the v^{and} function over the 2^n masked samples in Ψ.[3] $\mathbf{v}^{\text{and}} \in \mathbb{R}^{2^n}$ vectorizes all outputs of $v^{\text{and}}(\cdot)$ on all masked inputs, and the vector $\widehat{\mathbf{v}}^{\text{or}} \in \mathbb{R}^{2^n}$ is defined by $\forall T \subseteq N, \widehat{v}^{\text{or}}(x_T) \triangleq v^{\text{or}}(x_{N\setminus T})$.

The logical function $\phi(\cdot)$ in Equation (2) can be viewed as an inverse Möbius transform[3], which uses interaction effects $\{I_T^{\text{and}}\}$ and $\{I_T^{\text{or}}\}$ to reconstruct the above functions $v^{\text{and}}(\cdot)$ and $v^{\text{or}}(\cdot)$, and hence approximates the original network output $v(\cdot)$. Thus, learning $\{I_T^{\text{and}}\}$ and $\{I_T^{\text{or}}\}$ is equivalent to learning a decomposition of the network output $v(x') = v^{\text{and}}(x') + v^{\text{or}}(x')$, to achieve the sparse interaction effects (Ren et al., 2023; 2024a).

Empirical evidence. Figure 2 shows that extracted interactions are sparse and meaningful. Although there are 2^n possible token subsets, only a small number of interactions have salient effects in practice. Empirically, an input sample usually yields 50-150 interactions. These salient interactions often correspond to interpretable phrase-level patterns: AND interactions capture jointly activated token combinations, while OR interactions capture redundant or substitutable evidence. A positive interaction increases the target-token prediction score, while a negative interaction decreases the prediction score.

We also verify the faithfulness of the extracted interactions. As shown in Figure 2, the logical model $\phi(\mathbf{x}')$ closely matches the original LLM output $v(\mathbf{x}')$ across arbitrary masked states $\mathbf{x}' \in \Psi$. This indicates that interactions are not merely visual artifacts, but provide a compact functional approximation to the LLM prediction. The fact that very sparse interactions can accurately match LLM outputs on exponentially many samples has demonstrated the faithfulness of taking interactions as primitive inference patterns in an LLM.

[3]Please see Appendix B for details.

2.3 QUANTIFYING SHARED INTERACTIONS

Given the same prompt $\mathbf{x}$, we extract two sets of interactions, $(\Omega_A^{\text{and}}, \Omega_A^{\text{or}})$ and $(\Omega_B^{\text{and}}, \Omega_B^{\text{or}})$, from two LLMs v_A and v_B, respectively. An interaction is considered shared if it is extracted from both models and has the same sign of effect, *i.e.*, it contributes positively or negatively in both models.

Definition 1 (Shared interaction)**.** *Given the same prompt $\mathbf{x}$ as input to two different LLMs v_A and v_B, we define a binary metric to identify whether the AND interaction $S \subseteq N$ is shared by v_A and v_B, as follows.*

$$H_S^{and}=\mathbb{1}\Big[S\in\Omega_A^{and}\cap\Omega_B^{and}\Big]\cdot\mathbb{1}\Big[\text{sign}(I_{S,A}^{and})=\text{sign}(I_{S,B}^{and})\Big], \tag{4}$$

where $\text{sign}(\cdot)$ *denotes the sign function, indicating whether an interaction effect is positive or negative, and* $\mathbb{1}(\cdot) \in \{0, 1\}$ *is a trigger function that returns* 1 *if the given condition is satisfied. Shared OR interactions are defined in a similar manner,* $H_S^{or} = \mathbb{1}[S \in \Omega_A^{or} \cap \Omega_B^{or}] \cdot \mathbb{1}\left[\text{sign}(I_{S,A}^{or}) = \text{sign}(I_{S,B}^{or})\right]$.

Therefore, the sets of shared interactions, denoted by $(\Omega_{\text{sh}}^{\text{and}}, \Omega_{\text{sh}}^{\text{or}})$, are defined as

$$\Omega_{\text{sh}}^{\text{and}}=\{S\subseteq N: H_S^{\text{and}}=1\},\ \Omega_{\text{sh}}^{\text{or}}=\{S\subseteq N: H_S^{\text{or}}=1\}. \tag{5}$$

Specifically, we characterize interactions from two perspectives. Unless otherwise specified, the following metrics are computed over all interactions in $(\Omega_A^{\text{and}}, \Omega_A^{\text{or}})$ extracted from the LLM v_A.

Metric 1: complexity distribution of interactions. We measure the complexity of an interaction by its order. For an interaction $S \subseteq N$, the order is the number of input variables involved in S, *i.e.*, $\text{order}(S) = |S|$. Low-order interactions involve fewer input tokens and typically correspond to simpler phrase patterns, whereas high-order interactions involve a larger number of tokens and capture more complex phrase patterns.

We aggregate the positive and negative interaction effects separately at each order k, as follows. In this way, the complexity distribution of interactions is represented by the strength of positive interactions of different orders $[\mathbb{p}^{(1),+}, \ldots, \mathbb{p}^{(n),+}]^T$ and that of negative interactions over orders $[\mathbb{n}^{(1),-}, \ldots, \mathbb{n}^{(n),-}]^T$ (see Figure 4(a)).

$$\mathbb{p}^{(k),+} = \sum_{\text{type}\in\{\text{and,or}\}} \sum_{S\in\Omega_A^{\text{type}}:|S|=k} \max(I_{S,A}^{\text{type}}, 0), \quad \mathbb{n}^{(k),-} = \sum_{\text{type}\in\{\text{and,or}\}} \sum_{S\in\Omega_A^{\text{type}}:|S|=k} \min(I_{S,A}^{\text{type}}, 0). \tag{6}$$

Similarly, we can obtain the distribution of shared interactions $(\mathbb{p}_{\text{sh}}^{(k),+}, \mathbb{n}_{\text{sh}}^{(k),-})$ based on the sets of shared interactions in $\Omega_{\text{sh}}^{\text{and}}$ and $\Omega_{\text{sh}}^{\text{or}}$, respectively.

This distribution indicates whether the representation is dominated by low-order patterns or more complex high-order patterns. Zhou et al. (2024); Liu et al. (2023) have shown that high-order interactions are often less stable and more likely to reflect overfitted patterns, whereas low-order interactions tend to represent more reliable patterns.

Metric 2: cancellation between positive and negative interactions. Besides the order of interactions, we further quantify the cancellation between positive and negative interaction effects. Positive interactions increase the prediction score of the target token, while negative interactions decrease it. If the positive and negative effects largely offset each other, the corresponding interactions have limited net influence on the final prediction. Prior studies Ren et al. (2024b); He et al. (2025) suggest that such mutually offsetting interactions often arise in the overfitting stage of neural networks and are more likely to represent noise patterns. Thus, we define the uncancelled effect ratio as

$$\rho = \frac{\left|\sum_{\text{type}\in\{\text{and, or}\}} \sum_{S\in\Omega^{\text{type}}} I_{S,A}^{\text{type}}\right|}{\sum_{\text{type}\in\{\text{and, or}\}} \sum_{S\in\Omega^{\text{type}}} \left|I_{S,A}^{\text{type}}\right|}. \tag{7}$$

Similarly, the uncancelled effect ratio for shared interactions ρ_{sh} is defined by using shared interactions in $\Omega_{\text{sh}}^{\text{and}}$ and $\Omega_{\text{sh}}^{\text{or}}$. If ρ is close to 0, positive and negative effects largely offset each other, showing that the extracted interactions act like noise and make little net contribution to the target-token prediction. In contrast, a larger ρ indicates that most interaction effects point in a consistent direction, indicating more efficient feature representations.

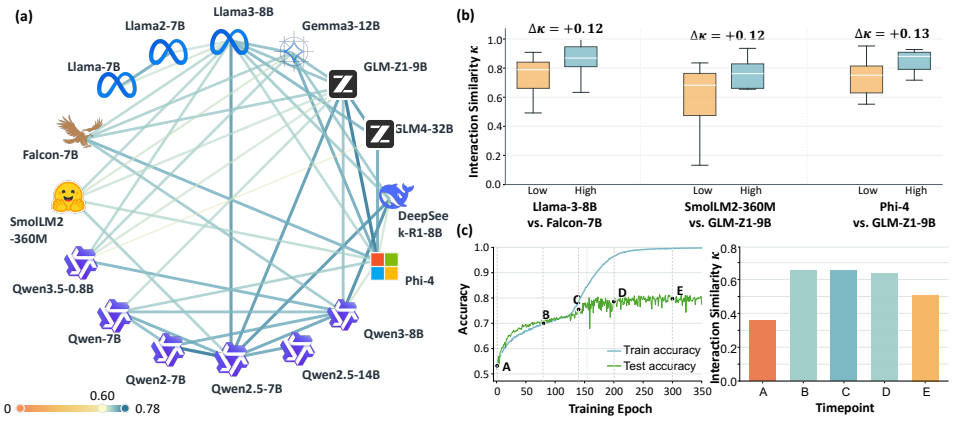

Figure 3: Shared interaction patterns and prediction-level convergence across LLMs. (a) Pairwise interaction similarity κ across 16 evaluated LLMs. Each node represents a model, and each edge represents the interaction similarity between a model pair. Substantial interaction similarity is observed both within and across model families. (b) Sample-level interaction similarity for low- and high-confidence predictions in three representative model pairs. High-confidence predictions exhibit consistently higher interaction similarity. (c) Evolution of interaction similarity during training. Interaction similarity increases as learning progresses, but decreases when the model is overfitted (during timepoints C-E when the training-testing accuracy gap begins increasing).

3 Convergent Interactions Across LLMs

In this section, we study interaction convergence at two complementary levels. We first examine whether independently trained LLMs share a non-trivial set of interaction patterns and whether such shared interactions progressively emerge during training. Second, we analyze the structural and numerical properties of these shared interactions.

3.1 Shared Interaction Patterns Across LLMs

Models and datasets. We evaluate 16 open-weight LLMs[1] spanning different model families, training data, and parameter scales, ranging from 360M to 32B parameters. The evaluated models include Gemma3-12B (Kamath et al., 2025), DeepSeek-R1-Distill-LLaMA-8B (Guo et al., 2025), Falcon-7B (Almazrouei et al., 2023), Phi-4 (Abdin et al., 2024), and SmolLM2-360M (Allal et al., 2025). Besides, we include Qwen-7B, Qwen2-7B, Qwen2.5-7B, Qwen2.5-14B, Qwen3-8B, and Qwen3.5-0.8B from the Qwen family (Bai et al., 2023; Qwen et al., 2025; Yang et al., 2025; Qwen Team, 2026), GLM-Z1-9B and GLM-4-32B from the GLM family (Glm et al., 2024), and LLaMA-7B, LLaMA2-7B, and LLaMA3-8B from the LLaMA family (Touvron et al., 2023a;b; Grattafiori et al., 2024). This diverse model collection enables us to examine whether shared interaction patterns emerge across models with substantially different capacities and training histories.

For evaluation, we sample test examples from two public datasets: the AdaptLLM finance tasks (Cheng et al., 2024) and WikiText (Merity et al., 2016), which cover domain-specific financial text and general-domain language modeling, respectively.

Metrics for shared interaction patterns across LLMs. Let us input the same prompt $\mathbf{x}$ to a pair of LLMs A and B. From the perspective of LLM A, we decompose the prediction score of the target token $v(\mathbf{x})$ into the utility of shared interactions and that of non-shared interactions according to Equation (2), as follows.

$$v(\mathbf{x}) \approx \phi(\mathbf{x}) = v^{\text{shared}}(\mathbf{x}) + v^{\text{non}}(\mathbf{x}) + b,$$
$$s.t.\ v^{\text{shared}}(\mathbf{x}) = \sum_{S\in\Omega^{\text{and}}_{\text{sh}}} I^{\text{and}}_S(\mathbf{x}) + \sum_{S\in\Omega^{\text{or}}_{\text{sh}}} I^{\text{or}}_S(\mathbf{x}), \quad v^{\text{non}}(\mathbf{x}) = \sum_{S\in\Omega^{\text{and}}\setminus\Omega^{\text{and}}_{\text{sh}}} I^{\text{and}}_S(\mathbf{x}) + \sum_{S\in\Omega^{\text{or}}\setminus\Omega^{\text{or}}_{\text{sh}}} I^{\text{or}}_S(\mathbf{x}), \tag{8}$$

where $v^{\text{shared}}(\mathbf{x})$ denotes the utility derived from interactions that are shared by the LLM B, and $v^{\text{non}}(\mathbf{x})$ denotes the utility based on interactions exclusively encoded by the LLM A. Ω^{and} and Ω^{or}

denote the set of AND interactions and OR interactions encoded by the LLM A. $\Omega^{\text{and}}_{\text{sh}} \subseteq \Omega^{\text{and}}$ and $\Omega^{\text{or}}_{\text{sh}} \subseteq \Omega^{\text{or}}$ denote the subsets of interactions within Ω^{and} and Ω^{or}, respectively, which are shared by the LLM B.

We define the metric κ to measure the ratio of the prediction score derived from shared interactions.

$$\kappa = \mathbb{E}_{\mathbf{x}\in\mathcal{D}} \frac{\left|v^{\text{shared}}(\mathbf{x})\right|}{|v^{\text{shared}}(\mathbf{x})| + |v^{\text{non}}(\mathbf{x})|}, \tag{9}$$

where the κ metric is averaged over all testing samples in the set $\mathcal{D}$. κ serves as a heuristic measure of **interaction similarity**.

For each pair of LLMs, we compute κ from the perspective of each model and report the average of the two values as their pairwise interaction similarity. Figure 3 reveals three notable findings.[4]

(1) Shared interaction patterns are broadly observed across different LLMs. Despite large differences in model family, parameter scale, and training data, the evaluated model pairs achieve high interaction similarity κ ranging from 61.8% to 78.2%, as shown in Figure 3(a). Moreover, high interaction similarity is observed not only within the same model family (*e.g.*, the κ value between Qwen-7B and Qwen3-8B is 70.8%), but also across independently developed families (*e.g.*, the κ value between GLM-Z1-9B and Phi-4 is 73.1%). Thus, the shared interactions are not merely occasional overlaps or a consequence of common model lineage.

(2) High-confidence predictions tend to rely more strongly on shared interactions. Specifically, we divide the test samples into two groups according to their target-token prediction confidence: (a) high-confidence predictions and (b) low-confidence predictions, please refer to Appendix C for experimental details. We then measure the interaction similarity κ within the two sample groups. As shown in Figure 3(b), samples with high-confidence predictions exhibit larger interaction similarity κ than those with low-confidence predictions. This indicates that when the LLM is more confident about the target-token prediction, a larger proportion of their prediction score is carried by shared interaction patterns.

(3) Neural networks trained independently on different data can converge toward shared interaction patterns. We independently train two BERT-large models (Devlin et al., 2019) with different random initializations on two disjoint subsets of the SST-2 (Socher et al., 2013) training set, please refer to Appendix C for experimental details. As shown in Figure 3(c), two independently trained models trained on disjoint datasets can spontaneously develop shared interactions (κ increases). Such shared interactions gradually diminish as the models enter the overfitting regime (κ declines). These results indicate that data similarity is not the root cause of shared interactions.

Experimental details. Let the same input prompt $\mathbf{x}$ be given to each pair of LLMs. To ensure a fair comparison, interaction extraction from the two LLMs must be conducted with respect to the prediction of the same target token. Thus, we determine the target token for explanation as follows. Specifically, we select a target token that receives a relatively high prediction score from both LLMs. For each LLM, we rank the prediction scores of all candidate tokens. We then sum the ranks of each token across the two LLMs, and select the token with the lowest summed rank as the target token.

For each LLM and prompt, we analyze AND–OR interactions with respect to the target-token score in Equation (1). We follow Chen et al. (2024) to construct masked input $\mathbf{x}_S$ by keeping the variables in $S \subseteq N$ unchanged and replacing all other variables with the baseline embedding.[5]

3.2 What Characterizes the Shared Interactions?

Comparing interactions extracted from two LLMs. Given two LLMs and an input prompt $\mathbf{x}$, we use Equation (4) to identify a set of interactions shared by the two LLMs to predict the target token, as well as two sets of non-shared interactions corresponding to the two LLMs, respectively.

[4] We report the results on the AdaptLLM finance dataset in Figure 3 and Figure 4. Results on WikiText are provided in Appendix D, where we observe the same overall trend.

[5] Variables are defined at the word level, and all subword tokens belonging to the same variable are masked together to ensure consistency. In the experiments, for computational efficiency, we randomly select ten semantically meaningful words from each prompt as input variables.

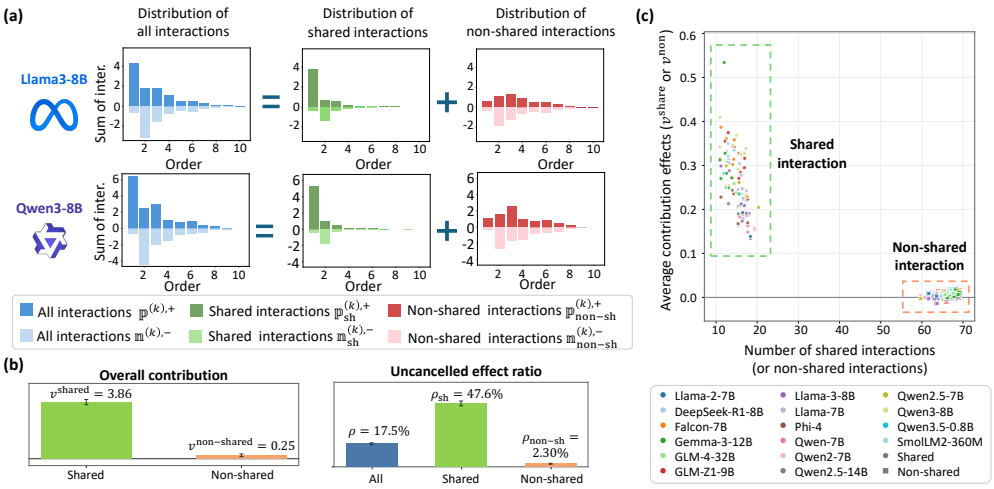

Figure 4: Characteristics of shared and non-shared interactions. (a) For representative model pairs, the distribution of all interactions over different orders can be decomposed into shared and non-shared components. (b) Most non-shared interactions exhibit significant mutual cancellation of the positive and negative effects. This makes the overall contribution of shared interactions $v^{\text{shared}}(\mathbf{x})$ much greater than the contribution of non-shared interactions $v^{\text{non}}(\mathbf{x})$. We also report the higher uncancelled-effect ratio for shared interactions ρ_{sh} and the lower uncancelled-effect ratio for non-shared interactions $\rho_{\text{non-sh}}$. (c) Comparison of the number and average contribution effect of shared interactions and non-shared interactions. Each dot represents a specific LLM.

Figure 4(a) shows the distributions of all interactions $[\mathbb{p}^{(1),+},\dots,\mathbb{p}^{(n),+}]^T$, $[\mathbb{m}^{(1),-},\dots,\mathbb{m}^{(n),-}]^T$, shared interactions $[\mathbb{p}_{\text{sh}}^{(1),+},\dots,\mathbb{p}_{\text{sh}}^{(n),+}]^T$, $[\mathbb{m}_{\text{sh}}^{(1),-},\dots,\mathbb{m}_{\text{sh}}^{(n),-}]^T$ and non-shared interactions $[\mathbb{p}_{\text{non-sh}}^{(1),+},\dots,\mathbb{p}_{\text{non-sh}}^{(n),+}]^T$, $[\mathbb{m}_{\text{non-sh}}^{(1),-},\dots,\mathbb{m}_{\text{non-sh}}^{(n),-}]^T$, which are extracted from each pair of LLMs. Strength of shared interactions of each k-th order $\mathbb{p}_{\text{sh}}^{(k)}$ and $\mathbb{m}_{\text{sh}}^{(k)}$ are computed in Equation (6) using shared interactions in $\Omega_{\text{sh}}^{\text{and}}$ and $\Omega_{\text{sh}}^{\text{or}}$. Strength of non-shared interactions $\mathbb{p}_{\text{non-sh}}^{(k)}$ and $\mathbb{m}_{\text{non-sh}}^{(k)}$ are computed on non-shared interactions in $\Omega^{\text{and}} \setminus \Omega_{\text{sh}}^{\text{and}}$ and $\Omega^{\text{or}} \setminus \Omega_{\text{sh}}^{\text{or}}$. We find that each prompt produces a specific distribution of interactions across different orders. For some prompts, the LLM predominantly models shared interactions, whereas for others, non-shared interactions dominate. The complexity of the interactions triggered also varies across samples[6].

(1) Shared interactions are more concentrated at lower order, which represent simpler patterns between a few tokens, than non-shared interactions. Figure 4(a) reports the average distribution of interactions over all input prompts. We find that compared to shared interactions, non-shared interactions are more often associated with higher-order interaction patterns. This suggests that relatively simple interactions are more likely to be consistently modeled by different LLMs.

(2) The prediction score $v(\mathbf{x})$ is mainly attributed to shared interactions. Most shared interactions are positive, *i.e.*, they mainly boost the prediction score $v(\mathbf{x})$ of the target token. In comparison, the substantial mutual cancellation among non-shared interactions suggests that these non-shared interactions behave more like noise than meaningful patterns. This phenomenon is consistently observed across different LLMs and datasets, as shown in Figure 4(b).

(3) Shared interactions usually represent a few salient inference patterns, while non-shared interactions usually correspond to numerous noise signals. Figure 4(c) compares the number of shared and non-shared interactions. For the prediction of each specific target token, the number of shared interactions is much smaller (ranging from 11 to 21) than non-shared interactions (ranging from 59 to 69). This figure also compares the average strength of shared interactions with that of non-shared interactions. Shared interactions tend to have much stronger effects on the prediction score than the non-shared interactions.

[6]Please see Appendix D for examples for individual prompts.

4 Related Work

Towards representation similarity across neural networks. A growing body of work has investigated whether independently trained neural networks learn shared representations, using representation-similarity measures such as canonical correlation analysis Raghu et al. (2017) and centered kernal alignment Kornblith et al. (2019). More recently, the platonic representation hypothesis (Huh et al., 2024) suggests that increasingly capable models tend to learn more similar representation geometry. Recent LLM-specific studies have further reported substantial representation similarity across different LLMs (Huang et al., 2025; Son et al., 2025; Ciernik et al., 2025). However, these methods mainly quantify *how similar* two representation spaces are, rather than explicitly disentangling *what* symbolic inference patterns are shared or model-specific. Our framework addresses this gap by decomposing cross-model commonality into identifiable symbolic interaction patterns with directly quantifiable prediction effects.

Towards symbolic understanding of an LLM's inference logic. A large body of work has investigated whether intermediate neural representations can be assigned clear and interpretable meanings, including circuit analysis (Elhage et al., 2021), sparse autoencoders (Huben et al., 2024; Templeton et al., 2026), and J-space (Gurnee et al., 2026). However, the polysemanticity of neurons and internal features remains a major obstacle to moving mechanistic interpretability from empirical observations toward rigorous analytical modeling.

To this end, interaction-based explanation provides a complementary route that does not require assigning a unique semantic meaning to individual neurons or features (Li & Zhang, 2023; Chen et al., 2024). Instead, it characterizes the network through its overall input–output behavior and decomposes a prediction into a sparse set of interaction patterns with explicit input-level semantics and quantifiable effects. Theoretical sparsity and universal-matching results further provide formal guarantees for this decomposition (Ren et al., 2023; 2024a). This formulation enables us to explicitly disentangle interaction patterns that are shared across models from those that are model-specific.

Uncovering the root causes of neural network performance. Critically, the proof of symbolic interaction-based explanations establishes a new lens for interpreting AI models. Interaction patterns capture not only inference logic in each sample, but also the root causes of a DNN's global properties. The interaction complexity correlates exponentially with adversarial vulnerability (Ren et al., 2021) and negatively with learnability and generalization power (Liu et al., 2023; Zhou et al., 2024). Deng et al. (2022) have proved a representation bottleneck in interaction. (Wang et al., 2021) have used interactions to explain adversarial transferability. (Deng et al., 2024) have found that different attribution methods can be unified as specific forms of interaction effect reallocation.

Prior interaction studies have mainly analyzed the representations and learning dynamics of individual neural networks. In contrast, our work extends the analysis from interactions within a single model to whether different LLMs encode *shared symbolic interaction patterns* when making the same prediction.

5 Conclusion and Discussions

In this paper, we find that 16 classic open-weight LLMs[1] learn shared symbolic inference patterns. Despite differences in architecture and parameter scale, diverse LLMs rely on similar interaction patterns to predict the target token given the same input prompt. This phenomenon is particularly prominent for high-confidence predictions and becomes increasingly apparent during model training. Moreover, compared with non-shared interactions, interactions shared by different LLMs tend to be much simpler, more coherent, and they exert stable effects on model outputs with much less positive-negative offset effects.

Latent data distillation or training-data similarity across open LLMs may **not** *be the primary cause of shared interactions.* (1) We find similarity within model families (*e.g.*, the LLaMA family, the Qwen family) is not necessarily stronger than cross-family similarity, despite greater potential for distillation and shared training data within an LLM family. (2) Models released in overlapping time windows also exhibit substantial similarity (*e.g.*, Qwen3-8B and GLM-Z1-9B), undermining the distillation hypothesis. (3) The experiments in Figure 3(c) further supports this claim.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language editing and polishing, literature search and summarization, and developing plotting code with author-provided content. We did not use generative AI tools to independently generate research ideas, conduct experiments, create synthetic datasets, or prove mathematical claims. All AI-assisted content was reviewed and verified by the authors. We take full responsibility for the final content of this work, including any text, claims, code, or artifacts produced with the assistance of generative AI.

REFERENCES

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. Smollm2: When smol goes big – data-centric training of a small language model, 2025. URL https://arxiv.org/abs/2502.02737.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, et al. The falcon series of open language models. *arXiv preprint arXiv:2311.16867*, 2023.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Yamini Bansal, Preetum Nakkiran, and Boaz Barak. Revisiting model stitching to compare neural representations. *Advances in neural information processing systems*, 34:225–236, 2021.

Landon Butler, Abhineet Agarwal, Justin Kang, Yigit Efe Erginbas, Bin Yu, and Kannan Ramchandran. Proxyspex: Inference-efficient interpretability via sparse feature interactions in llms. *Advances in Neural Information Processing Systems*, 38:72306–72340, 2026.

Lu Chen, Siyu Lou, Benhao Huang, and Quanshi Zhang. Defining and extracting generalizable interaction primitives from dnns. In *International Conference on Learning Representations*, volume 2024, pp. 23780–23802, 2024.

Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. Convfinqa: Exploring the chain of numerical reasoning in conversational finance question answering. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pp. 6279–6292, 2022.

Daixuan Cheng, Shaohan Huang, and Furu Wei. Adapting large language models via reading comprehension. In *International Conference on Learning Representations*, volume 2024, pp. 48624–48652, 2024.

Laure Ciernik, Lorenz Linhardt, Marco Morik, Jonas Dippel, Simon Kornblith, and Lukas Muttenthaler. Objective drives the consistency of representational similarity across datasets. In *Proceedings of the 42nd International Conference on Machine Learning*, ICML'25. JMLR.org, 2025.

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

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

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, 2019.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12, 2021.

Team Glm, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Wes Gurnee, Nicholas Sofroniew, Adam Pearce, Mateusz Piotrowski, Isaac Kauvar, Runjin Chen, Anna Soligo, Paul Bogdan, Euan Ong, Rowan Wang, et al. Verbalizable representations form a global workspace in language models. *arXiv preprint arXiv:2607.15495*, 2026.

Yuxuan He, Junpeng Zhang, Lei Cheng, Hongyuan Zhang, and Quanshi Zhang. Technical report: Quantifying and analyzing the generalization power of a dnn. *arXiv preprint arXiv:2505.06993*, 2025.

Youcheng Huang, Chen Huang, Duanyu Feng, Wenqiang Lei, and Jiancheng Lv. Cross-model transferability among large language models on the platonic representations of concepts. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3686–3704, 2025.

Robert Huben, Hoagy Cunningham, Logan Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations*, volume 2024, pp. 7827–7845, 2024.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. Position: The platonic representation hypothesis. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20617–20642. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/huh24a.html.

Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 4, 2025.

Justin S Kang, Yigit E Erginbas, Landon Butler, Ramtin Pedarsani, and Kannan Ramchandran. Learning to understand: Identifying interactions via the möbius transform. *Advances in Neural Information Processing Systems*, 37:46160–46202, 2024.

Justin Singh Kang, Landon Butler, Abhineet Agarwal, Yigit Efe Erginbas, Ramtin Pedarsani, Kannan Ramchandran, and Bin Yu. Spex: Scaling feature interaction explanations for llms. *arXiv preprint arXiv:2502.13870*, 2025.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMlR, 2019.

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

Dongrui Liu, Huiqi Deng, Xu Cheng, Qihan Ren, Kangrui Wang, and Quanshi Zhang. Towards the difficulty for a deep neural network to learn concepts of different complexities. *Advances in Neural Information Processing Systems*, 36:41283–41304, 2023.

Pekka Malo, Ankur Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65(4):782–796, 2014.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

Qwen, :, 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, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL `https://arxiv.org/abs/2412.15115`.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in neural information processing systems*, 30, 2017.

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

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, 2023.

Qihan Ren, Jiayang Gao, Wen Shen, and Quanshi Zhang. Where we have arrived in proving the emergence of sparse interaction primitives in dnns. In *The Twelfth International Conference on Learning Representations*, 2024a.

Qihan Ren, Junpeng Zhang, Yang Xu, Yue Xin, Dongrui Liu, and Quanshi Zhang. Towards the dynamics of a dnn learning symbolic interactions. *Advances in Neural Information Processing Systems*, 37:50653–50688, 2024b.

Gian-Carlo Rota. On the foundations of combinatorial theory: I. theory of möbius functions. In *Classic Papers in Combinatorics*, pp. 332–360. Springer, 1964.

Agam Shah, Abhinav Gullapalli, Ruchit Vithani, Michael Galarnyk, and Sudheer Chava. Finer-ord: financial named entity recognition open research dataset. *arXiv preprint arXiv:2302.11157*, 2023.

Ankur Sinha and Tanmay Khandait. Impact of news on the commodity market: Dataset and results. In *Future of Information and Communication Conference*, pp. 589–601. Springer, 2021.

Ankur Sinha, Satishwar Kedas, Rishu Kumar, and Pekka Malo. Sentfin 1.0: Entity-aware sentiment analysis for financial news. *Journal of the Association for Information Science and Technology*, 73(9):1314–1335, 2022.

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, 2013.

Daniel Son, Sanjana Rathore, Andrew Rufail, Adrian Simon, Daniel Zhang, Soham Dave, Cole Blondin, Kevin Zhu, and Sean O'Brien. Semantic convergence: Investigating shared representations across scaled llms. *arXiv preprint arXiv:2507.22918*, 2025.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *arXiv preprint arXiv:2605.29358*, 2026.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Xin Wang, Jie Ren, Shuyun Lin, Xiangming Zhu, Yisen Wang, and Quanshi Zhang. A unified approach to interpreting and boosting adversarial transferability. In *International Conference on Learning Representations*, 2021.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

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.

A SPARSITY PROPERTY OF INTERACTIONS

Ren et al. (2024a) showed that a DNN tends to encode only a small number of salient AND-OR interactions when the following three conditions hold. Under these conditions, most possible interactions have negligible effects, and the remaining sparse interactions can faithfully approximate the network output over all masked samples $\{\mathbf{x}_S \mid S \subseteq N\}$.

1. **Bounded interaction order.** The network does not rely on extremely high-order interactions. That is, interactions involving too many input variables have zero or negligible contribution to the output. Formally, there exists an order threshold M such that the interaction effect $I(S)$ vanishes for any subset S with $|S| > M$.
2. **Monotonic response under masking.** When more input variables are masked, the average network response decreases monotonically. Specifically, let $\bar{u}^{(m)}$ denote the average output change when m variables are revealed, compared with the fully masked baseline:

   $$\bar{u}^{(m)} = \mathbb{E}_{S\subseteq N,|S|=m}\left[v(\mathbf{x}_S) - v(\mathbf{x}_\emptyset)\right].$$

   Then, for $m' < m$, the average response satisfies

   $$\bar{u}^{(m')} \leq \bar{u}^{(m)}.$$

   This condition implies that adding more input variables, on average, provides more evidence for the model prediction.
3. **Polynomial lower bound on average response.** The average response does not decay too sharply when fewer variables are revealed. In particular, for any $m' < m$, there exists a positive constant $p > 0$ such that

   $$\bar{u}^{(m')} \geq \left(\frac{m'}{m}\right)^p \bar{u}^{(m)}.$$

   This polynomial lower bound rules out the case where the model output is dominated by dense, extremely high-order interactions.

Together, these conditions imply that the network output is mainly governed by a limited number of lower-order interactions, while most high-order interactions have negligible effects. This provides theoretical support for the sparsity assumption used in our interaction-based analysis.

Empirical verification Given an input sample, the previous subsection summarizes the theoretical conditions under which sparse interactions emerge. We now empirically examine whether such sparsity also appears in LLMs.

For each model-dataset setting, we randomly sample 20 prompts and extract the AND-OR interactions used for target-token prediction. Following Ren et al. (2024a), we normalize each interaction strength by the largest absolute interaction strength under the same model-dataset setting. We then collect all extracted interactions from the sampled prompts, sort them in descending order according to their absolute strengths, and visualize the resulting distribution.

Figure 5 shows the sorted interaction-strength curves for multiple model-dataset settings, including DeepSeek-R1-Distill-LLaMA-8B, Falcon-7B, and LLaMA3-8B on the AdaptLLM finance tasks dataset, as well as GLM-Z1-9B, Qwen2-7B, and Qwen2.5-14B on the WikiText dataset. Across all settings, the curves exhibit a sharp drop near the beginning and quickly approach zero. This indicates that only a very small fraction of interactions have salient effects, while the vast majority contribute negligibly to the prediction score.

These results provide empirical support for the sparsity assumption used in our analysis. Although the number of possible interactions grows exponentially with the number of input variables, LLM predictions are dominated by a compact set of high-strength interactions in practice. Therefore, the extracted AND-OR interactions provide a sparse and tractable representation of the model's inference behavior.

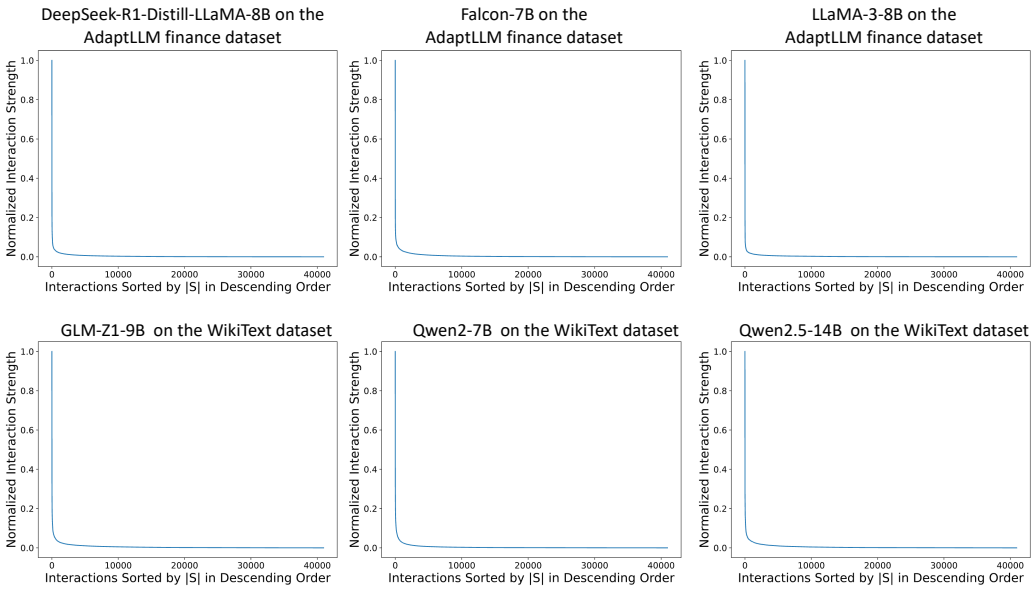

Figure 5: Sparsity of extracted interactions across different model-dataset settings. For each setting, we merge the interactions extracted from all selected samples, normalize their strengths, and sort them by absolute magnitude in descending order. Only a small fraction of interactions have large strengths, whereas most interaction effects rapidly decay to values close to zero. This demonstrates that the extracted interactions are highly sparse in practice.

B Möbius Transform and Logical-Model Reconstruction

This section explains how the interaction effects in the logical model $\phi(\cdot)$ can be derived from the Möbius transform, and how these interactions reconstruct the model output on arbitrary masked states.

Let $N = \{1, \ldots, n\}$ denote the set of input variables, and let $\mathbf{x}_S$ denote the masked input where variables in $S \subseteq N$ are kept and all other variables are masked. For any set function f defined on masked inputs, its Möbius transform is given by

$$\mathcal{M}_f(S) = \sum_{L \subseteq S} (-1)^{|S|-|L|} f(\mathbf{x}_L). \tag{10}$$

The coefficient $\mathcal{M}_f(S)$ measures the irreducible effect of the variable subset S, after subtracting the effects that can already be explained by its smaller subsets.

In the AND–OR logical model, the prediction function is decomposed into two parts,

$$\forall T \subseteq N \quad v(\mathbf{x}_T) = v^{\text{and}}(\mathbf{x}_T) + v^{\text{or}}(\mathbf{x}_T). \tag{11}$$

The AND interaction effect is defined as the Möbius coefficient of v^{and}:

$$I_S^{\text{and}} = \sum_{L \subseteq S} (-1)^{|S|-|L|} v^{\text{and}}(\mathbf{x}_L). \tag{12}$$

For OR interactions, we apply the Möbius transform to the reversed set function $\widehat{v}^{\text{or}}$, where

$$\widehat{v}^{\text{or}}(\mathbf{x}_S) = v^{\text{or}}(\mathbf{x}_{N \setminus S}). \tag{13}$$

Then the OR interaction effect is defined as

$$I_S^{\text{or}} = -\sum_{L \subseteq S} (-1)^{|S|-|L|} v^{\text{or}}(\mathbf{x}_{N \setminus L}). \tag{14}$$

The negative sign makes the OR interaction activated when at least one variable in S is present in the masked input.

We next show that these interaction effects reconstruct the logical model. For a masked sample $\mathbf{x}_T$, an AND interaction S is activated only when $S \subseteq T$. Therefore,

$$\begin{aligned}\sum_{S\subseteq N} I_S^{\text{and}} \mathbb{1}_{\text{and}}(S|\mathbf{x}_T) &= \sum_{S\subseteq T, S\neq\emptyset} I_S^{\text{and}} \\ &= \sum_{S\subseteq T} I_S^{\text{and}} - v^{\text{and}}(\mathbf{x}_\emptyset). \end{aligned} \tag{15}$$

Substituting the definition of I_S^{and}, we have

$$\begin{aligned}\sum_{S\subseteq T} I_S^{\text{and}} &= \sum_{S\subseteq T}\sum_{L\subseteq S} (-1)^{|S|-|L|} v^{\text{and}}(\mathbf{x}_L) \\ &= \sum_{L\subseteq T} v^{\text{and}}(\mathbf{x}_L) \sum_{S: L\subseteq S\subseteq T} (-1)^{|S|-|L|}. \end{aligned} \tag{16}$$

The inner summation is zero for $L \neq T$ and one for $L = T$. Hence,

$$\sum_{S\subseteq N} I_S^{\text{and}} \mathbb{1}_{\text{and}}(S|\mathbf{x}_T) = v^{\text{and}}(\mathbf{x}_T) - v^{\text{and}}(\mathbf{x}_\emptyset). \tag{17}$$

Similarly, an OR interaction S is activated when $S \cap T \neq \emptyset$. Thus,

$$\begin{aligned}&\sum_{S\subseteq N} I_S^{\text{or}} \mathbb{1}_{\text{or}}(S|\mathbf{x}_T) \\ &= \sum_{S\cap T\neq\emptyset, S\neq\emptyset} I_S^{\text{or}} \\ &= -\sum_{S\cap T\neq\emptyset, S\neq\emptyset} \sum_{L\subseteq S} (-1)^{|S|-|L|} v^{\text{or}}(\mathbf{x}_{N\setminus L}) \\ &= -\sum_{L\subseteq N} v^{\text{or}}(\mathbf{x}_{N\setminus L}) \sum_{S: S\cap T\neq\emptyset,\, S\supseteq L} (-1)^{|S|-|L|}. \end{aligned} \tag{18}$$

By the inclusion–exclusion identity, all terms cancel except the terms corresponding to $L = N$ and $L = N \setminus T$. Therefore,

$$\sum_{S\subseteq N} I_S^{\text{or}} \mathbb{1}_{\text{or}}(S|\mathbf{x}_T) = v^{\text{or}}(\mathbf{x}_T) - v^{\text{or}}(\mathbf{x}_\emptyset). \tag{19}$$

Finally, let the bias term be

$$b_0 = v(\mathbf{x}_\emptyset). \tag{20}$$

Then the logical model is reconstructed as

$$\begin{aligned}\phi(\mathbf{x}_T) &= \sum_{S\subseteq N} I_S^{\text{and}} \mathbb{1}_{\text{and}}(S|\mathbf{x}_T) \\ &\quad + \sum_{S\subseteq N} I_S^{\text{or}} \mathbb{1}_{\text{or}}(S|\mathbf{x}_T) + b_0 \\ &= v^{\text{and}}(\mathbf{x}_T) - v^{\text{and}}(\mathbf{x}_\emptyset) \\ &\quad + v^{\text{or}}(\mathbf{x}_T) - v^{\text{or}}(\mathbf{x}_\emptyset) + v(\mathbf{x}_\emptyset) \\ &= v(\mathbf{x}_T) - v(\mathbf{x}_\emptyset) + v(\mathbf{x}_\emptyset) \\ &= v(\mathbf{x}_T). \end{aligned} \tag{21}$$

Thus, the AND-OR interactions, together with the bias term, reconstruct the model output on any masked state $\mathbf{x}_T$. This shows that the interaction effects can be understood as Möbius components of the prediction function, while the logical model $\phi(\cdot)$ corresponds to the inverse reconstruction from these components.

Table 1: Licenses of the models and datasets used in our experiments.

Resource	Models / Dataset	License
Qwen family	Qwen-7B	Tongyi Qianwen License Agreement
Qwen family	Qwen2-7B, Qwen2.5-7B/14B, Qwen3-8B, Qwen3.5-0.8B	Apache-2.0
LLaMA family	LLaMA-7B	Meta non-commercial research license
LLaMA family	LLaMA2-7B, LLaMA3-8B	Meta community license agreements
Other LLMs	Falcon-7B, SmolLM2-360M	Apache-2.0
Other LLMs	GLM-Z1-9B-0414, GLM-4-32B-0414, Phi-4, DeepSeek-R1-Distill-LLaMA-8B	MIT
Other LLMs	Gemma-3-12B	Gemma Terms of Use
Dataset	AdaptLLM finance tasks	No unified license; see original datasets
Dataset	WikiText-103	CC BY-SA 3.0 / GFDL

C Experimental Details

C.1 Models and Datasets

Evaluation on open-weight LLMs. We evaluate 16 open-weight LLMs spanning different model families, training data, and parameter scales, ranging from 360M to 32B parameters. The evaluated models include Gemma3-12B (Kamath et al., 2025), DeepSeek-R1-Distill-LLaMA-8B (Guo et al., 2025), Falcon-7B (Almazrouei et al., 2023), Phi-4 (Abdin et al., 2024), and SmolLM2-360M (Allal et al., 2025). Besides, we include Qwen-7B, Qwen2-7B, Qwen2.5-7B, Qwen2.5-14B, Qwen3-8B, and Qwen3.5-0.8B from the Qwen family (Bai et al., 2023; Qwen et al., 2025; Yang et al., 2025; Qwen Team, 2026), GLM-Z1-9B and GLM-4-32B from the GLM family (Glm et al., 2024), and LLaMA-7B, LLaMA2-7B, and LLaMA3-8B from the LLaMA family (Touvron et al., 2023a;b; Grattafiori et al., 2024). The licenses of these models are summarized in Table 1.

For evaluation, we build sentence-level next-token prediction examples from two public text sources, with 50 samples from each dataset. Although this sample size is limited, it is sufficient for our initial exploratory analysis, because each prompt provided many extracted interactions, *i.e.*, $80 \times 50 = 4000$, providing a rich characterization of the interaction distribution. Our goal is not to build a large-scale benchmark, but to examine whether cross-model consistency in interaction patterns can be observed under a controlled setting. In the finance-domain setting, we use the AdaptLLM finance tasks (Cheng et al., 2024), which collect several financial NLP datasets, including ConvFinQA (Chen et al., 2022), FPB (Malo et al., 2014), FiQA-SA (Sinha & Khandait, 2021), Headline (Sinha et al., 2022), and NER (Shah et al., 2023). In the general-domain setting, we use WikiText-103 (Merity et al., 2016). For both sources, we split the raw text into sentences and retain sentences with sufficient length and semantic content. Each example is constructed by using a sentence prefix as the input prompt. The target token is then selected according to the prediction scores of the paired LLMs, ensuring that both models are evaluated on the same target-token prediction. This setup enables us to analyze interaction patterns in controlled sentence-level contexts across both financial and general-domain text.

Due to the computational cost of pairwise interaction extraction, we do not evaluate all possible model pairs. Instead, we select representative pairs to cover two types of comparisons. First, we compare models within the same family, such as Qwen and LLaMA, to examine how interaction patterns change across model generations and parameter scales. Second, we compare models from different families, such as Qwen, LLaMA, DeepSeek, GLM, Falcon, Phi, Gemma and SmolLM2, to test whether shared interactions also emerge across different architectures and training recipes. This design allows us to analyze both within-family consistency and cross-family convergence while keeping the computation tractable.

Evaluation of similarity changes during training. We train two BERT-large models on the Stanford Sentiment Treebank (SST-2), a binary sentiment classification dataset of movie reviews. To

Table 2: Release time of the evaluated open-weight language models. Models are ordered chronologically by their public release dates.

Model	Parameter scale	Release time
LLaMA-7B	7B	Feb. 2023
Falcon-7B	7B	May 2023
LLaMA2-7B	7B	Jul. 2023
Qwen-7B	7B	Aug. 2023
LLaMA3-8B	8B	Apr. 2024
Qwen2-7B	7B	Jun. 2024
Qwen2.5-7B	7B	Sep. 2024
Qwen2.5-14B	14B	Sep. 2024
SmolLM2-360M	360M	Nov. 2024
Phi-4	14B	Dec. 2024
DeepSeek-R1-Distill-LLaMA-8B	8B	Jan. 2025
Gemma3-12B	12B	Mar. 2025
GLM-Z1-9B-0414	9B	Apr. 2025
GLM-4-32B-0414	32B	Apr. 2025
Qwen3-8B	8B	Apr. 2025
Qwen3.5-0.8B	0.8B	Mar. 2026

create a controlled setting, we randomly partition the original training set into two balanced, non-overlapping subsets with matched label distributions. The two subsets are then used to train the two models separately from different random initializations. This setup allows us to examine whether independently initialized models trained on different data gradually develop more similar interaction patterns during learning, while minimizing confounding effects from differences in the overall training-data distribution.

Evaluation of interaction similarity corresponding to samples with different next-token prediction confidence level. For each test example, let $\mathbf{x}_i$ denote the input context and y_i the target token. We define model m's target-token prediction confidence as $c_{m,i} = p_m(y_i \mid x_i)$, *i.e.*, the softmax probability assigned to the target-token. For each model pair (A, B), we define the pair-level confidence as $c_{A,B,i} = \min(c_{A,i}, c_{B,i})$, so that an example is considered high-confidence only when both models assign high probability to the target token. Because confidence scales may differ substantially across models, we perform the split independently within each model pair rather than using a global confidence threshold. Specifically, the bottom and top quartiles of $c_{A,B,i}$ form the low- and high-confidence groups, respectively.

For every example, we retain the $K = 80$ interactions with the largest absolute interaction strength for each model and compute the interaction similarity κ. We compare the average $\bar{\kappa}$ between the high- and that between low-confidence groups and summarize the within-pair confidence effect as

$$\Delta\kappa = \mathbb{E}[\bar{\kappa} \mid \text{High}] - \mathbb{E}[\bar{\kappa} \mid \text{Low}] .$$

Positive $\Delta\kappa$ indicates that high-confidence predictions exhibit stronger interaction similarity.

C.2 Extraction of interactions

Selecting input variables. Given an input prompt with n input variables, extracting AND-OR interactions requires evaluating the model on 2^n masked samples. In this paper, we define input variables at the word level, rather than the token level, to facilitate semantic analysis. However, real-world prompts usually contain many words, making exhaustive interaction extraction computationally infeasible. Following Chen et al. (2024), we therefore select a subset of words as input variables and treat the remaining words as a fixed background. This allows us to extract AND-OR interactions among the selected variables while keeping the computation manageable.

Specifically, we select 10 words from the prompt $\mathbf{x}$ of each sample. For prompts containing more than 10 words, we remove words with weak semantic content, such as articles, prepositions, and conjunctions, before selecting the variables.

Extracting interactions. The inference logic of neural networks is often too complex to be faithfully and compactly represented by only one type of interaction, either AND or OR. To address this issue, Zhou et al. (2024) decomposed the network output $v(\mathbf{x}_T)$ into two components: an AND component $u_T^{\text{and}} = 0.5v(\mathbf{x}_T) + \gamma_T$, and an OR component $u_T^{\text{or}} = 0.5v(\mathbf{x}_T) - \gamma_T$, where $\{\gamma_T\}$ are learnable parameters. Thus, finding an appropriate AND-OR decomposition of v is equivalent to learning the optimal values of $\{\gamma_T\}$, where $\gamma_T \in \mathbb{R}$.

Following Zhou et al. (2024) and Chen et al. (2024), we learn $\{\gamma_T\}$ by minimizing the L_1 norm of both AND and OR interaction effects, encouraging the sparsest possible AND-OR explanation:

$$\min_{\{\gamma_T\}} \|\mathbf{I}_{\text{and}}\|_1 + \|\mathbf{I}_{\text{or}}\|_1, \tag{22}$$

where $\mathbf{I}_{\text{and}} = [I_{T_1}^{\text{and}}, \ldots, I_{T_{2^n}}^{\text{and}}]^\top$ and $\mathbf{I}_{\text{or}} = [I_{T_1}^{\text{or}}, \ldots, I_{T_{2^n}}^{\text{or}}]^\top$, with $T_k \subseteq N$.

Algorithm 1 provides the pseudocode for the above procedure.

Defining interactions. For simplicity, we refer to the top-K salient subsets of input variables as interactions in our analysis. We use $K = 80$ for main results, and present results with different K values in Appendix D. Specifically, after computing the AND-OR interaction effects for all candidate subsets, we rank them by their absolute effect magnitudes and retain only the subsets with salient effects. These selected subsets are treated as the effective interaction patterns used by the model for the target-token prediction, while subsets with near-zero effects are regarded as negligible. This convention is consistent with the sparsity property of interaction-based explanations, where only a small fraction of all possible variable combinations contribute substantially to the model output.

C.3 Computational Resources

Experiments were conducted on a Linux server equipped with NVIDIA A800 80GB PCIe GPUs and two Intel(R) Xeon(R) Platinum 8352V CPUs @ 2.10GHz. The server has 36 CPU cores per socket, 2 sockets, and 2 threads per core, totaling 144 CPU threads.

For each LLM, masked-sample inference was performed on a single GPU. Given $n = 10$ selected input variables, each prompt requires $2^{10} = 1024$ forward passes, which we evaluate with a batch size of 512. Thus, the runtime mainly depends on the inference speed of each model. After obtaining the model outputs on masked samples, the optimization for learning AND-OR interaction effects is performed on CPU and usually takes 30-60 seconds per sample.

Package versions. We implemented model inference using PyTorch and HuggingFace Transformers. The main package versions used in our experiments are PyTorch 2.12.0+cu126 and Transformers 5.17.0.

Remarks on computational cost. Similar to Shapley-value estimation, exact interaction extraction has an exponential worst-case cost because it requires evaluating the model on masked variants of the input. In practice, however, this cost can be reduced substantially. Existing methods exploit the fact that only a limited number of interactions are usually salient: for example, the Sparse Möbius Transform can reduce the complexity to $O(nK \log n)$ when there are only K important interactions (Kang et al., 2024); Spectral Explainer targets important interactions in long-sequence LLM inputs (Kang et al., 2025); and ProxySPEX further improves efficiency by using lightweight proxy models to approximate the local behavior of the original model (Butler et al., 2026). In addition, interactions for LLMs need not always be computed at the token level. Depending on the analysis goal, variables can be defined as words, phrases, or sentences, which greatly reduces the effective input dimension. For studies such as ours, where the goal is to estimate the distribution and consistency of salient interactions rather than enumerate every possible interaction, it is often sufficient to compute interactions over a small set of semantically meaningful variables. Therefore, although exact extraction is expensive in theory, sparsity-aware algorithms, flexible variable granularity, and engineering optimizations make interaction-based analysis feasible in practical LLM settings.

Algorithm 1 Compute AND/OR interactions and select salient interactions

Require: A deep neural network v, input sample $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$, indexed by $N = \{1, 2, \ldots, n\}$, noise threshold ζ, salience threshold τ, which equals to the K-th largest absolute effect across both AND and OR interactions, convergence threshold δ, baseline value $\mathbf{b}_{\text{base}} = [b_1, b_2, \ldots, b_n]^T$.

Ensure: AND interaction I_T^{and}, OR interaction I_T^{or}, and salient interaction sets Ω_{and} and Ω_{or}.

Initialize learnable parameters $\{\gamma_L\}$ and $\{\epsilon_L\}$ for all $L \subseteq N$.
Compute output $v(\mathbf{x}_\emptyset)$, where $\mathbf{x}_\emptyset$ is the masked sample with all variable values replaced by baseline values $\mathbf{b}_{\text{base}}$, *i.e.*, $v(\mathbf{x}_\emptyset) = v(\mathbf{b}_{\text{base}})$.
Initialize previous loss $\mathcal{L}_{\text{prev}} \leftarrow \infty$.
repeat
    for each subset $L \subseteq N$ **do**
        Compute masked sample $\mathbf{x}_L$ by replacing the values of variables not in L with their baseline values.
        Compute network output $v(\mathbf{x}_L)$.
        Comput noise term ϵ_L constrained in $[-\zeta, \zeta]$, where $\zeta = 0.01 \cdot |v(\mathbf{x}) - v(\mathbf{x}_\emptyset)|$.
        Decompose $v(\mathbf{x}_L)$ into AND and OR components:
        $u_L^{\text{and}} \leftarrow 0.5 \cdot (v(\mathbf{x}_L) - \epsilon_L) + \gamma_L$
        $u_L^{\text{or}} \leftarrow 0.5 \cdot (v(\mathbf{x}_L) - \epsilon_L) - \gamma_L$
    end for
    for each subset $T \subseteq N$ **do**
        Compute scalar weight for AND interaction I_T^{and}:

$$I_T^{\text{and}} \leftarrow \sum_{L \subseteq T} (-1)^{|T|-|L|} u_L^{\text{and}}$$

        Compute scalar weight for OR interaction I_T^{or}:

$$I_T^{\text{or}} \leftarrow -\sum_{L \subseteq T} (-1)^{|T|-|L|} u_{N \setminus L}^{\text{or}}$$

    end for
    Compute current loss $\mathcal{L} \leftarrow \sum_{T \subseteq N} (|I_T^{\text{and}}| + |I_T^{\text{or}}|)$.
    Optimize parameters $\{\gamma_L\}$ and $\{\epsilon_L\}$ to minimize $\mathcal{L}$.
    Check for convergence: $|\mathcal{L} - \mathcal{L}_{\text{prev}}| < \delta$
    Update previous loss: $\mathcal{L}_{\text{prev}} \leftarrow \mathcal{L}$.
until convergence
Select salient AND interactions:

$$\Omega_{\text{and}} \leftarrow \{T \subseteq N : |I_T^{\text{and}}| \geq \tau\}$$

Select salient OR interactions:

$$\Omega_{\text{or}} \leftarrow \{T \subseteq N : |I_T^{\text{or}}| \geq \tau\}$$

return $I_T^{\text{and}}, I_T^{\text{or}}, \Omega_{\text{and}}$ and Ω_{or}.

D MORE EXPERIMENTAL RESULTS

Interaction distribution on individual prompts. Figure 6 presents additional prompt-level examples of interaction distributions for different LLM pairs. For each prompt, we group the extracted interaction effects by order and separately show positive and negative effects for all interactions, and shared interactions. These examples illustrate that the interaction distribution is input-dependent. Some prompts activate relatively strong shared interactions at low orders, while others contain more non-shared interactions and stronger positive-negative cancellation. This provides a more fine-grained view of the sample-level variation behind the averaged results reported in the main manuscript as shown in Figure 4.

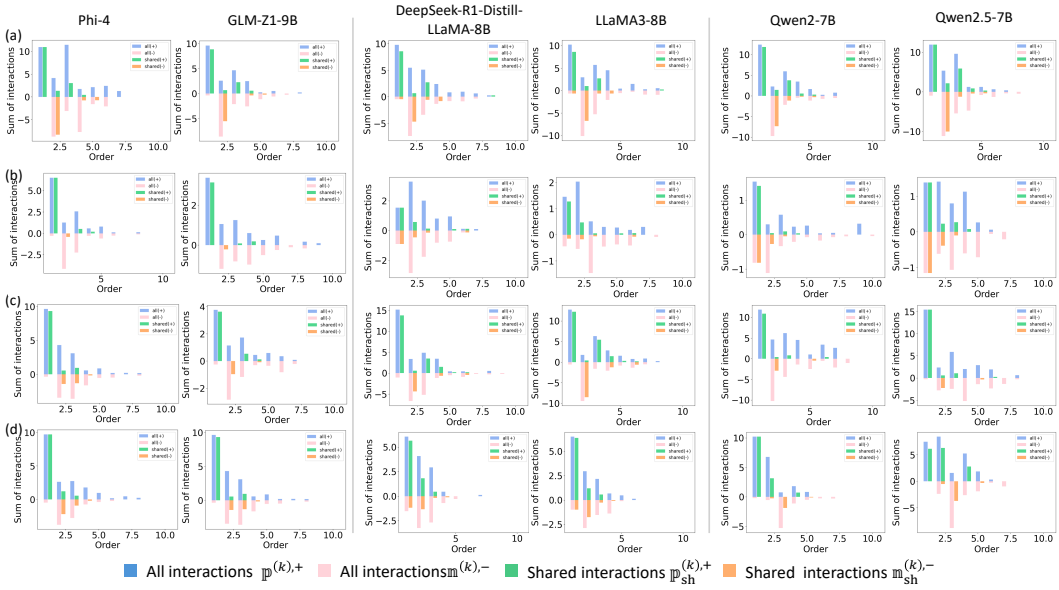

Figure 6: Prompt-level interaction distributions for different LLM pairs. Rows (a,b) show examples from the AdaptLLM finance tasks, and rows (c,d) show examples from WikiText. For each prompt, interaction effects are grouped by order and separated into all interactions and shared interactions, with positive and negative effects plotted separately.

Table 3: Ablation on the number of evaluation prompts. The last column reports the 95% confidence interval estimated using 100 prompts.

Model pair	50 prompts	100 prompts	95% CI (100 prompts)
LLaMA3-8B vs. GLM-Z1-9B	0.724	0.740	[0.695, 0.782]
Qwen3-8B vs. GLM-Z1-9B	0.710	0.750	[0.702, 0.793]
LLaMA-7B vs. LLaMA2-7B	0.729	0.764	[0.694, 0.777]
Qwen3-8B vs. LLaMA3-8B	0.712	0.745	[0.703, 0.792]

Ablation on the number of test samples. We examine whether the estimated interaction similarity κ is sensitive to the number of evaluation prompts. When increasing the number of prompts from 50 to 100, the resulting κ values remain highly consistent across model pairs, with Pearson correlation coefficients of 0.986 on the AdaptLLM Finance dataset and 0.983 on WikiText. Table 3 reports representative results for four model pairs. Notably, all estimates obtained using 50 prompts fall within the corresponding 95% confidence intervals estimated using 100 prompts, indicating that the main results are stable with respect to the number of evaluation samples.

We further increase the number of prompts to 500 for the Qwen3-8B–LLaMA3-8B pair and obtain $\kappa = 0.751$, which remains consistent with the 50- and 100-prompt estimates.

Ablation on the number of selected input words. We further examine the sensitivity of interaction similarity to the number and random selection of input variables. We consider $n \in \{8, 10\}$ selected input words. For each value of n, we independently sample two sets of input variables and recompute the interaction similarity κ. Table 4 reports the results for three representative model pairs. The estimated similarities remain comparable across both different values of n and different random selections of input words, suggesting that the observed cross-model interaction similarity is not sensitive to a particular choice of input variables.

Ablation on the number of extracted salient interactions. We further examine whether the estimated interaction similarity is sensitive to the number of salient interactions extracted in the analysis. Specifically, we vary $K \in \{60, 80, 100\}$ and recompute κ for representative model pairs on both

Table 4: Ablation on the number and random selection of input variables. For each n, two independently sampled sets of input words are evaluated.

Model pair	$n = 8$		$n = 10$	
	Set 1	Set 2	Set 1	Set 2
Qwen3-8B vs. LLaMA3-8B	0.7523	0.7310	0.7370	0.7471
Gemma3-12B vs. GLM-Z1-9B	0.7236	0.6915	0.7077	0.7208
LLaMA3-8B vs. Falcon-7B	0.7311	0.7588	0.7578	0.7546

Table 5: Ablation on the number of retained salient interactions K.

Dataset	Model pair	$K = 60$	$K = 80$	$K = 100$
AdaptLLM	LLaMA-7B vs. LLaMA2-7B	0.729	0.729	0.718
	Qwen3-8B vs. LLaMA3-8B	0.717	0.712	0.710
WikiText	LLaMA-7B vs. LLaMA2-7B	0.814	0.819	0.815
	Qwen3-8B vs. LLaMA3-8B	0.781	0.774	0.772

AdaptLLM Finance and WikiText. As shown in Table 5, the resulting values remain highly consistent across different choices of K. For example, the variation is within 0.011 for LLaMA-7B vs. LLaMA2-7B on AdaptLLM and within 0.009 for Qwen3-8B vs. LLaMA3-8B on WikiText. These results indicate that the observed cross-model interaction similarity is not sensitive to the specific number of retained salient interactions.

Interaction similarity between LLMs on WikiText. Figure 7 visualizes the pairwise interaction similarity κ among LLMs on the WikiText dataset, while Figure 8 compares the characteristics of shared and non-shared interactions. These results show that the observed interaction convergence also generalizes to general-domain language modeling, rather than being specific to the finance-domain setting.

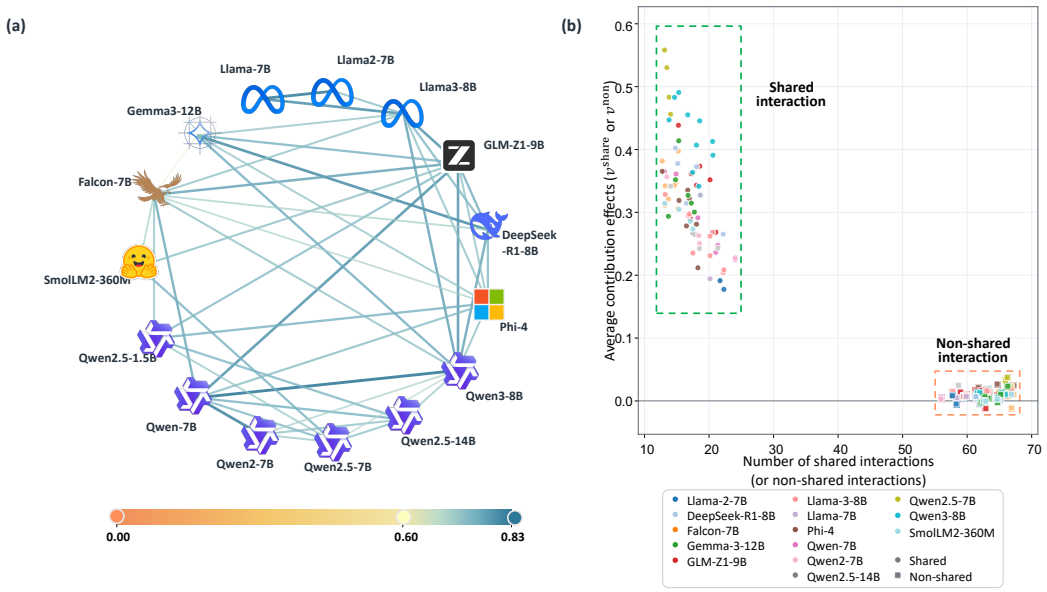

Figure 7: (a) Interaction similarity κ among LLMs on WikiText dataset. Each node denotes an LLM, and each edge represents the fraction of prediction utility explained by shared interactions between two models. (b) Comparison of the number and average contribution effect of shared interactions and non-shared interactions. Each dot represents a specific LLM.

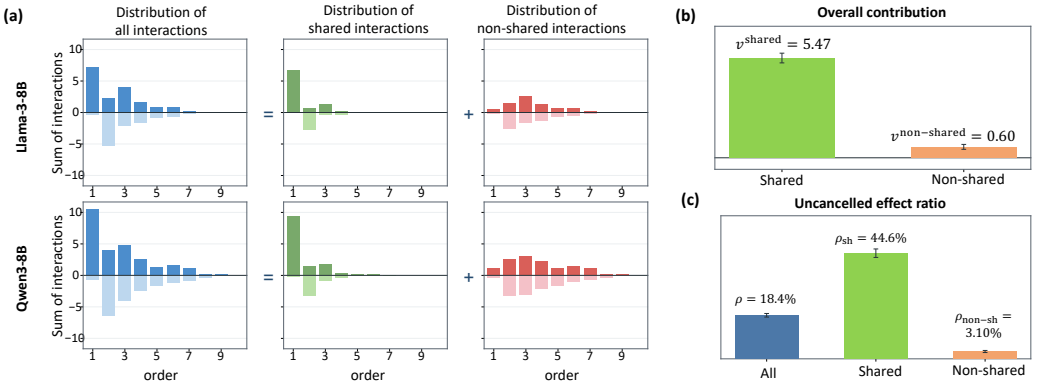

Figure 8: Characteristics of shared and non-shared interactions on WikiText dataset. (a) For representative model pairs, the distribution of all interactions over different orders can be decomposed into shared and non-shared components. (b) Most non-shared interactions exhibit significant mutual cancellation of the positive and negative effects. This makes the overall contribution of shared interactions $v^{\text{shared}}(\mathbf{x})$ much greater than the contribution of non-shared interactions $v^{\text{non}}(\mathbf{x})$. (c) We also report the higher uncancelled-effect ratio for shared interactions ρ_{sh} and the lower uncancelled-effect ratio for non-shared interactions $\rho_{\text{non-sh}}$.