Seeing to Generalize: How Visual Data Corrects Binding Shortcuts

Anonymous Authors[1]

Abstract

We document a phenomenon in which Vision Language Models (VLMs) outperform their underlying LLMs on purely text-only tasks, particularly in long-context information retrieval. To investigate this effect, we build a controlled synthetic retrieval task and find that a transformer trained only on text achieves perfect in-distribution accuracy but fails to generalize out of distribution, while subsequent training on an image-tokenized version of the same task nearly doubles text-only OOD performance. Using interchange interventions, attention knockouts, and linear probes, we causally identify the mechanism underlying this improvement: text-only training converges to positional binding—a shortcut that exploits token positions—whereas image-based training disrupts this shortcut, in part through the translation invariance inherent in visual inputs, forcing the model to adopt symbolic binding based on semantic content. We characterize the circuit-level implementations of each mechanism, identifying a binding signature—a marked surge in attribute decodability at entity positions—that distinguishes explicit from implicit binding and generalizes to large-scale pretrained models. Our results demonstrate how mechanistic interpretability tools can causally link cross-modal training to learned computational strategies, and suggest that visual supervision acts as a mechanism-level regularizer that promotes robust binding in language models.

1. Introduction

A central challenge for neural sequence models is the *binding problem*: how to reliably associate related pieces of information across a long context (Treisman, 1996; Feng & Steinhardt, 2024). Recent work has shown that transformers implement distinct binding mechanisms—positional, symbolic, and reflexive—and that the choice of mechanism has profound implications for generalization (Gur-Arieh et al., 2025; Wu et al., 2025; Urrutia et al., 2025). Understanding how and why models adopt different binding strategies is a fundamental question for mechanistic interpretability.

In this work, we connect this question to an empirical observation about Vision-Language Models (VLMs). VLMs extend Large Language Models (LLMs) with a vision encoder that maps images into token sequences, enabling multimodal reasoning (Bai et al., 2023; Liu et al., 2023). Surprisingly, VLMs systematically outperform their underlying LLMs on purely text-only information retrieval tasks, particularly in long-context settings. For instance, on the Indirect Retrieval task (averaged across context lengths), Qwen3-VL-8B achieves 63.5% accuracy, compared to 48.2% for its base model Qwen3-8B. This is puzzling: why should training on image-based tasks improve performance on text-based evaluations?

To answer this question, we designed controlled experiments centered on a text-only retrieval task (shown in Figure 1). We trained a small transformer model on contexts containing up to eight shapes and found that, while it achieved perfect in-distribution accuracy, its out-of-distribution (OOD) generalization was poor: on contexts with more than eight shapes, accuracy dropped to 37.2%. Crucially, when we continued training on an equivalent task where the textual context was replaced by tokenized images, OOD performance improved to 69.5%.

Using three complementary interpretability tools, we causally identify and characterize the mechanism underlying this improvement. **Interchange interventions** (Gur-Arieh et al., 2025) identify *which* binding mechanism a model employs. **Attention knockouts** (Geva et al., 2023) reveal *which* token-to-token connections are structurally necessary. **Linear probes** (Alain & Bengio, 2017) characterize *when and where* binding information becomes explicitly represented. Together, these tools provide convergent evidence: interchange identifies the computational regime, knockouts reveal the structural pathways, and probes characterize the representational consequences.

We find that text-only training converges to **positional binding**—a shortcut that relies on token positions rather

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

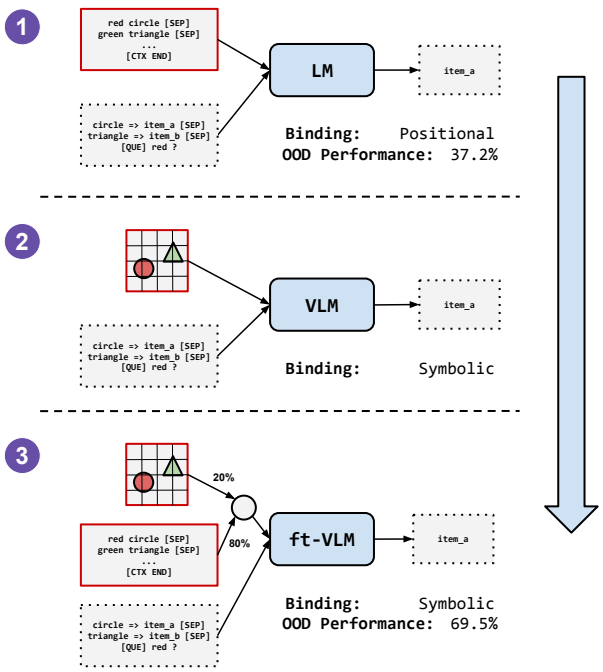


Figure 1. Overview of the training pipeline, showing the progression from text-only training to vision–language training.

than semantic content. Visual training disrupts this shortcut through the translation invariance inherent in images, forcing the model to adopt **symbolic binding**—a position-agnostic strategy that matches attributes to entities by semantic identity. This shift persists even after text-only examples are reintroduced. We further characterize how binding strategies vary across training regimes and visual encoders, demonstrate an analogous shift in pretrained LLM-to-VLM transitions, and provide three falsifiable hypotheses with supporting and contradictory evidence (Section 4).

Our primary contribution is a mechanistic analysis: we causally identify how cross-modal training reshapes a model's internal binding computation, characterize the circuits that implement each strategy, and show that the same shift occurs in large-scale pretrained models. Our findings suggest that training recipes could explicitly leverage cross-modal signals to shape computational strategies even for unimodal deployment.

2. VLMs Outperform LLMs on Text-Only Retrieval

We evaluate four publicly available model families—Qwen 2, 2.5, and 3 (Yang et al., 2024; 2025b;a), as well as InternLM 3 (Team, 2025)—comparing each text-only LLM with its VLM counterpart on two synthetic retrieval tasks. All models were evaluated as released, without any task-specific fine-tuning (see Appendix B for details).

Tasks. We employ two synthetic tasks: *Direct Retrieval*, where the model must look up a city associated with a named person in a long context (e.g., "James lives in Tokyo. ... Question: Where does John live? → Mumbai"), and *Indirect Retrieval*, which adds a compositional step—the query refers to a food rather than a person, requiring the model to first identify who likes that food and then retrieve their city (e.g., "Orange is liked by John. Mary lives in Paris. ... Question: Where does the person who likes Orange live? → Mumbai"). Both tasks are inspired by Feng & Steinhardt (2024) and require no prior world knowledge, since all information is contained in the context (see Appendix C for full task details).

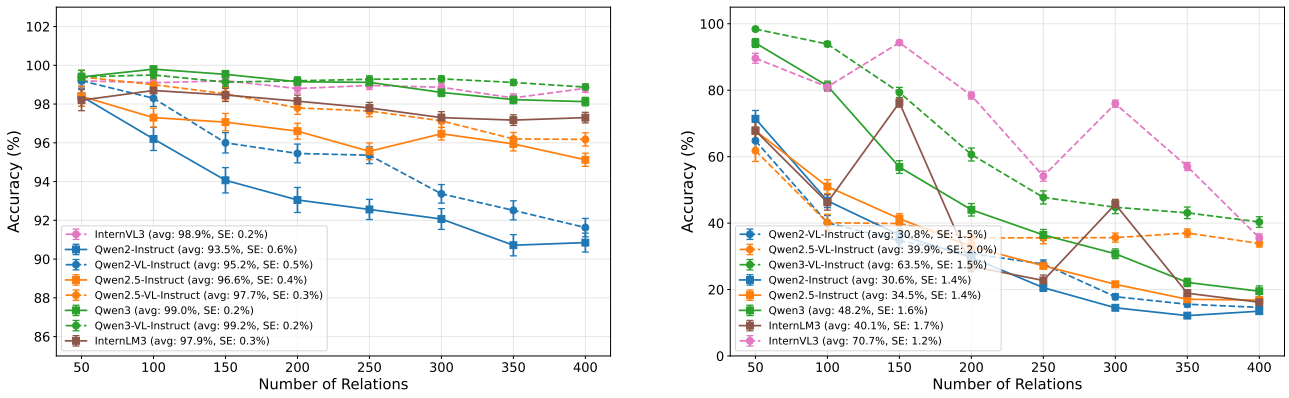
VLMs consistently outperform LLMs. Figure 2 presents the results across both tasks. For each context length l, we generated 10 independent contexts and evaluated every query position, yielding $10 \times l$ queries per length. In the Direct Retrieval task, VLMs consistently outperform their LLM counterparts across nearly all context lengths, with Qwen3 models achieving near-ceiling accuracies of approximately 98% even at $l = 400$. This gap becomes even more pronounced in the Indirect Retrieval task, where VLMs demonstrate a clear and persistent advantage. This is particularly surprising given that VLMs are trained primarily to enable LLMs to interpret visual information. Why do VLMs also exhibit superior performance on purely text-based retrieval tasks?

Note that all VLM variants have slightly larger parameter counts than their corresponding LLMs due to vision encoders and projectors; however, since text-only evaluation uses only the LLM backbone, these differences do not affect comparability. Individual model breakdowns are provided in Appendix A.

2.1. Controlled Task Formulation

To isolate the role of modality in this performance gap, we introduce a modified Indirect Retrieval task that enables controlled switching between modalities while keeping the underlying task unchanged. The task links colored shapes and unique target labels. The model must track objects defined by both their shape and color (e.g., a red triangle), then retrieve the correct item when queried using only the object's color (e.g., returning item a when asked about "red"), as illustrated in Figure 1. Using simple visual primitives—shapes and colors—allows the task to be presented identically across modalities, either as a textual description or as an image, without altering its structure.

Formally, let $\mathcal{A}$ be a set of attributes (colors), $\mathcal{E}$ a set of entities (shapes), and $\mathcal{I}$ a set of items. The goal is to map a query attribute $q \in \mathcal{A}$ to a target item $y \in \mathcal{I}$ through an intermediate entity $e \in \mathcal{E}$. This requires two reasoning steps:



(a) Direct Retrieval *(b)* Indirect Retrieval

Figure 2. Binding accuracy comparison between text-only LLMs and their VLM counterparts across four model families. VLMs consistently outperform LLMs as context length increases. The left panel shows Direct Retrieval; the right panel shows the harder Indirect Retrieval task. See Appendix A for individual model breakdowns.

identify the entity e associated with q using the context set $\mathcal{C} = \{(a_i, e_i)\}_i$, then retrieve the item y linked to e using the association set $\mathcal{B} = \{(e_j, y_j)\}_j$. Because the task uses controlled vocabulary and fixed logical structure, it provides a clean testbed for applying causal intervention and circuit analysis tools without confounds from natural language variability.

The unified prompt structure for both modalities is:

$$\mathbf{x} = [\mathbf{X}_{\text{context}}, \texttt{[CTX\_END]}, \mathbf{X}_{\text{associations}}, \texttt{[QUE]}, \mathbf{x}_{\text{query}}]$$

where $\mathbf{X}_{\text{associations}}$ defines Entity $\rightarrow$ Item assignments and `[CTX_END]` and `[QUE]` are special delimiters. In the **text modality**, the context is a sequence of attribute-entity pairs: $\mathbf{X}^{\text{text}}_{\text{context}} = [a_1, e_1, a_2, e_2, \ldots, a_N, e_N]$. In the **vision modality**, the context consists of a tokenized image rendered with each entity depicted in its associated color: $\mathbf{X}^{\text{image}}_{\text{context}} = [\texttt{<IMG>}_1, \texttt{<IMG>}_2, \ldots, \texttt{<IMG>}_N]$. Full task details, including vocabulary, prompt structure, and the auxiliary Feature Grounding task, are provided in Appendix D.

To investigate why VLMs outperform LLMs, we now replicate this phenomenon in a controlled setting that enables mechanistic analysis.

3. VLM Gains in a Controlled Setting

In this section, we replicate the performance gains of VLMs over LLMs reported in Section 2 by training a 12-layer decoder-only Transformer on the Indirect Retrieval task. We first train the model exclusively on the text modality, where the context $\mathbf{X}^{\text{text}}_{\text{context}}$ consists of sequential token pairs representing attribute–entity associations (e.g., *red triangle, blue circle*). Training follows a curriculum that gradually increases task complexity, from 2 object pairs with a restricted vocabulary to contexts containing up to 8 objects, yielding the baseline $\mathcal{M}_{\text{text-only}}$ (see Appendix E for full details).

Next, we continue training on the image modality. As described in Section 2.1, the task can be instantiated with rendered images depicting colored shapes. We replace the text context with token representations extracted from a frozen, pretrained image encoder, experimenting with three architectures: ResNet-152 (He et al., 2016), ViT-B/16 (Dosovitskiy et al., 2021), and DINOv3 (Siméoni et al., 2025). After image-only training, we transfer the model back to text on a mixed curriculum (20% image, 80% text), then expand the vocabulary for OOD evaluation, yielding $\mathcal{M}_{\text{image-text}}$. We train $\mathcal{M}_{\text{text-only}}$ with four random seeds and three encoders per seed, resulting in 11 valid runs for $\mathcal{M}_{\text{image-text}}$ (excluding one divergent run).

Figure 3 compares the models on the Indirect Retrieval task as the number of objects increases. The text-only model, $\mathcal{M}_{\text{text-only}}$ (blue curve), generalizes poorly to OOD sequence lengths: its accuracy drops sharply beyond the training maximum of eight items, yielding an average OOD accuracy of 37.2%. In contrast, $\mathcal{M}_{\text{image-text}}$ (green curve) shows substantially stronger generalization, achieving 69.5%. This replicates the VLM advantage observed at scale.

Ruling Out Positional Range. A natural hypothesis is that image encoders produce longer token sequences (e.g., 196 patch tokens per image), exposing the model to a broader range of positional indices than the text-only training. To test this, we fine-tune both models with unattendable noise tokens (Shen et al., 2023) inserted between context pairs, yielding $\mathcal{M}_{\text{noise-text}}$ and $\mathcal{M}_{\text{noise-image-text}}$. As shown in Figure 3, noise augmentation improves $\mathcal{M}_{\text{noise-text}}$ to 57.5% and $\mathcal{M}_{\text{noise-image-text}}$ to 83.6%. Critically, even the noise-augmented text model (57.5%) still substantially underperforms $\mathcal{M}_{\text{image-text}}$ without noise (69.5%), demonstrating that exposure to longer positional ranges is insufficient to ex-

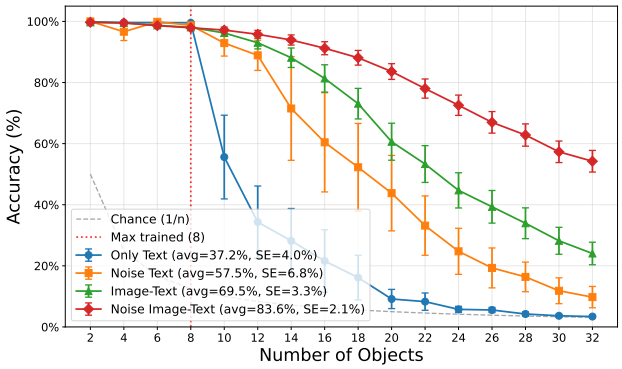


Figure 3. Combined generalization performance across all models. **Text-Only** (Blue) suffers from sharp degradation on OOD lengths (37.2% average). **Image-Text** (Green) significantly improves generalization (69.5% average). **Noise-Text** (Orange) benefits from positional range expansion but remains insufficient (57.5% average). **Noise-Image-Text** (Red) achieves the strongest robustness (83.6% average). See Appendix F for individual plots.

plain the full gains from visual training. The visual modality must provide a qualitatively different form of regularization beyond simple context-length expansion.

4. Explaining the Gains: Image Training Induces Binding Strategy Shifts

The previous section established that visual training substantially improves OOD generalization beyond what can be explained by increased exposure to longer positional ranges. In this section, we use mechanistic interpretability to show that this improvement corresponds to a shift in how the model binds and retrieves information. Specifically, the dominant binding strategy moves from a brittle, position-dependent mechanism to a more robust, symbolic one. We test three hypotheses:

H1: Visual training causes a shift from positional to symbolic binding. *Supported:* the shift is consistent across all 4 text-only seeds (positional) and all 11 valid image-text runs (symbolic; see Section 3 for details).

H2: Translation invariance in images is the primary driver of this shift. *Partially supported:* removing spatial variability at the task level yields mixed results (2/4 seeds retained positional binding), suggesting that encoder-level inductive biases beyond translation invariance also contribute.

H3: The shift is not explained by exposure to longer positional ranges. *Supported:* noise-augmented text models ($\mathcal{M}_{\text{noise-text}}$, 57.5% OOD) still substantially underperform image-trained models ($\mathcal{M}_{\text{image-text}}$, 69.5% OOD).

4.1. Background: Binding Mechanisms in Transformers

Recent work has identified multiple mechanisms that transformers use for variable binding (Gur-Arieh et al., 2025). In **positional binding**, the model retrieves an entity based on its ordinal position in the sequence, relying on positional encodings. In **symbolic binding**, retrieval is content-based: the query matches the associated entity in a position-agnostic way (termed "lexical" in Gur-Arieh et al. 2025; we use "symbolic" following Wu et al. 2025 to accommodate visual inputs). A third mechanism, **reflexive binding**, relies on pointer-like references; since our task structure makes symbolic binding viable, we focus on the competition between positional and symbolic strategies.

4.2. Methodology: Identifying Binding Mechanisms

To determine which binding mechanism our models rely on, we use the interchange intervention proposed by Gur-Arieh et al. (2025). This method constructs paired inputs—an original and a counterfactual—designed so that a positional strategy would retrieve based on position (which differs between pairs) while a symbolic strategy would retrieve based on semantic content (which may remain unchanged). By patching activations from the counterfactual into the original at different layers, we can measure which mechanism dominates. Full details are provided in Appendix G.

4.3. The Clean Case: A Complete Mechanism Shift

We first analyze binding mechanisms without noise augmentation. Figure 4 reports, for each layer, the proportion of times the model's mechanism is classified as positional, symbolic, reflexive, or mixed when solving text-modality inputs with up to eight objects. The dominant mechanism remains largely consistent across layers, except for the final layer. In this layer, $\mathcal{M}_{\text{text-only}}$ adopts an almost exclusively positional strategy, relying on token positions. In contrast, $\mathcal{M}_{\text{image-text}}$ primarily uses symbolic binding, matching semantic content directly. This pattern is consistent across all runs: text-only models converge to positional binding, while image-trained models converge to symbolic binding (see Appendix G for aggregate results).

A symbolic mechanism naturally generalizes to longer contexts because retrieval is based on semantic matching rather than position counting. Positional binding relies on locating "the i-th binding" within the sequence, which breaks when the model encounters positions beyond its training distribution. This distinction explains the OOD performance gap in Section 3.

4.4. Why Images Induce the Mechanism Shift

We hypothesize that **translation invariance**—intrinsic to visual inputs but absent from sequential text—plays a cen-

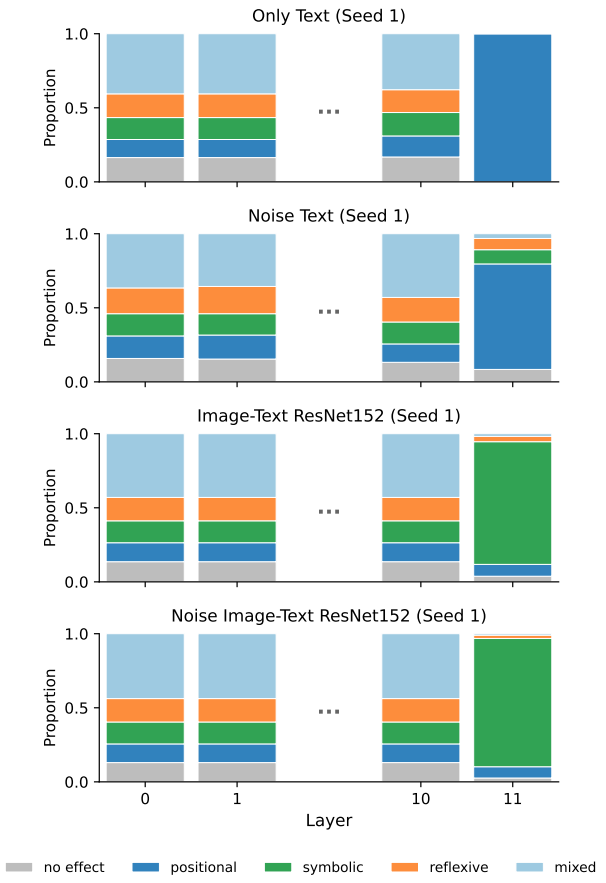


Figure 4. **Binding Mechanism Shift.** Interchange intervention results showing the dominant binding mechanism at each layer. Text-only models rely on the **positional** mechanism (blue). Noise augmentation introduces modest symbolic binding but remains predominantly positional. Image-trained models transition to a **symbolic** mechanism (orange).

tral role. In the text modality, object pairs appear at fixed positions that the model can exploit as shortcuts. In the image modality, spatial positions are arbitrary and carry no task-relevant information, rendering positional binding ineffective. This creates pressure for the model to discover symbolic binding based on semantic identity.

To test this, we trained models with fixed object positions across all images, eliminating spatial variability at the task level. Results were mixed: across four seeds, two models retained positional binding while two transitioned to symbolic binding. Because this experiment is limited to four seeds, this 2/2 split should be interpreted cautiously. Nonetheless, it suggests that pretrained image encoders retain multiple inductive biases—translation invariance among them—and that reducing spatial variability does not fully eliminate the pressure toward symbolic strategies. Translation invariance is an important contributing factor, but not the sole driver (see Appendix H for details).

MODEL	WINDOW-AVG. SYM./POS. RATIO	Δ
QWEN 2	1.481	—
QWEN 2-VL	1.609	+0.128
QWEN 2.5	1.589	—
QWEN 2.5-VL	1.846	+0.257
QWEN 3	1.112	—
QWEN 3-VL	1.218	+0.106
INTERNLM 3	1.676	—
INTERNVL 3	2.044	+0.368

Table 1. **Binding Mechanisms in Large-Scale Models.** VLM variants consistently exhibit higher symbolic-to-positional ratios than their text-only counterparts across all tested families.

4.5. The Noise Case: Mixed Mechanisms and Pre-trained LLMs

We now consider the case where noise tokens are added. $\mathcal{M}_{\text{noise-text}}$ exhibits a slightly different profile: its binding remains predominantly positional, but with a modest increase in symbolic behavior. Introducing unattendable noise tokens disrupts the clean positional regularities of the synthetic text data, making strict positional counting less reliable and encouraging partial reliance on semantic matching.

This mirrors observations in large pre-trained LLMs, which use a mixture of positional and symbolic mechanisms (Gur-Arieh et al., 2025). Natural language is inherently irregular, introducing variability that weakens positional shortcuts. However, this shift is incomplete: despite increased symbolic behavior, $\mathcal{M}_{\text{noise-text}}$ (57.5% OOD) still substantially underperforms $\mathcal{M}_{\text{image-text}}$ (69.5% OOD), indicating that noise alone does not induce fully robust binding. Visual training exerts a stronger and more direct pressure toward symbolic strategies, changing which mechanism is viable rather than merely degrading positional cues. Consistent with this, $\mathcal{M}_{\text{noise-image-text}}$ combines both effects: it exhibits symbolic binding similar to $\mathcal{M}_{\text{image-text}}$ yet achieves higher OOD performance (83.6%), suggesting complementary roles.

4.6. Validation on Large-Scale Models

To verify that our findings generalize beyond controlled experiments, we performed interchange interventions on the Qwen model family (Qwen 2, 2.5, and 3) and InternLM 3, employing the same methodology as Gur-Arieh et al. (2025). We use a **window-averaged binding score** that averages the symbolic-to-positional ratio across a designated binding window for each model (see Appendix G for details).

As shown in Table 1, VLM variants consistently exhibit a higher symbolic-to-positional ratio across both families. This aligns with the behavioral results in Section 2 and

indicates that visual training systematically shifts models toward symbolic binding.

5. Circuit-Level Implementation of Binding Mechanisms

Having established *that* visual training shifts models from positional to symbolic binding using interchange interventions, we now characterize *how* these mechanisms are implemented at the circuit level. We employ two complementary tools: **attention knockouts** (Geva et al., 2023), which mask attention between specific token pairs to identify causally necessary pathways, and **linear probes** (Alain & Bengio, 2017), which decode information from residual stream activations to trace when and where binding content becomes explicitly represented.

5.1. Three Circuit Architectures

Figure 5 illustrates the information flow in each circuit type. Early layers encode token-level information (color, shape, item identity), and colored arrows indicate attention-mediated information transfer. The key distinction is that the Positional Circuit transfers only position indices (implicit binding), while Symbolic Circuits transfer semantic content (explicit binding).

The Positional Circuit operates through two independent streams. In the query stream, the question token (the color in the query) attends to the matching color in the context and copies its *position index*; the `[ANSWER]` token then copies this position. In the association stream, the shape token finds its matching shape in the context and copies its position index; the item token then copies this position. The answer is retrieved by matching position indices between the two streams. Crucially, attribute information is never transferred between tokens—the "binding" is implicit, relying solely on shared position indices.

Symbolic Circuit A (Color-Key) uses color as the retrieval key. The context shape token first accumulates color identity from the adjacent context color, creating an explicit attribute-entity bundle. In the association stream, the association shape retrieves this stored color from the context. Both the association item and the answer token converge on this color information to produce the correct output.

Symbolic Circuit B (Shape-Key) is similar to Circuit A but uses shape rather than color as the retrieval key: the query retrieves a shape identity from the context, and the answer token uses this for final retrieval. Despite the different intermediate key, Circuit B shares the fundamental property of Circuit A: semantic content is explicitly transferred between tokens rather than position indices.

5.2. Causal Evidence: Attention Knockouts

To validate these circuit architectures, we use attention knockouts to identify critical token-to-token connections. For each source-target pair, we mask all attention from the source to the target and measure the resulting accuracy drop. High knockout effect indicates essential information flow.

The results, shown in Figure 6, confirm the predicted pathways. For the **Positional Circuit** (Figure 6a), the critical pathways are Question→Context Color (position lookup), Answer→Question (position copy), Association Shape→Context Shape (position lookup), and Association Item→Association Shape (position copy). Notably, there is no critical attention between color and shape tokens—the circuit never routes semantic content between positions.

For **Symbolic Circuit A**, the critical pathways are Context Shape→Context Color (binding step), Association Shape→Context Shape (color retrieval from context), Association Item→Association Shape (color propagation), and Answer→Query (color copy). These attribute-routing pathways are absent in the positional circuit.

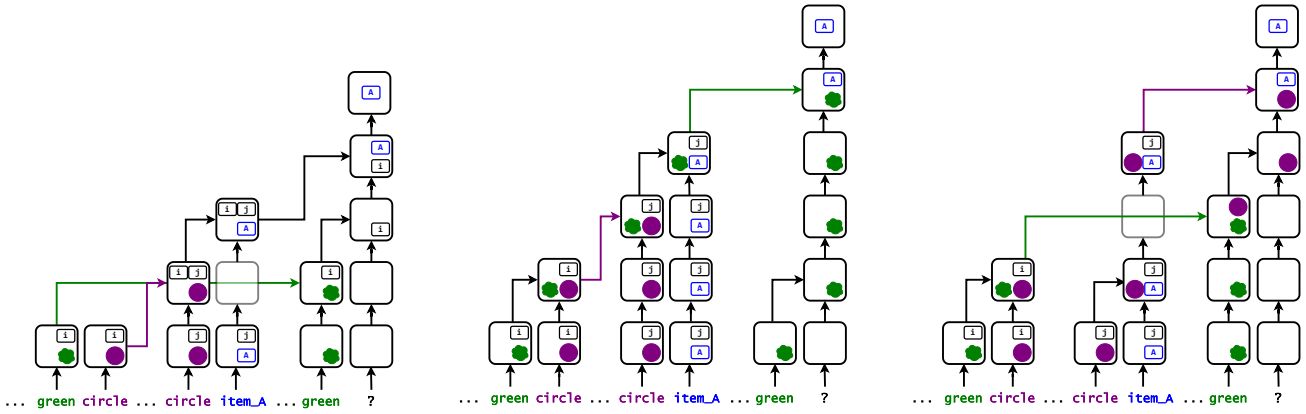
For **Symbolic Circuit B**, the critical pathways shift: Query→Context Shape becomes critical (the query retrieves shape identity from context), and the Association Item→Association Shape pathway propagates shape rather than color. Full per-seed knockout matrices and circuit-variant breakdowns are provided in Appendix I.

5.3. Representational Evidence: Linear Probes

We complement the knockout analysis with linear probes trained on residual stream activations to decode position indices, color identity, shape identity, and item identity at each layer and token position. By tracking when and where each type of information becomes decodable, we trace semantic content through the network.

We define the **binding signature** as a diagnostic pattern: a marked surge in attribute decodability at entity token positions. When a model explicitly binds attributes to entities—i.e., stores "red" as a property of "circle"—linear probes can decode the attribute identity from the entity token's residual stream above chance. Conversely, when binding is implicit (as in the positional mechanism), entity tokens carry no attribute information. The binding signature thus provides a detectable marker of explicit binding that can be applied even to models where full circuit analysis is infeasible.

Figure 7 presents the probe results. In the **Positional Circuit** (Figure 7a), position information accumulates at query and association positions, but attribute identity (color, shape) remains *undecodable* at entity positions throughout the network. The model never represents bound units like "red circle"—only position indices are transferred.



(a) Positional Circuit *(b)* Symbolic A (Color-Key) *(c)* Symbolic B (Shape-Key)

Figure 5. **Circuits corresponding to the binding mechanisms.** Early layers encode token-level information (color, shape, item identity). Colored arrows indicate attention-mediated information transfer between tokens. The key distinction: the Positional Circuit transfers only position indices (implicit binding), while Symbolic Circuits transfer semantic content (explicit binding).

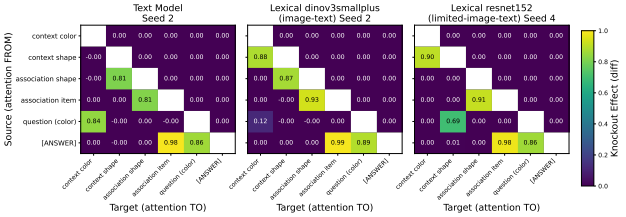


Figure 6. **Attention knockout analysis.** Heatmaps showing the drop in performance when attention between specific token pairs is masked. (a) The Positional mechanism relies on position-matching pathways (Question→Context Color, Association Shape→Context Shape). (b, c) Symbolic mechanisms show attribute-routing pathways. Aggregate results across all seeds are in Appendix J.

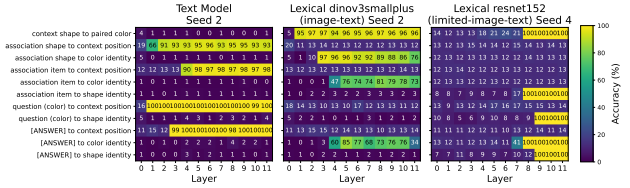


Figure 7. **Linear probe analysis.** Probing accuracy for task-relevant features across representative runs. (a) The Positional model accumulates position but not attribute information at entity positions. (b, c) Symbolic models exhibit the binding signature: attribute decodability surges at entity positions. Aggregate results across all seeds are in Appendix K.

In the **Symbolic Circuits** (Figure 7b,c), we observe the binding signature: attribute decodability surges at entity positions. For Circuit A, color identity first becomes decodable at the context shape token (the binding step) and then at the association shape and item tokens (retrieval and propagation). For Circuit B, shape identity becomes decodable at the query token after it attends to the context shape. Despite the different intermediate key, both variants exhibit the fundamental property of explicit attribute transfer.

The knockout and probe results triangulate: for Symbolic Circuit A, knockouts show that attention from Context Shape→Context Color is causally necessary, and probes confirm that color information becomes decodable at the context shape position immediately after this attention step. The knockout identifies the causal mechanism; the probe confirms its representational consequence. This convergence—causal necessity plus decodable content—provides stronger evidence than either signal alone.

5.4. Robustness Across Seeds and Encoders

The circuit patterns are highly consistent across experimental runs. All four text-only seeds exhibit the positional pattern: high position decodability, low attribute decodability, and sparse position-matching knockout pathways. All eleven valid image-text runs (see Section 3 for details) exhibit the symbolic pattern with the binding signature and attribute-routing knockout pathways. The specific circuit variant (A vs. B) varies by seed and encoder, but explicit attribute transfer is universal across image-trained runs (see Appendix I for per-run breakdowns).

6. Related Work

Multi-modal training effects on language models. Recent work has systematically compared multi-modal training regimes against their language-only backbones to understand how visual grounding reshapes linguistic capabilities. From a theoretical standpoint, multi-modal training has been argued to promote robustness and generalization by discouraging reliance on modality-specific correlations (Xue et al.,

2024) and by inducing a more faithful and structured latent semantic space (Huang et al., 2021). Empirically, several studies show that VLMs can retain or even improve text-only performance, with Dai et al. (2024b) reporting gains in language understanding, mathematics, coding, and reasoning, and Ratzlaff et al. (2025) observing improvements in commonsense reasoning. However, these benefits are not uniform: multi-modal training can also introduce regressions, such as degraded mathematical reasoning in some LLaVA-style models (Ratzlaff et al., 2025).

Mechanistic Interpretability. In this work, we adopt a mechanistic interpretability perspective to analyze the internal computations of Transformers. We combine interchange interventions (Meng et al., 2022; Geiger et al., 2021; Finlayson et al., 2021; Vig et al., 2020; Geiger et al., 2020) to causally identify which hidden activations drive specific outputs and how binding and compositional information is propagated (Feng & Steinhardt, 2024; Saravanan et al., 2025; Gur-Arieh et al., 2025; Wu et al., 2025), attention knockout techniques (Geva et al., 2023; Gur-Arieh et al., 2025) to assess the functional role of specific attention pathways, and linear probing methods (Alain & Bengio, 2017; Ravichander et al., 2021; Belinkov, 2022) to characterize which variables are explicitly encoded across layers.

Binding. Binding—the ability to correctly associate entities with their attributes (Treisman, 1996; Feng & Steinhardt, 2024)—has recently been examined through mechanistic analyses that probe how Transformers implement and represent such associations. Prior work has leveraged interchange interventions (Feng & Steinhardt, 2024; Saravanan et al., 2025; Gur-Arieh et al., 2025; Prakash et al., 2025; Wu et al., 2025) and analyses of attention head behavior (Wu et al., 2025; Urrutia et al., 2025) to characterize how binding information is encoded, propagated, and learned within the network (Wu et al., 2025). We build directly on this line of research, grounding our analysis in the taxonomy of binding mechanisms introduced by Gur-Arieh et al. (2025), which distinguishes between positional (Dai et al., 2024a; Prakash et al., 2024; 2025), symbolic, and reflexive binding, and use it as a unifying framework to interpret the binding behaviors observed in our models. A more detailed discussion of related work, including extended coverage of length generalization benchmarks, is provided in Appendix M.

7. Conclusion

We showed that training on visual data can improve performance on purely text-based retrieval tasks by reshaping how models implement binding. Through controlled experiments and mechanistic interpretability analyses, we found that while text-only training encourages positional binding behavior, visual training induces a shift toward symbolic binding. This shift accounts for the improvements in OOD generalization and long-context reasoning observed both in our synthetic setting and in large pretrained VLMs.

From a methodological standpoint, our work demonstrates how combining causal intervention, circuit analysis, and representational probing can provide a complete mechanistic picture: interchange interventions identify which strategy a model uses, attention knockouts reveal how it is structurally implemented, and linear probes characterize where binding information is represented. The *binding signature*—a surge in attribute decodability at entity positions—provides a lightweight diagnostic that can be applied to large models where full circuit analysis is infeasible. This triad of tools could similarly evaluate other training interventions, from RLHF to data augmentation, by tracking whether they induce shifts toward more robust internal computation.

More broadly, our findings suggest that multimodal objectives may offer a principled way to mitigate shortcut learning and strengthen generalization, and that training recipes could explicitly leverage cross-modal signals to shape computational strategies even for unimodal deployment. An important direction for future work is to translate these insights into concrete architectural or training design choices, and to identify which modality properties—such as translation invariance in visual models—most effectively drive beneficial representational shifts.

Limitations. Our study focuses on information-retrieval-style tasks with a minimal vocabulary because they provide a clean testbed for isolating binding mechanisms using established mechanistic interpretability tools (Gur-Arieh et al., 2025). Introducing full natural language would confound this analysis with additional variability. While binding is a necessary—but not sufficient—component of reasoning, and models with stronger retrieval abilities are likely to enjoy advantages in downstream tasks (Feng & Steinhardt, 2024), our evaluation does not establish that the observed symbolic mechanisms directly enable broader abstract reasoning in naturalistic settings. Additionally, our large-model comparisons are limited to publicly available model families (Qwen and InternLM), and differences in instruction-tuning data or RLHF procedures between LLM and VLM variants are not publicly documented and could contribute to observed effects. Finally, while we identify translation invariance as a key driver of the binding shift, other encoder-level inductive biases likely also play a role. Evaluating transfer to natural long-context benchmarks while preserving mechanistic clarity is a valuable but non-trivial direction for future work.

References

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes, 2017. URL https://openreview.net/forum?id=ryF7rTqgl.

Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., and Zhou, J. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023. URL https://arxiv.org/abs/2308.12966.

Bastings, J., Baroni, M., Weston, J., Cho, K., and Kiela, D. Jump to better conclusions: Scan both left and right. *arXiv preprint arXiv:1809.04640*, 2018.

Belinkov, Y. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, March 2022. doi: 10.1162/coli_a_00422. URL https://aclanthology.org/2022.cl-1.7/.

Cooper, A., Kato, K., Shih, C.-H., Yamane, H., Vinken, K., Takemoto, K., Sunagawa, T., Yeh, H.-W., Yamanaka, J., Mason, I., et al. Rethinking vlms and llms for image classification. *Scientific Reports*, 15(1):19692, 2025.

Csordás, R., Irie, K., and Schmidhuber, J. The devil is in the detail: Simple tricks improve systematic generalization of transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 619–634, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.49. URL https://aclanthology.org/2021.emnlp-main.49.

Dai, Q., Heinzerling, B., and Inui, K. Representational analysis of binding in language models. In *2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024*, pp. 17468–17493. Association for Computational Linguistics (ACL), 2024a.

Dai, W., Lee, N., Wang, B., Yang, Z., Liu, Z., Barker, J., Rintamaki, T., Shoeybi, M., Catanzaro, B., and Ping, W. Nvlm: Open frontier-class multimodal llms. *CoRR*, 2024b.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Feng, J. and Steinhardt, J. How do language models bind entities in context? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=zb3b6oKO77.

Finlayson, M., Mueller, A., Gehrmann, S., Shieber, S. M., Linzen, T., and Belinkov, Y. Causal analysis of syntactic agreement mechanisms in neural language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1828–1843, 2021.

Geiger, A., Richardson, K., and Potts, C. Neural natural language inference models partially embed theories of lexical entailment and negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pp. 163–173, 2020.

Geiger, A., Lu, H., Icard, T., and Potts, C. Causal abstractions of neural networks. *Advances in Neural Information Processing Systems*, 34:9574–9586, 2021.

Geva, M., Bastings, J., Filippova, K., and Globerson, A. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12216–12235, 2023.

Gur-Arieh, Y., Geva, M., and Geiger, A. Mixing mechanisms: How language models retrieve bound entities in-context. *arXiv preprint arXiv:2510.06182*, 2025.

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

Huang, Y., Du, C., Xue, Z., Chen, X., Zhao, H., and Huang, L. What makes multi-modal learning better than single (provably). *Advances in Neural Information Processing Systems*, 34:10944–10956, 2021.

Jelassi, S., d'Ascoli, S., Domingo-Enrich, C., Wu, Y., Li, Y., and Charton, F. Length generalization in arithmetic transformers. *arXiv preprint arXiv:2306.15400*, 2023.

Lake, B. and Baroni, M. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International conference on machine learning*, pp. 2873–2882. PMLR, 2018.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in gpt. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372. Curran Associates, Inc., 2022. URL https://proceedings.neurips.

cc/paper_files/paper/2022/file/6f1d43d5a82a37e89b0665b33bf3a182-Paper-Conference.pdf.

Prakash, N., Shaham, T. R., Haklay, T., Belinkov, Y., and Bau, D. Fine-tuning enhances existing mechanisms: A case study on entity tracking. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=8sKcAWOf2D.

Prakash, N., Shapira, N., Sharma, A. S., Riedl, C., Belinkov, Y., Shaham, T. R., Bau, D., and Geiger, A. Language models use lookbacks to track beliefs, 2025. URL https://arxiv.org/abs/2505.14685.

Ratzlaff, N., Luo, M., Su, X., Lal, V., and Howard, P. Training-free mitigation of language reasoning degradation after multimodal instruction tuning. In *Proceedings of the AAAI Symposium Series*, volume 5, pp. 384–388, 2025.

Ravichander, A., Belinkov, Y., and Hovy, E. Probing the probing paradigm: Does probing accuracy entail task relevance? In Merlo, P., Tiedemann, J., and Tsarfaty, R. (eds.), *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 3363–3377, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.295. URL https://aclanthology.org/2021.eacl-main.295/.

Ruis, L., Andreas, J., Baroni, M., Bouchacourt, D., and Lake, B. M. A benchmark for systematic generalization in grounded language understanding. *Advances in Neural Information Processing Systems*, 33, 2020.

Saparov, A., Pang, R. Y., Padmakumar, V., Joshi, N., Kazemi, M., Kim, N., and He, H. Testing the general deductive reasoning capacity of large language models using ood examples. *Advances in Neural Information Processing Systems*, 36:3083–3105, 2023.

Saravanan, D., Tapaswi, M., and Gandhi, V. Investigating mechanisms for in-context vision language binding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4891–4895, June 2025.

Shen, R., Bubeck, S., Eldan, R., Lee, Y. T., Li, Y., and Zhang, Y. Positional description matters for transformers arithmetic. *arXiv preprint arXiv:2311.14737*, 2023.

Siméoni, O., Vo, H. V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., Massa, F., Haziza, D., Wehrstedt, L., Wang, J., Darcet, T., Moutakanni, T., Sentana, L., Roberts, C., Vedaldi, A., Tolan, J., Brandt, J., Couprie, C., Mairal, J., Jégou, H., Labatut, P., and Bojanowski, P. Dinov3, 2025. URL https://arxiv.org/abs/2508.10104.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Team, I. Internlm3 technical report. https://github.com/InternLM/InternLM, 2025.

Treisman, A. The binding problem. 6(2):171–178, 1996. ISSN 0959-4388. doi: 10.1016/s0959-4388(96)80070-5.

Urrutia, F., Salas, J., Kozachinskiy, A., Calderon, C. B., Pasten, H., and Rojas, C. Decoupling positional and symbolic attention behavior in transformers. *arXiv preprint arXiv:2511.11579*, 2025.

Vig, J., Gehrmann, S., Belinkov, Y., Qian, S., Nevo, D., Singer, Y., and Shieber, S. Investigating gender bias in language models using causal mediation analysis. *Advances in neural information processing systems*, 33: 12388–12401, 2020.

Wu, B., Xu, C., Dai, X., Wan, A., Zhang, P., Yan, Z., Tomizuka, M., Gonzalez, J., Keutzer, K., and Vajda, P. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*, 2020.

Wu, Y., Geiger, A., and Millière, R. How do transformers learn variable binding in symbolic programs? In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=kVtyv7bpnw.

Xue, Y., Joshi, S., Nguyen, D., and Mirzasoleiman, B. Understanding the robustness of multi-modal contrastive learning to distribution shift. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=rtl4XnJYBh.

Yang, A., Yang, B., Hui, B., Zheng, B., Yu, B., Zhou, C., Li, C., Li, C., Liu, D., Huang, F., Dong, G., Wei, H., Lin, H., Tang, J., Wang, J., Yang, J., Tu, J., Zhang, J., Ma, J., Xu, J., Zhou, J., Bai, J., He, J., Lin, J., Dang, K., Lu, K., Chen, K., Yang, K., Li, M., Xue, M., Ni, N., Zhang, P., Wang, P., Peng, R., Men, R., Gao, R., Lin, R., Wang, S., Bai, S., Tan, S., Zhu, T., Li, T., Liu, T., Ge, W., Deng, X., Zhou, X., Ren, X., Zhang, X., Wei, X., Ren, X., Fan, Y., Yao, Y., Zhang, Y., Wan, Y., Chu, Y., Liu, Y., Cui, Z., Zhang, Z., and Fan, Z. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., Zheng, C., Liu, D., Zhou, F., Huang, F., Hu, F., Ge, H., Wei, H., Lin, H., Tang, J., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Zhou, J., Lin, J., Dang, K., Bao, K., Yang, K., Yu, L., Deng, L., Li, M., Xue, M., Li, M., Zhang, P., Wang, P., Zhu, Q., Men, R., Gao, R., Liu, S., Luo, S., Li, T., Tang, T., Yin, W., Ren, X., Wang, X., Zhang, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Zhang, Y., Wan, Y., Liu, Y., Wang, Z., Cui, Z., Zhang, Z., Zhou, Z., and Qiu, Z. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., and Qiu, Z. Qwen2.5 technical report, 2025b. URL https://arxiv.org/abs/2412.15115.

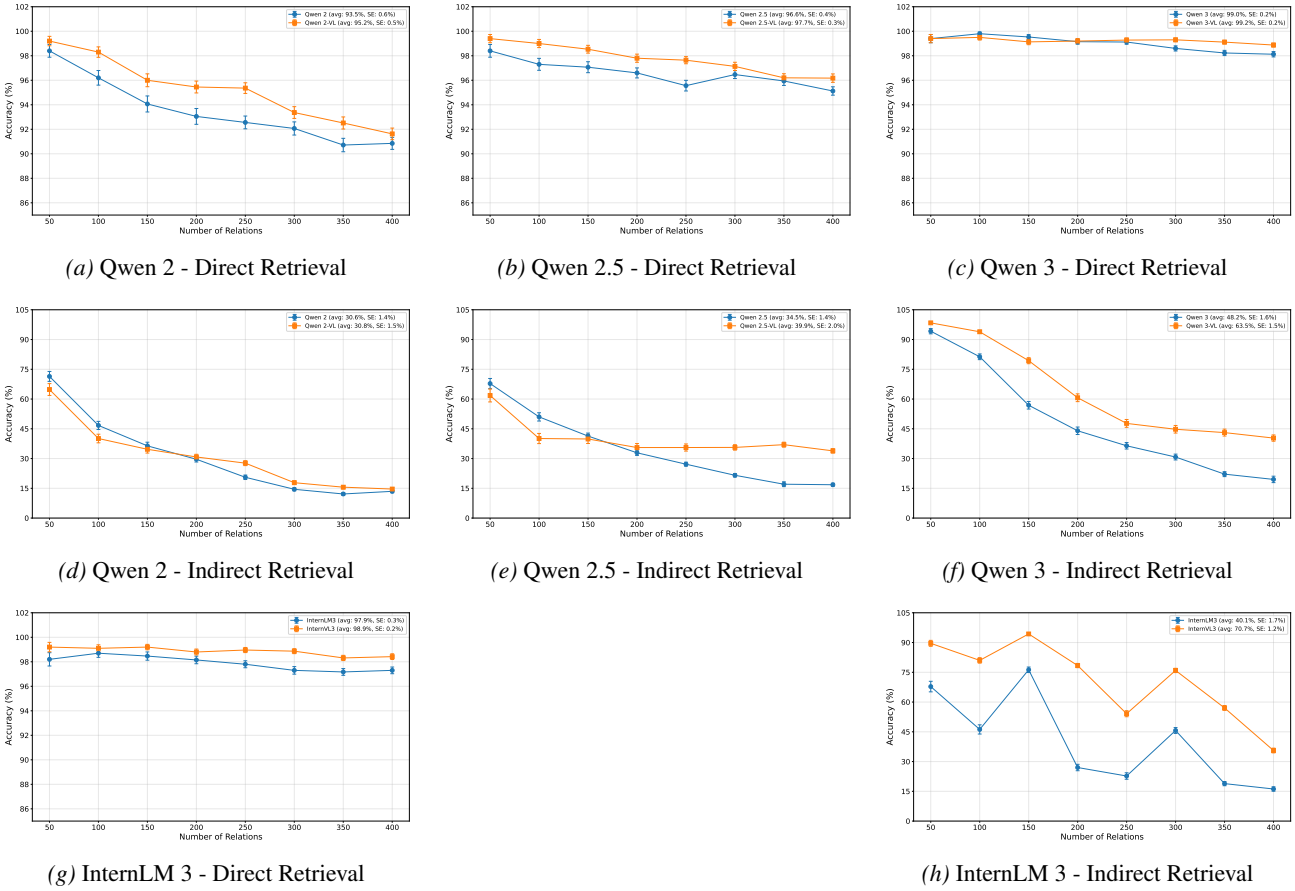
Zhang, Y., Backurs, A., Bubeck, S., Eldan, R., Gunasekar, S., and Wagner, T. Unveiling transformers with lego: a synthetic reasoning task. *arXiv preprint arXiv:2206.04301*, 2022.

Zhou, H., Bradley, A., Littwin, E., Razin, N., Saremi, O., Susskind, J. M., Bengio, S., and Nakkiran, P. What algorithms can transformers learn? a study in length generalization. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=AssIuHnmHX.

Zhou, Y., Alon, U., Chen, X., Wang, X., Agarwal, R., and Zhou, D. Transformers can achieve length generalization but not robustly. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2024b. URL https://openreview.net/forum?id=DWkWIh3vFJ.

A. Detailed Performance Breakdown

Figure 8 presents the full performance breakdown for all evaluated model families. The first two rows show Qwen model generations (Direct and Indirect Retrieval), and the third row shows the InternLM 3 family. Across all families, the VLM variants maintain higher accuracy at longer context lengths compared to their text-only counterparts.



(a) Qwen 2 - Direct Retrieval *(b)* Qwen 2.5 - Direct Retrieval *(c)* Qwen 3 - Direct Retrieval

(d) Qwen 2 - Indirect Retrieval *(e)* Qwen 2.5 - Indirect Retrieval *(f)* Qwen 3 - Indirect Retrieval

(g) InternLM 3 - Direct Retrieval *(h)* InternLM 3 - Indirect Retrieval

Figure 8. Detailed performance breakdown for all Qwen and InternLM model families on the Direct Retrieval and Indirect Retrieval tasks.

B. Detailed Pre-trained Models Results

This appendix provides details on the evaluation setup for the pre-trained model experiments presented in Section 2.

Models. We evaluated four model families, comparing their text-only (Instruct) and vision-language (VL-Instruct) variants:

- **Qwen 2**: `Qwen2-7B-Instruct` and `Qwen2-VL-7B-Instruct`
- **Qwen 2.5**: `Qwen2.5-7B-Instruct` and `Qwen2.5-VL-7B-Instruct`
- **Qwen 3**: `Qwen3-8B` and `Qwen3-VL-8B-Instruct`
- **InternLM 3**: `InternLM3-8B-Instruct` and `InternVL3-9B`

Inference Setup. All models were loaded using the HuggingFace Transformers library at full precision on a single NVIDIA RTX 3090 GPU. We used each model's default inference settings without modifying temperature or sampling parameters. Prompts were formatted using the model's chat template with the default system prompt ("You are a helpful assistant.").

Evaluation Protocol. For each query, we checked whether the model's generated output contained the correct city name. We report mean accuracy and standard error across all query positions for each context length.

C. Pretrained Model Retrieval Tasks

This appendix provides full details on the two synthetic retrieval tasks used to evaluate pretrained LLMs and VLMs in Section 2.

Direct Retrieval. The objective is to determine where a person lives given a long textual context. Formally, let $\mathcal{N}$ denote a set of names and C a set of cities. Each input consists of a context containing l statements of the form "n_i lives in c_i," where each $n_i \in \mathcal{N}$ and each $c_i \in C$ appears exactly once within the context. The query then asks for the city associated with one of the names mentioned. For example (with the model's output highlighted in blue):

> **Context:** James lives in Tokyo. Mary lives in Paris. John lives in Mumbai. **Question:** In what city does John live? John lives in Mumbai

The model is expected to generate the correct city c_j given the query name n_j.

Indirect Retrieval. This task introduces an additional reasoning step. In addition to $\mathcal{N}$ and C, we define a set of foods $\mathcal{F}$. The context is divided into two parts. The first contains l statements of the form "f_i is liked by n_i." The second contains statements of the same form as in the Direct Retrieval task, "n_i lives in c_i." Each name n_i appears exactly once in each part, so the index i links a food f_i to a city c_i through the same person n_i. The query refers to a food rather than a name, requiring the model to first identify the corresponding person and then retrieve their city. For instance:

> **Context:** Banana is liked by Mary. Orange is liked by John. Mary lives in Paris. John lives in Mumbai. **Question:** Where does the person that likes Orange live? Answer with just the city name. **City:** Mumbai

The model must first identify that Orange is liked by John, then retrieve that John lives in Mumbai. Importantly, neither task requires prior world knowledge (e.g., real-world capitals), since all necessary information is contained within the provided context.

Both tasks are inspired by the *Capitals* task of Feng & Steinhardt (2024). Direct Retrieval tests basic context lookup, while Indirect Retrieval adds a compositional retrieval step.

D. Task Details

This appendix provides implementation details for the Indirect Retrieval task introduced in Section 2.1.

Vocabulary. We define three disjoint vocabularies for the task:

- **Colors:** The full vocabulary consists of 216 colors from the web-safe color palette, formed by a $6 \times 6 \times 6$ RGB grid using hexadecimal values `00`, `33`, `66`, `99`, `cc`, and `ff` for each channel. During image-modality training, we restrict to the first 8 colors. In the text modality, colors are represented as tokens `color0001` through `color0216`.
- **Shapes:** We use 13 distinct shapes: rectangle, circle, hexagon, triangle, pentagon, rhombus, octagon, star, heart, semicircle, cross, arrow, and annulus. In the text modality, shapes are represented as tokens `shape0001` through `shape0013`.
- **Items:** Target labels are represented as tokens `item0001`, `item0002`, etc. Training uses up to 32 distinct items.

Prompt Structure. Both modalities share a common prompt structure with the following special tokens:

- `[CONTEXTEND]` separates the context (attribute-entity pairs) from the association list.
- `[SEP]` separates individual entity-item associations.
- `[QUESTION]` marks the beginning of the query.
- `[ANSWER]` precedes the target output position.

An example text-modality prompt with 3 objects is:

```
color0113 shape0055 color0119 shape0041 color0153 shape0091
[CONTEXTEND] shape0041 item0029 [SEP] shape0055 item0025
[SEP] shape0091 item0020 [QUESTION] color0119 [ANSWER]
```

Here, the query asks for the item associated with `color0119`. Since `color0119` is paired with `shape0041` in the context, and `shape0041` maps to `item0029`, the correct answer is `item0029`.

In the image modality, the context is replaced by a sequence of image tokens extracted from a frozen encoder (e.g., 196 tokens per image for ViT-based encoders), while the associations and query remain in text.

Image Rendering. For the image modality, each colored shape is rendered as a 224×224 pixel image. Shapes are drawn as filled SVG graphics on a white background. Figure 9 shows example renderings of colored shapes used as visual context.

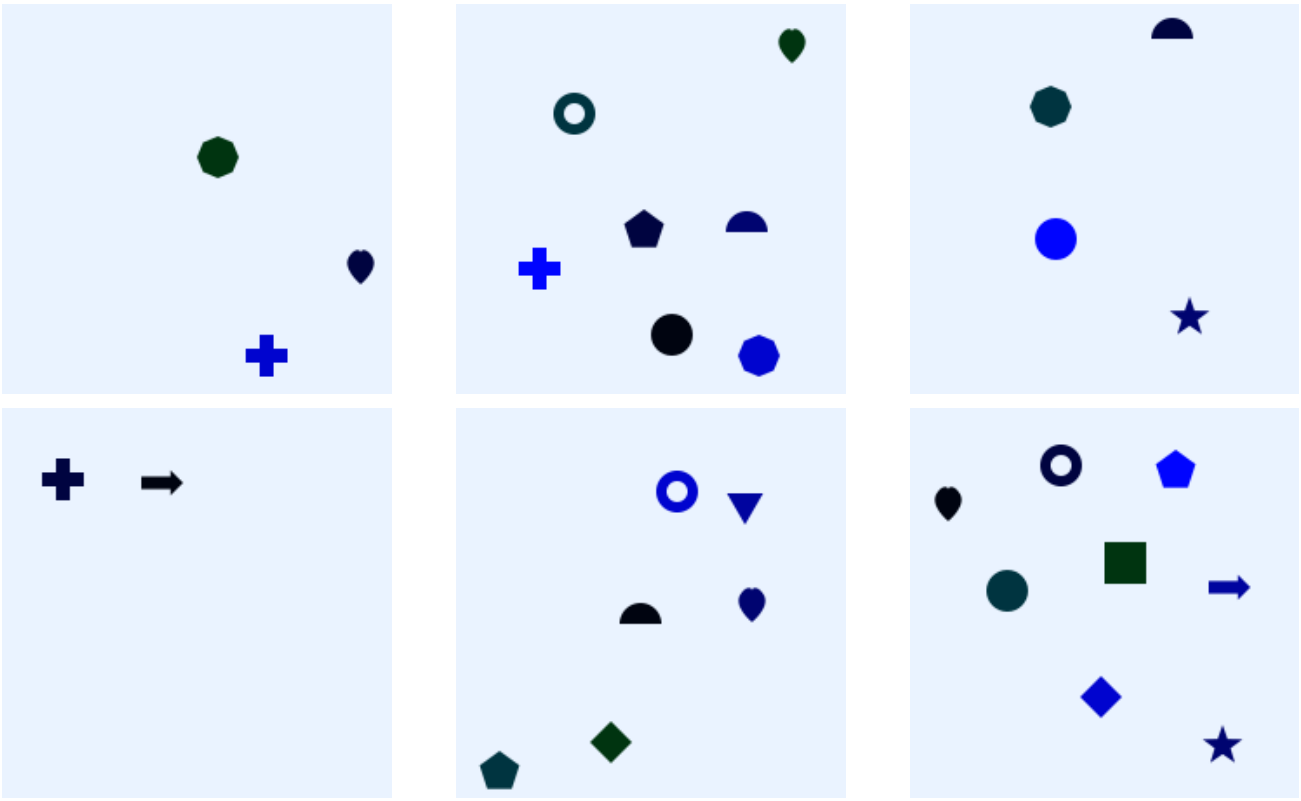


Figure 9. Example rendered contexts for the image modality, showing colored shapes drawn as filled SVG graphics on a white background.

Feature Grounding Task. In addition to the main Indirect Retrieval task, we use an auxiliary *Feature Grounding* task to stabilize training and disentangle reasoning failures from perceptual failures. This simpler task requires the model to directly map an attribute to its corresponding entity without the intermediate item-retrieval step. Given a context of attribute-entity pairs and a query attribute, the model must output the associated entity.

An example text-modality prompt is:

```
color0022 shape0192 color0054 shape0209 [CONTEXTEND]
[QUESTION] color0022 [ANSWER]
```

The correct answer is `shape0192`. In the image modality, the context is similarly replaced by image tokens while the query remains in text.

Symmetric Variations. We consider symmetric variations of both tasks where the roles of entity and attribute are interchangeable. Specifically, we include both configurations: querying by color to retrieve a shape, and querying by shape to retrieve a color. For the Indirect Retrieval task, this extends to assigning items to shapes (retrieved via color) versus assigning items to colors (retrieved via shape). This symmetric formulation is used during training to increase data diversity. All evaluation results reported in the main text use the color-to-shape query direction.

E. Detailed Train Procedure

E.1. Model Architecture

We use a 12-layer decoder-only Transformer with the architecture detailed in Table 2. The model employs Rotary Positional Embeddings (RoPE) (Su et al., 2024) applied to the query and key projections, using a fixed base frequency of $\theta = 10000$ with interleaved rotation on even and odd dimension pairs. The vocabulary consists of approximately 1000 tokens, covering all color and shape combinations used in our experiments.

Table 2. Model architecture hyperparameters.

Hyperparameter	**Value**
Number of layers	12
Hidden dimension (d_{model})	128
FFN dimension (d_{ff})	512
Attention heads	4
Positional encoding	RoPE ($\theta = 10000$)
Dropout	0
Vocabulary size	$\sim$1000

E.2. Training Configuration

All models are trained using the AdamW optimizer with a learning rate of 1×10^{-4}, no weight decay, and gradient clipping with a maximum norm of 1.0. We use a batch size of 32 and train with bf16 mixed precision. The learning rate schedule consists of a 2000-step linear warmup followed by a constant learning rate. Training continues until validation performance plateaus, using early stopping. Experiments were conducted on NVIDIA A6000 and A40 GPUs.

E.3. Text-Only Baseline

We begin by establishing a baseline model trained on the 1-hop indirect retrieval task defined in Section 2.1. We train exclusively on the text modality, where the context $\mathbf{X}_{\text{context}}^{\text{text}}$ consists of sequential token pairs representing attribute-entity associations (e.g., "red triangle", "blue circle").

Training a Transformer to solve this task from scratch is highly unstable due to the sparsity of the supervision signal. Since the intermediate tokens in the prompt are randomized or independent of the context, the model can only be trained to predict the final answer token. This means there is no direct reward for uncovering the intermediate logical steps (grounding and retrieval), requiring the model to discover the entire reasoning chain at once without partial credit for partial logic. To address this, we employ a five-stage curriculum to incrementally increase task complexity:

1. **Fixed Layout:** Two object pairs, restricted vocabulary (8 colors, 8 shapes), and fixed prompt order (Context $\rightarrow$ Bindings).
2. **Vocabulary Expansion:** Vocabulary increases to 216 colors and 216 shapes, while structure remains fixed.
3. **Variable Length:** Random sampling of 2 to 8 object pairs per prompt.
4. **Randomized Order:** Shuffling of pairs within the context and binding sections to prevent positional memorization.
5. **Scale:** Context capacity increased to support up to 32 items.

We denote the resulting model as $\mathcal{M}_{\text{text-only}}$. To ensure reliability, we repeated this full training curriculum using 4 distinct random seeds. For the experiments, we report the average performance across 4 seeds and the standard error (SE).

As shown in Figure 3, $\mathcal{M}_{\text{text-only}}$ (blue curve) generalizes poorly to OOD sequence lengths. Accuracy drops sharply beyond the training maximum of 8 items, resulting in an average OOD accuracy of 37.2% (SE: 4.0%).

E.4. Noise Training

After completing the text curriculum, we continue training $\mathcal{M}_{\text{text-only}}$ with noise augmentation to extend its positional range. We insert 100 unattendable noise tokens between each context pair, exposing the model to longer positional indices without adding semantic information. The augmented context is formulated as:

$$\mathbf{X}_{\text{context}}^{\text{noise}} = [a_1, e_1, \underbrace{[\text{noise}], \ldots, [\text{noise}]}_{100}, a_2, e_2, \ldots, a_N, e_N]$$

This produces $\mathcal{M}_{\text{noise-text}}$. As shown in Figure 3 (orange curve), noise augmentation substantially improves OOD generalization, increasing average accuracy from 37.2% to 57.5% (SE: 6.8%).

Critically, noise training serves as a prerequisite for the image training pipeline described below. By extending the model's positional range, noise training enables the model to skip the curriculum stages when training on images, where patch tokens naturally produce longer sequences.

E.5. Feature Grounding Pre-training

Before proceeding to image training, we train $\mathcal{M}_{\text{noise-text}}$ on the Feature Grounding auxiliary task in text mode. This task trains the model to predict associations in both directions: given an attribute, predict its associated entity (e.g., given "red", predict "triangle"), and vice versa. This establishes the structural requirements needed for the indirect retrieval task before introducing visual inputs.

E.6. Image Training

Following the Feature Grounding pre-training, we train on the indirect retrieval task in the vision modality. Images are rendered at 224×224 resolution using SVG-based rendering, where each entity is depicted with its corresponding attribute (e.g., colored shapes on a canvas).

Image Encoders. We experiment with three frozen pre-trained image encoders:

- **ResNet-152** (He et al., 2016): A supervised CNN producing 2048-dimensional spatial feature maps from the final convolutional stage. We discard global average pooling and fully connected layers, flattening the feature maps to produce pseudo-patch embeddings.
- **ViT-B/16** (Wu et al., 2020): A supervised Transformer producing 768-dimensional patch embeddings. We use features from the final layer, discarding the [CLS] token.
- **DINOv3** (Siméoni et al., 2025): A self-supervised Transformer producing 768-dimensional patch embeddings. We discard both [CLS] and register tokens to preserve spatial correspondence.

Visual Projector. To map visual representations into the language model's embedding space, we employ a 3-layer MLP projector with a hidden dimension multiplier of $4\times$ (i.e., if the encoder outputs d_{enc}-dimensional features, the hidden layer has dimension $4 \cdot d_{\text{enc}}$). We maintain a strict one-to-one mapping where each visual patch is projected to a single token, without pooling. The resulting features are normalized via LayerNorm both before and after projection, and scaled by $1/\sqrt{d_{\text{model}}}$ before concatenation into the input sequence.

Training. We train on image prompts, jointly optimizing for both 1-hop retrieval and Feature Grounding. For training stability, this stage uses a restricted distribution (8 colors, 13 shapes, up to 8 items), yielding $\mathcal{M}_{\text{image}}$.

E.7. Text Transfer and Vocabulary Expansion

After image training, we transfer the model back to the text domain in two stages:

Mixed Modality Training. We train on a 20/80 split of image and text tasks, maintaining the restricted vocabulary (8 colors, 13 shapes). This produces $\mathcal{M}_{\text{image-text-limited}}$.

Vocabulary Expansion. Since the image training uses only 13 shapes (limited by what can be rendered distinguishably), we cannot directly evaluate OOD generalization to 32 items. To enable OOD evaluation, we continue training $\mathcal{M}_{\text{image-text-limited}}$ on text-only data, expanding the distribution to match the full complexity of the baseline (216 colors, 216 shapes, up to 32 items). This produces our final model $\mathcal{M}_{\text{image-text}}$.

Noise Augmentation (Optional). To further improve generalization, we can apply noise training to $\mathcal{M}_{\text{image-text}}$, producing $\mathcal{M}_{\text{noise-image-text}}$.

E.8. Experimental Setup

We conducted experiments using 4 random seeds. For each seed, we trained separate versions of $\mathcal{M}_{\text{image}}$ using each of the three image encoders (ResNet-152, DINOv3, ViT), resulting in a total of 12 experimental runs (4 seeds $\times$ 3 encoders). We report aggregated performance across 11 valid runs, excluding a single divergent run (Seed 2 with ViT encoder).

As shown in Figure 3, $\mathcal{M}_{\text{image-text}}$ (green curve) exhibits significantly improved performance over the text-only baseline, achieving an average accuracy of 69.5% (SE: 3.3%). This substantial improvement demonstrates that visual training enhances text-based generalization, replicating the phenomenon observed in large-scale VLMs.

E.9. Analysis: Context Length vs. Visual Training

The superior performance of $\mathcal{M}_{\text{image-text}}$ raises a natural question: *Why does visual training improve text-based generalization?* One hypothesis is that the improvement stems from exposure to longer sequences during visual training. Since image encoders produce sequences of patch tokens (e.g., 196 tokens for a 14×14 patch grid), the model encounters a broader range of positional indices compared to text-only training.

To test whether context length exposure alone explains the performance gap, we compare models with and without noise augmentation (described in Section E.4). The noise-augmented text model $\mathcal{M}_{\text{noise-text}}$ achieves 57.5% average OOD accuracy, while the image-text model $\mathcal{M}_{\text{image-text}}$ achieves 69.5%. Applying noise augmentation to the image-text model yields $\mathcal{M}_{\text{noise-image-text}}$ at 83.6%.

Critically, even with noise augmentation, $\mathcal{M}_{\text{noise-text}}$ (57.5%) remains substantially worse than $\mathcal{M}_{\text{image-text}}$ without noise (69.5%). This demonstrates that while exposure to longer positional ranges helps, it is insufficient to fully explain the generalization advantages conferred by visual training. The visual modality must provide qualitatively different inductive biases beyond simple context length expansion.

E.10. Key Findings

The controlled experiments presented in this section reveal several critical insights:

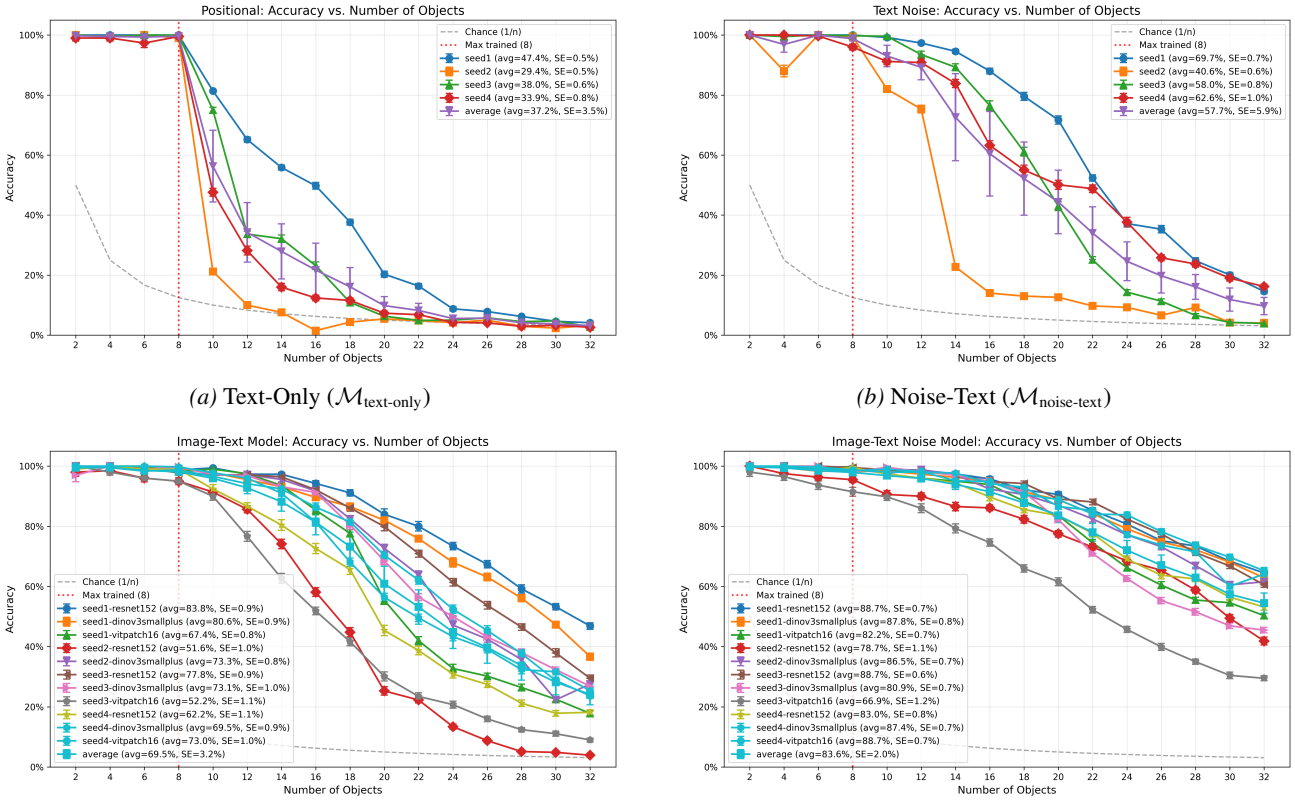
- Text-only models trained on short contexts exhibit poor OOD generalization (37.2% average accuracy on longer sequences).
- Visual training substantially improves generalization (69.5%), replicating the VLM advantage observed in large-scale models.
- Noise augmentation, which exposes models to longer positional ranges, provides substantial improvement to text-only models (57.5%), but remains insufficient to match multimodal performance.
- Even after noise augmentation, the vision-trained model maintains superior performance, suggesting that visual training induces qualitatively different computational mechanisms beyond positional range expansion.

- Combining visual training with noise augmentation yields the strongest performance (83.6%), suggesting complementary regularization effects.

These findings demonstrate that the benefits of multimodal training extend beyond simple exposure to longer sequences. The critical question remains: *What underlying computational mechanism does visual training induce that enables superior generalization?* In the following section, we employ mechanistic interpretability techniques to reveal that the answer lies in a fundamental shift in how models bind and retrieve information—transitioning from brittle positional strategies to robust identity-based binding mechanisms.

F. Detailed Accuracy Sweeps (Scratch Models)

Figure 10 shows individual generalization curves for each scratch model variant. Models trained with meaningful visual input ($\mathcal{M}_{\text{image-text}}$) generalize beyond the training distribution, while text-only and noise-augmented variants show sharp accuracy drops at longer sequence lengths.



(a) Text-Only ($\mathcal{M}_{\text{text-only}}$)

(b) Noise-Text ($\mathcal{M}_{\text{noise-text}}$)

(c) Image-Text ($\mathcal{M}_{\text{image-text}}$)

(d) Noise-Image-Text ($\mathcal{M}_{\text{noise-image-text}}$)

Figure 10. Individual generalization plots for the four scratch model variants. The vertical dashed line indicates the maximum sequence length seen during training (8 items).

G. Detailed Interchange Results

We performed interchange interventions across 4 random seeds for the Text-Only model and 11 runs (across 3 encoders) for the Image-Text model. Figure 11 illustrates the consistency of the mechanism type despite variations in circuit depth.

It is important to note that the model can be marked as "Reflexive" if the answer is already stored in its activations from a previous step. In this context, the Reflexive signal represents a readout of stored information rather than the retrieval process. Therefore, the critical indicator of the binding mechanism is the *first* spike in causal effect.

G.1. Global Binding Score

In subsection 4.6 in the main text, we report a **window-averaged binding score** that averages the symbolic-to-positional ratio across a designated binding window for each model, rather than relying on a single peak layer. This provides a more stable and representative measure of the dominant binding strategy.

Figure 12 shows the per-layer symbolic-to-positional ratio for all evaluated models; shaded regions indicate the exact binding window used for each model. Binding windows were selected by identifying the contiguous region around the peak intervention effect where the symbolic-to-positional ratio remained elevated above the baseline trend for each model. Table 3 lists the exact layer ranges. While the VLM variants consistently show higher ratios than their LLM counterparts, the exact layer of peak effect varies across models. To obtain a robust metric, we compute the average ratio within the designated window for each model. Figure 13 visualizes these window-averaged scores, which are reported in Table 1.

Model	Binding Window
Qwen 2	L19–L26
Qwen 2.5	L19–L26
Qwen 3	L23–L32
InternLM 3	L31–L42

Table 3. **Binding windows used for the window-averaged binding score.** Windows were chosen by inspecting the peak intervention effects.

H. Positional Invariance Experiment Details

To directly test the causal role of translation invariance in driving the binding mechanism shift, we trained models on a modified visual task where object positions were fixed across all images. In the standard visual task, object positions are sampled uniformly at random across the image canvas. In the fixed-position variant, each object type (e.g., red circle, blue triangle) always appears at the same spatial coordinates, eliminating translation invariance at the task level while keeping all other aspects of the training pipeline identical.

We trained four independent models with different random seeds. After training, we evaluated their binding mechanisms using the same interchange intervention protocol described in Section 4 and Appendix G. Table 4 summarizes the results. Because this experiment is limited to four seeds, the observed 2/2 split should be interpreted cautiously.

Seed	Dominant Mechanism
1	Positional
2	Symbolic
3	Positional
4	Symbolic

Table 4. **Binding mechanisms with fixed object positions.** Across four random seeds, two models retained positional binding while two transitioned to symbolic binding, indicating that translation invariance contributes to but does not fully explain the mechanism shift.

The mixed results suggest that pretrained image encoders retain translation-invariant features from their architecture and pretraining, even when spatial variability is reduced at the task level. Consequently, other properties of visual training likely also contribute to the shift toward symbolic binding.

I. Detailed Circuit Analysis

This section provides detailed analysis of how each binding mechanism operates at the circuit level, complementing the main-text presentation in Section 5 with per-seed breakdowns and additional details. Figure 6 and Figure 7 present representative knockout and probe results in the main text; here we provide per-seed visualizations and expanded methodology.

I.1. Methodology

Attention Knockouts. For each source-target token pair, we mask all attention from the source to the target and measure the resulting accuracy drop. High knockout effect indicates that the information flow between those tokens is essential for the task.

Linear Probes. We train lightweight classifiers on residual stream activations to decode position indices, color identity, shape identity, and item identity at each layer and token position. By tracking when and where each type of information becomes decodable, we trace semantic content through the network.

I.2. The Positional Circuit

The text-only model implements binding through position indices rather than semantic content. The knockout and probe results reveal how this works in detail (Figures 14a and 15a).

Step-by-Step Information Flow. The circuit operates through two independent streams that converge at the answer:

1. **Query Stream:** The question token (the color in the query) attends to the matching color in the context section. It copies the *position index* of that context color into its own activations. The `[ANSWER]` token then copies this position from the question token next to it.

2. **Association Stream:** Independently, the shape token in the association section attends to the matching shape in the context section and copies its position index. The item token in the association section then copies this position from the shape token next to it.

3. **Retrieval:** The `[ANSWER]` token looks for the item in the association section that holds the same position index it is holding. When the positions match, retrieval succeeds.

Evidence from Knockouts. The knockout matrix confirms these critical pathways: Question → Context Color (position lookup), Answer → Question (position copy), Association Shape → Context Shape (position lookup), and Association Item → Association Shape (position copy). Notably, there is no critical attention between color and shape tokens—the circuit never routes semantic content between positions.

Evidence from Probes. The probes show position information accumulating at each step: first at the question and association shape tokens (after the lookup), then at the answer and association item tokens (after the copy). Critically, attribute identity (color, shape) remains *undecodable* at entity positions throughout—the model never represents "red circle" as a bound unit. Only position indices are transferred.

This explains why the positional mechanism fails on longer sequences: it relies on position counting, and unfamiliar position indices break the heuristics.

I.3. Symbolic Circuit A (Color-Key)

Image-trained models using the Color-Key variant implement explicit attribute binding (Figures 14b and 15b). Unlike the positional circuit, semantic content—not just position indices—is transferred between tokens.

Step-by-Step Information Flow. The circuit operates through two streams that converge at the answer:

1. **Binding in Context:** The context shape token accumulates the *color identity* from the adjacent context color token. This creates an explicit attribute-entity bundle: the shape now "knows" its color.

2. **Association Stream:** The shape in the association section looks back at the matching shape in the context section and copies the color into its activations. The item in the association section then copies the color from the adjacent shape.

3. **Query Stream:** Independently, the `[ANSWER]` token copies color from the adjacent query (color token).

4. **Retrieval:** The `[ANSWER]` token looks for the item in the association section that has the same color in its activations. When the colors match, retrieval succeeds.

Evidence from Knockouts. The knockout matrix confirms these attribute-routing pathways: Context Shape → Context Color (binding), Association Shape → Context Shape (color retrieval), Association Item → Association Shape (color propagation), and Answer → Query (color copy). These pathways are absent in the positional circuit.

Evidence from Probes. The probes reveal the **binding signature**: color identity first becomes decodable at the context shape token (binding), then at the association shape and item tokens (retrieval and propagation). This confirms that semantic content is being actively moved between tokens.

I.4. Symbolic Circuit B (Shape-Key)

The Shape-Key variant uses a different intermediate representation but achieves the same explicit binding (Figures 14c and 15c).

Step-by-Step Information Flow. The circuit again operates through two streams:

1. **Binding in Context:** As in Circuit A, the context shape token accumulates color identity from the adjacent context color.

2. **Association Stream:** The item in the association section copies the adjacent *shape identity* into its activations.

3. **Query Stream:** Independently, the query (color token) looks back at the context and finds the shape marked with the same color. It copies the *shape identity* into its activations. The `[ANSWER]` token then copies this shape from the adjacent query.

4. **Retrieval:** The `[ANSWER]` token looks for the item in the association section that has the same shape in its activations. When the shapes match, retrieval succeeds.

Evidence from Knockouts. The critical pathways differ from Circuit A: Query → Context Shape is now critical (the query retrieves shape identity), and Association Item → Association Shape (shape propagation).

Evidence from Probes. The binding signature appears with shape identity rather than color. Shape information becomes decodable at the query token after attending to the context shape, then at the `[ANSWER]` token. Despite the different intermediate key, the fundamental property is preserved: semantic content transfers between tokens, enabling length-agnostic retrieval.

I.5. Consistency Across Seeds and Encoders

The patterns are consistent across experimental runs (Figures 16 and 17). All text models exhibit the positional pattern; all image-text models exhibit the symbolic pattern. The specific variant (A vs. B) varies by seed and encoder, but explicit attribute transfer is universal across image-trained runs. This confirms that symbolic binding is a robust emergent property of vision-language training.

I.6. Extension to Large-Scale Models

We extended our analysis to the Qwen model family (Qwen 2, 2.5, and 3). Full circuit analysis via attention knockouts is computationally prohibitive at this scale, so we use linear probes as a proxy: if a model implements symbolic binding, attribute information should become decodable at entity positions.

The VLM variants exhibit the binding signature. Color decodability at entity tokens is consistently higher in VLMs than in their text-only counterparts across the layer ranges where binding occurs. Moreover, the *ordering* of when probes rise coincides with the proposed order of Symbolic Circuit A: color information first becomes decodable at the context shape token, then at the association shape token, reflecting binding in context before propagation through associations. Text-only baselines show low attribute decodability at these positions throughout.

See Appendix L for detailed results. The consistency across model scales suggests symbolic binding is a general property of vision-language training.

J. Attention Knockout Analysis

We present the aggregate attention knockout results in Figure 18, complementing the main-text analysis in Section 5.

K. Linear Probe Analysis

In this section, we provide the aggregate linear probe results across all runs, complementing the main-text analysis in Section 5.

Probe Training Details. We use a standard linear probing setup to measure representational content without introducing additional modeling capacity. Architecture: a single linear layer, $\texttt{nn.Linear}(d_{\text{model}}, \text{num\_classes})$. Training: frozen activations with an 80/20 train–validation split, trained using cross-entropy loss and Adam (learning rate = 0.01) for 100 epochs. Rationale: training independent probes per layer allows us to identify when semantic attributes (e.g., color identity) become linearly separable in the residual stream.

Figure 19 illustrates the average probing accuracy for task-relevant features, confirming that the trends observed in the representative examples (Figure 7) are consistent across random seeds and different image encoders.

L. Qwen Probe Results

We present detailed linear probe results for the Qwen model family. Figure 20 shows color decodability at the context shape token, where binding first occurs. Figure 21 shows color decodability at the association shape token, reflecting propagation of bound attributes. Across all three model generations, VLMs consistently show higher color decodability at entity positions than their text-only counterparts.

M. Related Work (Extended Version)

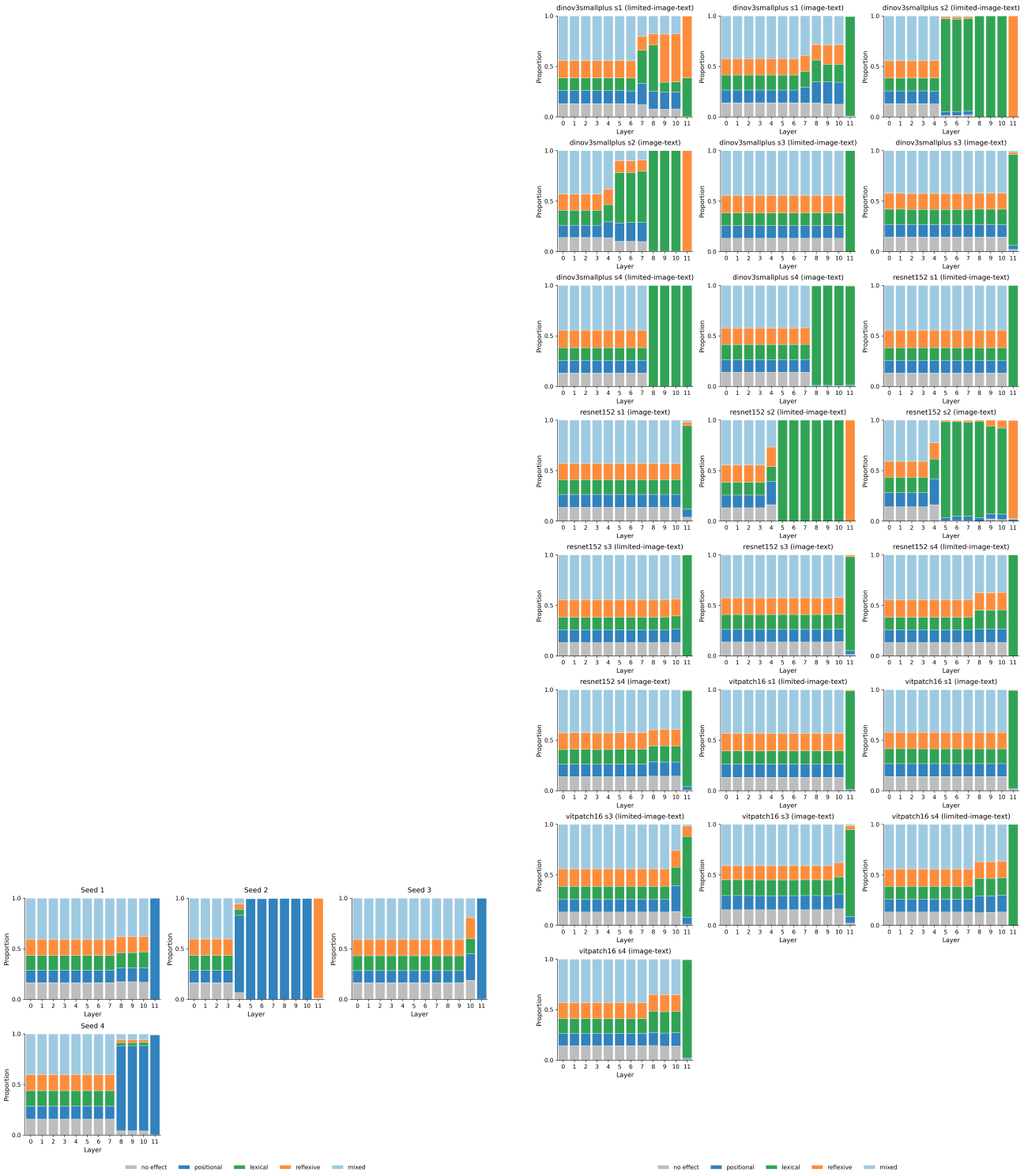
Multi-modal training effects on language models. Multi-modal training has been studied by comparing multi-modal models against their original unimodal backbone language models in order to assess how augmenting them with visual modalities alters their behavior and performance. Several works provide theoretical grounding suggesting that multi-modal training promote robustness and generalization beyond the training distribution, arguing that it discourages reliance on spurious, modality-specific correlations that are over-represented in unimodal training data (Xue et al., 2024) and instead promote more faithful estimation of the underlying latent semantic space (Huang et al., 2021). Empirically, vision–language models (VLMs) have been shown to retain, and in some cases even improve, strong performance on purely textual tasks. For instance, Dai et al. (2024b) demonstrate that, under appropriate training regimes, a VLM can preserve or enhance the text-only capabilities of its language backbone across benchmarks in language understanding, mathematics, coding, and reasoning, while Ratzlaff et al. (2025) reporting similar findings on commonsense reasoning. However, these effects are highly model- and backbone-dependent: Ratzlaff et al. (2025) show that for LLaVA-style models, multi-modal training can either improve or degrade textual performance depending on the underlying LLM and, in some cases, introduce degradation in mathematical reasoning. Complementarily, studies have also examined the impact in the opposite direction, from language to vision. Cooper et al. (2025) find that while pure VLMs outperform hybrid VLM–LLM systems on perceptual tasks such as object and scene recognition, the inclusion of an LLM yields gains on tasks requiring higher-level reasoning or external knowledge.

Mechanistic interpretability Mechanistic interpretability provides a principled framework for analyzing how neural networks implement computations internally, enabling causal rather than purely correlational explanations of model behavior. In this work, we adopt a suite of complementary interpretability tools to characterize and compare the mechanisms underlying the models we study, before and after image training. First, we employ interchange interventions (Meng et al., 2022; Geiger et al., 2021; Finlayson et al., 2021; Vig et al., 2020; Geiger et al., 2020), which causally intervene on hidden states by swapping representations between paired examples, allowing us to identify which internal activations are responsible for specific outputs. This approach is closely related to recent studies on binding and compositional representations in Transformers (Feng & Steinhardt, 2024; Saravanan et al., 2025; Gur-Arieh et al., 2025; Wu et al., 2025), and enables fine-grained analysis of how information is localized and propagated through the network. Second, we use attention knockout techniques (Geva et al., 2023; Gur-Arieh et al., 2025), which selectively zero out attention connections between targeted token pairs, to evaluate how disrupting specific information pathways affects task performance. Finally, we rely on

linear probing methods (Alain & Bengio, 2017; Ravichander et al., 2021; Belinkov, 2022), training lightweight classifiers on hidden representations to assess whether particular concepts or variables are linearly decodable, thereby providing insight into what information is encoded at different layers of the model. Together, these tools allow us to causally and representationally dissect the internal computations of Transformers and to connect observed generalization behavior with concrete underlying mechanisms.

Binding Binding refers to a model's ability to correctly associate entities with their corresponding attributes (Treisman, 1996; Feng & Steinhardt, 2024). Recent work has investigated the internal mechanisms by which Transformers implement binding, typically through controlled retrieval tasks that require recovering an attribute given an entity in the query (Feng & Steinhardt, 2024; Saravanan et al., 2025; Wu et al., 2025; Gur-Arieh et al., 2025; Prakash et al., 2025). These studies intervene on model activations to characterize the information used to form bindings and how these mechanisms evolve during learning, discovering different behaviors that this mechanism shows through training (Wu et al., 2025). Using the taxonomy introduced by Gur-Arieh et al. (2025), prior work, distinct binding strategies in language models have been identified, including positional mechanisms that rely on token order or relative position (Dai et al., 2024a; Prakash et al., 2024; 2025), symbolic (termed lexical in the original work) mechanisms that exploit content-based cues, and reflexive mechanisms that use a self-referential association between tokens (Gur-Arieh et al., 2025). Complementary analyses based on attention patterns further support this view, with Urrutia et al. (2025) providing evidence for both positional and symbolic binding heads in language models trained on retrieval tasks similar to ours. Moreover, detailed studies of attention head behavior reveal the emergence of specialized heads that systematically route information across token positions to facilitate binding operations (Wu et al., 2025).

Length generalization A substantial body of work has demonstrated that neural networks often struggle with length generalization, i.e., extrapolating to input lengths longer than those observed during training, and has introduced benchmarks to systematically evaluate this capability (Lake & Baroni, 2018; Bastings et al., 2018; Ruis et al., 2020; Zhang et al., 2022; Saparov et al., 2023; Zhou et al., 2024a). In the context of Transformers, these limitations have been observed across a variety of settings, including arithmetic tasks such as summing longer numbers (Zhou et al., 2024b), varying the length of chains of reasoning (Zhang et al., 2022), deductive reasoning (Saparov et al., 2023), and general algorithmic problems (Zhou et al., 2024a). Nevertheless, several studies indicate that Transformers can exhibit weak but non-negligible extrapolation under specific conditions. For instance, pre-training has been shown to improve robustness to length extrapolation (Zhang et al., 2022), while architectural choices such as relative positional encodings (Csordás et al., 2021), NoPE (Shen et al., 2023), and the integration of FIRE and randomized positional encodings (Zhou et al., 2024b) can mitigate positional overfitting. Complementarily, carefully designed input representations that better align with the underlying task structure have been shown to facilitate length generalization (Shen et al., 2023; Zhou et al., 2024b). Finally, even minimal exposure to longer sequences during training, through the inclusion of a small number of long examples, can substantially improve extrapolation performance (Jelassi et al., 2023).



(a) All Text-Only Seeds

(b) All Image-Text Runs

Figure 11. **Aggregate Interchange Results.** (a) Across all 4 random seeds, the text-only models consistently converge to a Positional Context mechanism, though the specific layer where this mechanism activates varies (e.g., Layer 5 vs Layer 11). (b) Conversely, models exposed to the visual curriculum consistently develop Symbolic binding mechanisms, regardless of the specific image encoder used (ResNet vs ViT vs DINO).

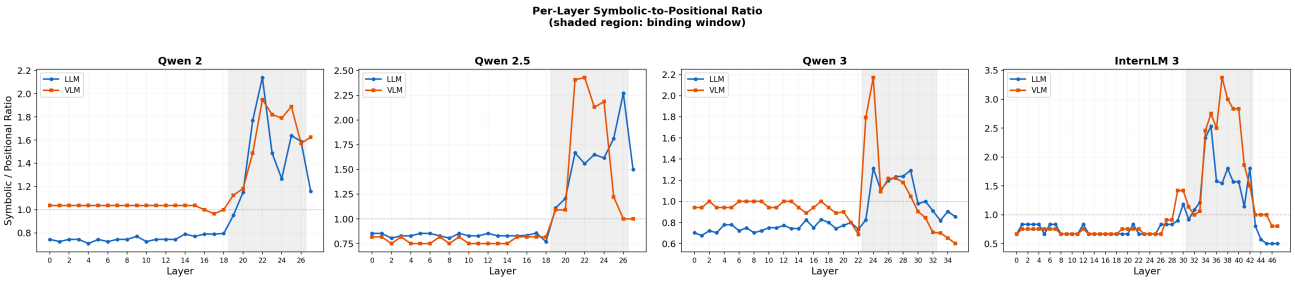


Figure 12. **Per-layer symbolic-to-positional ratio.** Each panel shows the ratio across layers for a model pair. VLM variants (orange) consistently exhibit higher ratios than their LLM counterparts (blue), though the peak layer varies.

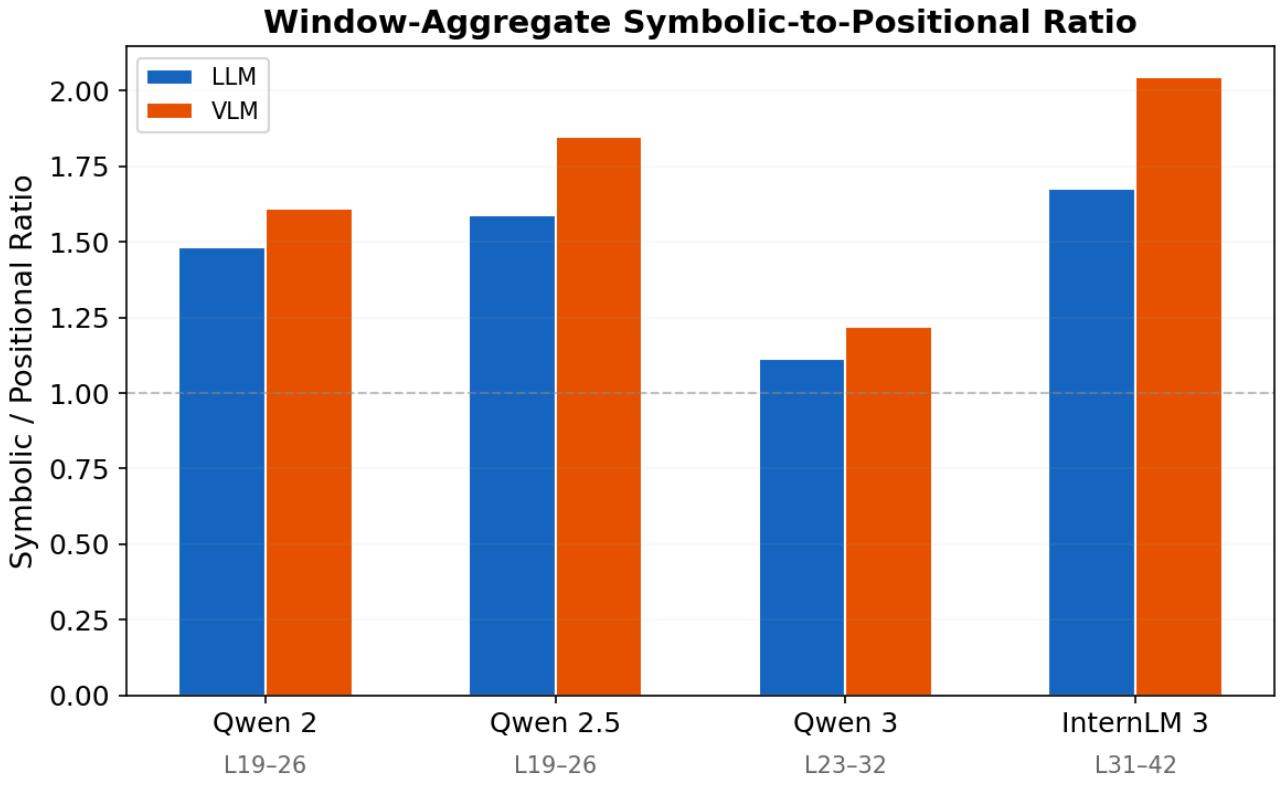
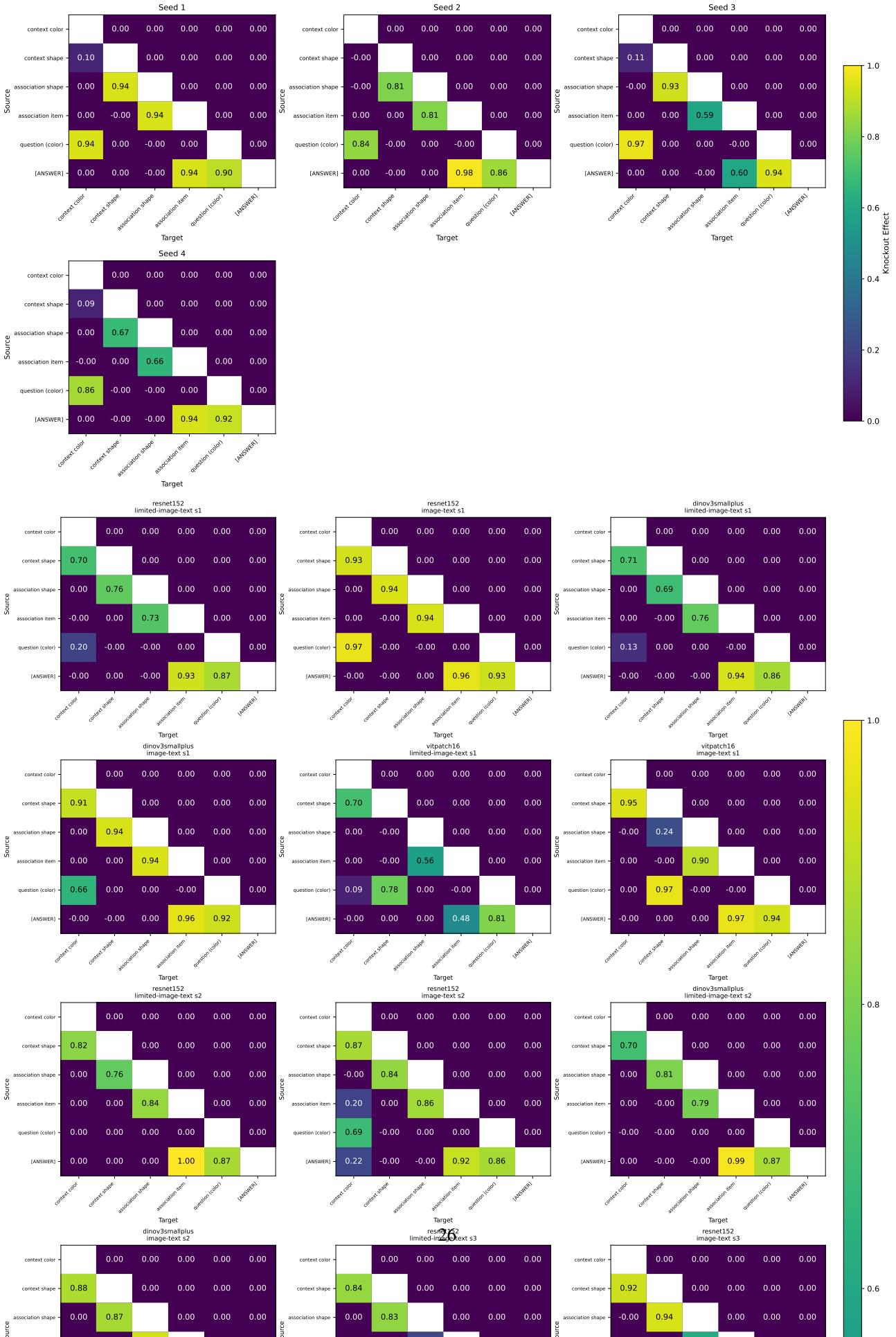
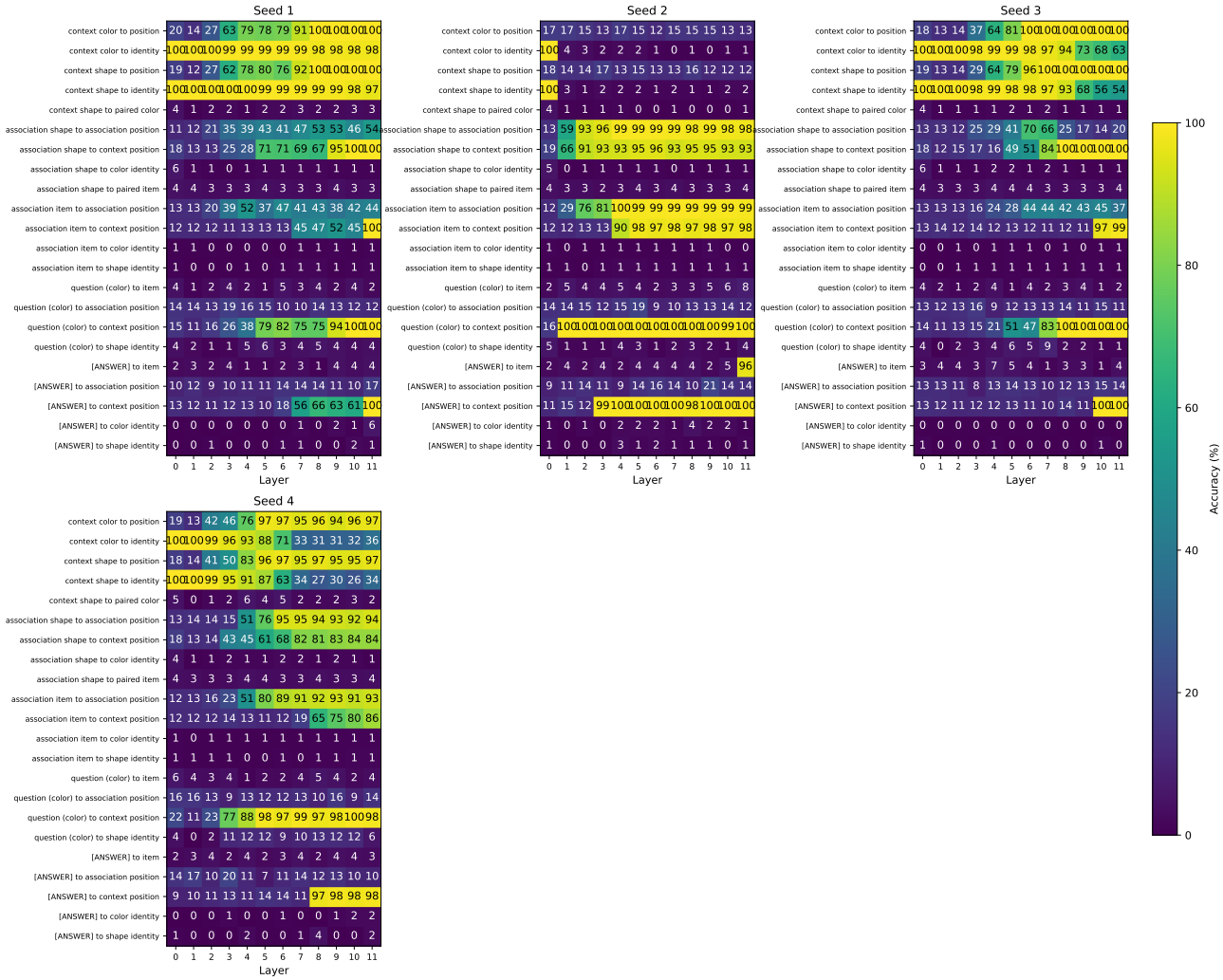
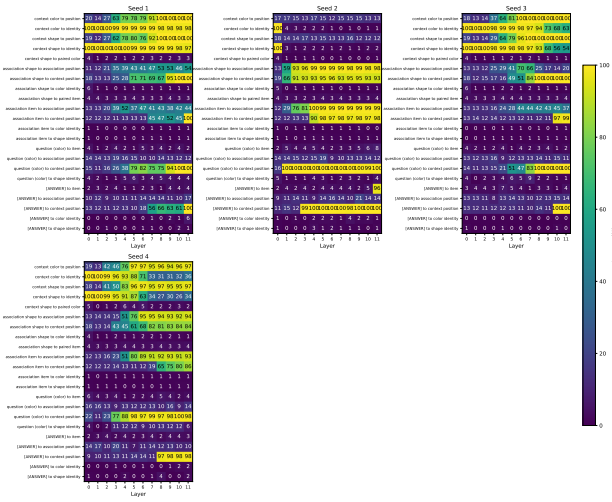


Figure 13. **Window-averaged binding scores.** For each model, we average the symbolic-to-positional ratio across the binding window indicated in Figure 12. This aggregated metric confirms a consistent shift toward symbolic binding in VLM variants across all tested families.

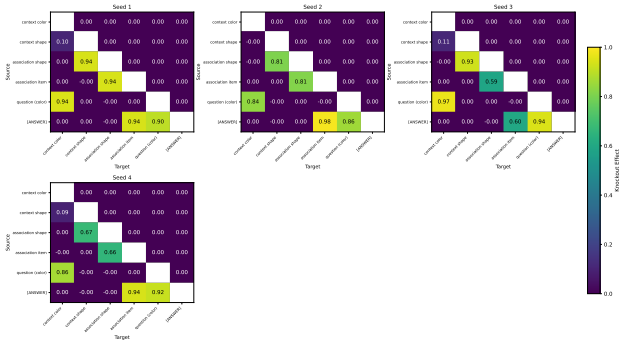

Seed 1
Seed 2
Seed 3
Seed 4
resnet152 limited-image-text s1
resnet152 image-text s1
dinov3smallplus limited-image-text s1
dinov3smallplus image-text s1
vitpatch16 limited-image-text s1
vitpatch16 image-text s1
resnet152 limited-image-text s2
resnet152 image-text s2
dinov3smallplus limited-image-text s2
dinov3smallplus image-text s2
resnet152 limited-image-text s3
resnet152 image-text s3
Source
Target
context color
context shape
association shape
association item
question (color)
[ANSWER]
Knockout Effect


Seed 1
Seed 2
Seed 3
Seed 4
context color to position
context color to identity
context shape to position
context shape to identity
context shape to paired color
association shape to association position
association shape to context position
association shape to color identity
association shape to paired item
association item to association position
association item to context position
association item to color identity
association item to shape identity
question (color) to item
question (color) to association position
question (color) to context position
question (color) to shape identity
[ANSWER] to item
[ANSWER] to association position
[ANSWER] to context position
[ANSWER] to color identity
[ANSWER] to shape identity
0 1 2 3 4 5 6 7 8 9 10 11
Layer
Accuracy (%)
0
20
40
60
80
100


resnet152 limited-image-text s1
resnet152 image-text s1
dinov3smallplus limited-image-text s1
dinov3smallplus image-text s1
vitpatch16 limited-image-text s1
vitpatch16 image-text s1
resnet152 limited-image-text s2
resnet152 image-text s2
dinov3smallplus limited-image-text s2
Source
Target
context color
context shape
association shape
association item
question (color)
[ANSWER]
0.0
0.2
0.4
0.6
0.8
1.0



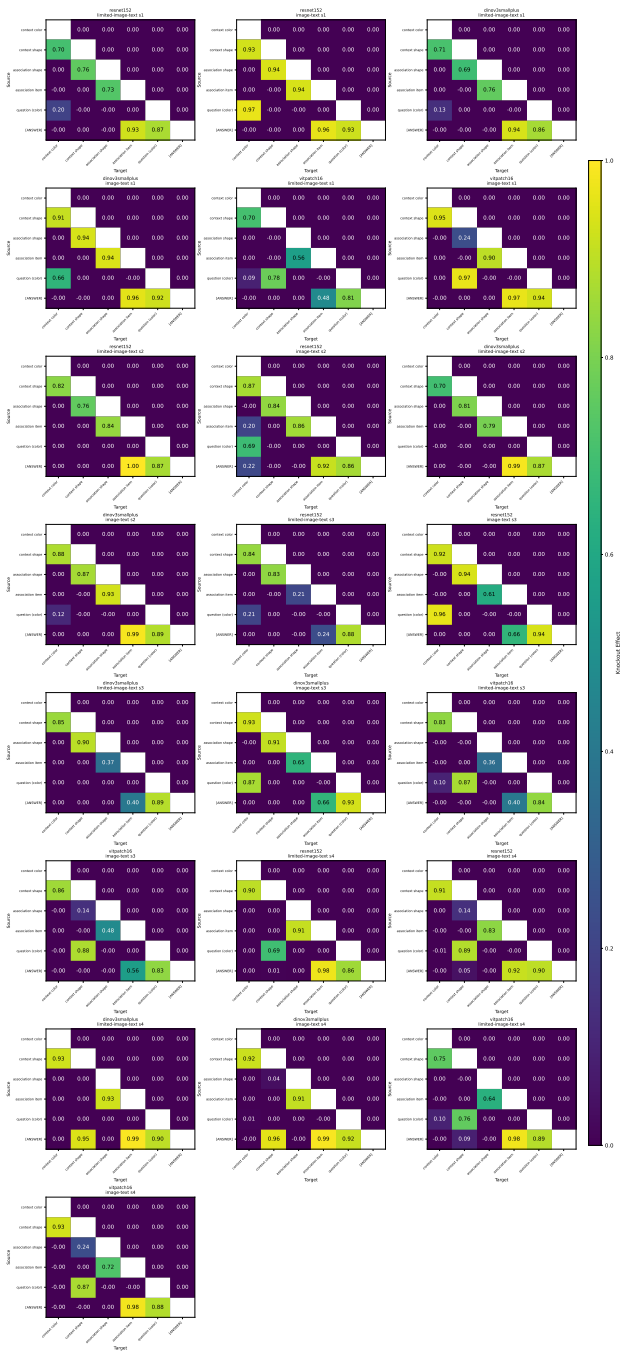
(a) Text-Only - All Seeds (Probes)



(b) Text-Only - All Seeds (Knockouts)

Figure 16. **Aggregate Results: Text Models.** All four seeds show the positional pattern: high position decodability, low attribute decodability at entities, and sparse position-matching knockout matrices.

(a) Image-Text - All Runs (Probes)



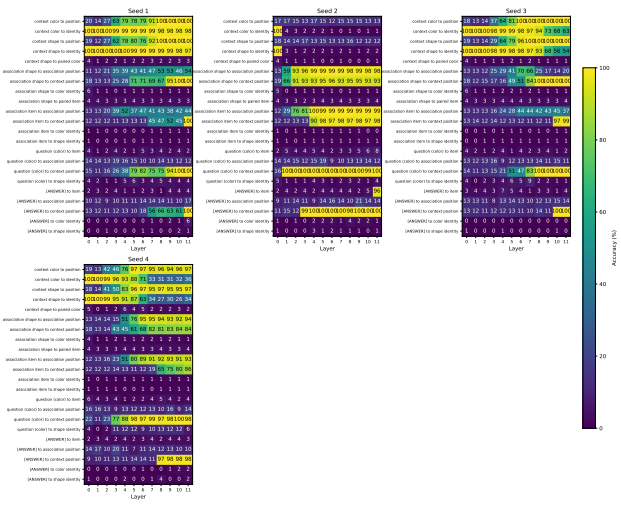
(b) Image-Text - All Runs (Knockouts)

Figure 17. **Aggregate Results: Image-Text Models.** All eleven valid runs (4 seeds × 3 encoders, excluding one divergent run) show the symbolic pattern: binding signature in probes and attribute-routing pathways in knockouts.

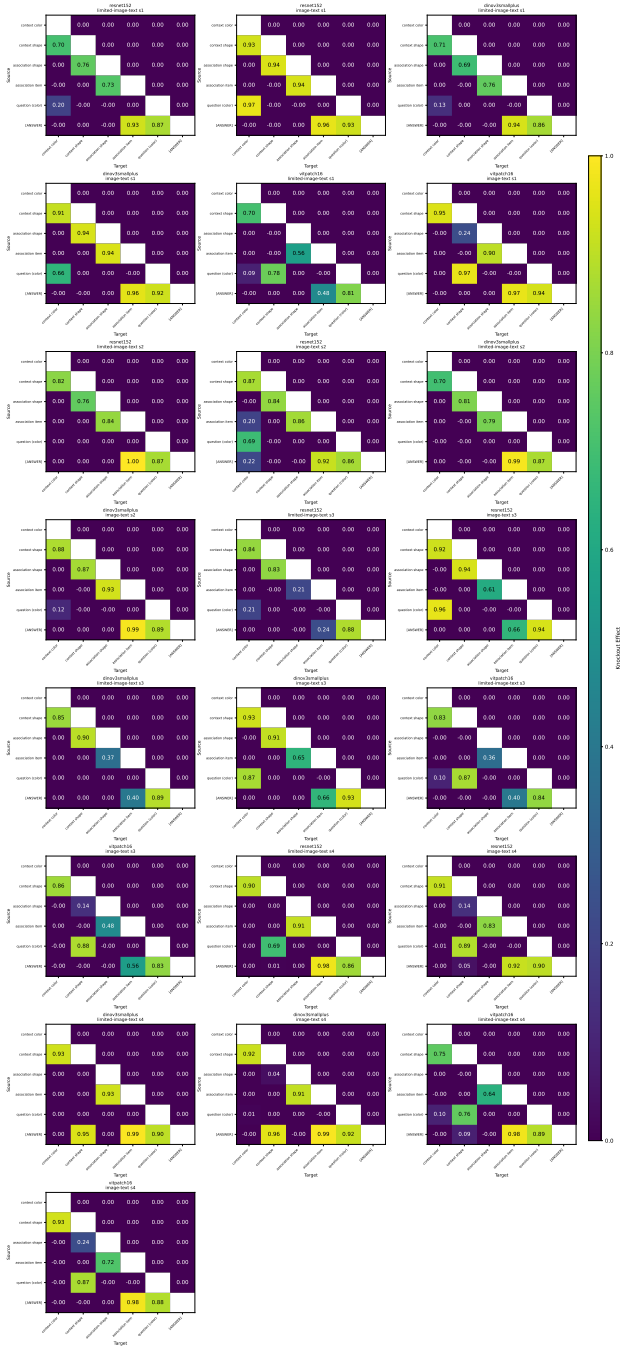
(a) Text-Only ($\mathcal{M}_{\text{text-only}}$) - All Seeds

(b) Symbolic Image-Text ($\mathcal{M}_{\text{image-text}}$) - All Runs

Figure 18. **Aggregate Attention Knockout Analysis.** Average impact of head masking across all experimental runs. The positional text models consistently rely on a single critical circuit, while the symbolic image-text models utilize more distributed mechanisms.

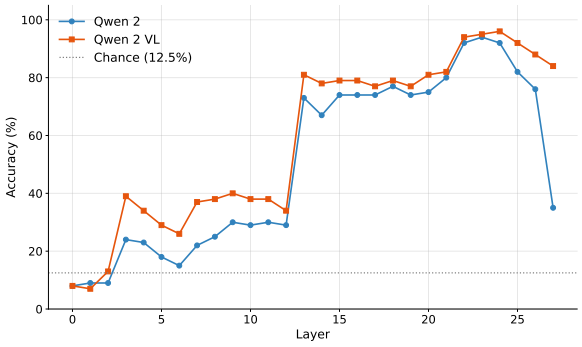


(a) Text-Only ($\mathcal{M}_{\text{text-only}}$) - All Seeds

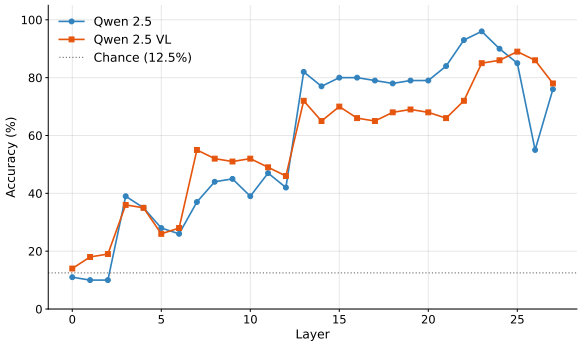


(b) Symbolic Image-Text ($\mathcal{M}_{\text{image-text}}$) - All Runs

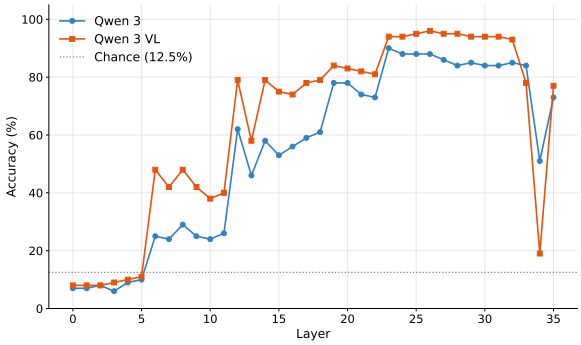
Figure 19. **Aggregate Linear Probe Analysis.** Average probing accuracy across all experimental runs.



(a) Qwen 2

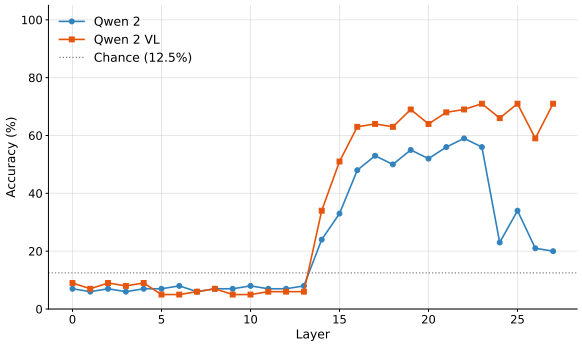


(b) Qwen 2.5

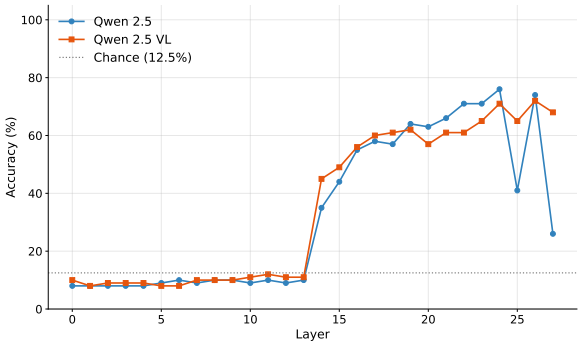


(c) Qwen 3

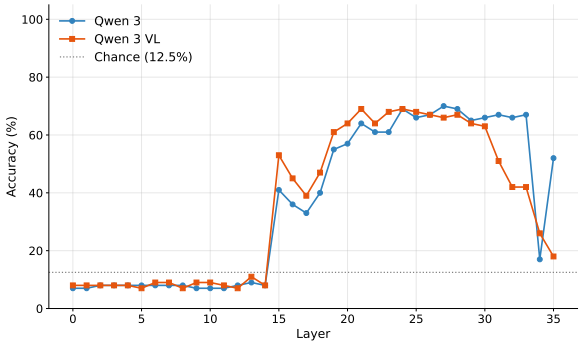
Figure 20. **Color at Context Shape.** Linear probes decoding color identity at the context shape token position. The VLM models show higher color decodability than their text-only counterparts, indicating binding in context.



(a) Qwen 2



(b) Qwen 2.5



(c) Qwen 3

Figure 21. **Color at Association Shape.** Linear probes decoding color identity at the association shape token position. The VLM models show higher color decodability, consistent with color propagation from context to association.