Understanding Counting Mechanisms in Large Language and Vision-Language Models

Supplementary Material

A. Related Work

Counting competence and evaluation instability in LLMs. A growing body of evidence suggests that current LLMs exhibit *fragile* counting ability even on simple tasks. Fu et al. report systematic failures on character-counting queries (e.g., counting occurrences of a letter), showing that errors persist across models and stem from the intrinsic difficulty of repetition-tracking rather than token frequency or exposure [9]. Zhang et al. argue from computational and empirical perspectives that transformer LLMs lack an inherent mechanism for unbounded counting and that common subword tokenization schemes can further degrade performance by obscuring item boundaries [30]. Complementing these findings, Ball et al. show that performance on deterministic tasks such as counting is highly sensitive to seemingly benign prompt/content variations, cautioning against extrapolating from single-prompt evaluations (the *fixed-effect fallacy*) [3]. Orthogonally, Yehudai et al. analyze *when* transformers can count and show that exact counting of a token's frequency in a string is feasible when the model state dimension scales linearly with context length, providing conditions and constructions that clarify capability limits [28]. Together, these works motivate analyses that go beyond aggregate accuracy to *how* numerical information is internally represented and why surface behavior can flip under minor rephrasings.

Mechanistic and causal analyses of numeracy. A parallel line of work uses controlled tasks and interpretability tools to probe internal number mechanisms. Golkar et al. introduce a contextual counting task and show that autoregressive transformers can learn effective counting strategies; intriguingly, removing positional embeddings or using rotary encodings changes where and how the mechanism emerges [11]. Stolfo et al. apply a causal mediation analysis framework to arithmetic reasoning, intervening on layerwise activations to identify mid–late layers and specific components that causally mediate correct versus incorrect arithmetic predictions [22]. In our work, we adopt compatible mediation-style *effect scores* to quantify causal influence of activations on numeric outputs, in spirit similar to their CMA-based scoring. Shah et al. probe numeral embeddings and hidden states, finding cognitive-like magnitude effects (distance, ratio, size), hinting at an emergent "mental number line" [21]. Our work builds on these insights but focuses specifically on *counting* and contributes a unified causal picture across both text-only LLMs and LVLMs: we show that latent count states are (i) primarily stored in contextual activations, (ii) concentrate in the final token/region, (iii) emerge progressively across layers, and (iv) are *linearly additive* and transferable across contexts.

Counting in LVLMs and the role of vision. In multimodal models, counting introduces additional challenges due to visual encoding, spatial aggregation, and modality fusion. Paiss et al. demonstrate that CLIP—despite strong retrieval—often ignores cardinality; they improve counting via a counting-aware contrastive loss and show broader attentional coverage after fine-tuning [19]. Guo et al. expose striking failures in *compositional* counting (multiple object types), indicating unreliable binding between category and quantity [12]. *LLaVA-Interp* analyzes LLaVA's *visual* tokens with Logit Lens and ablations, showing that visual representations increasingly align with vocabulary space across layers and can project onto content-descriptive tokens (including numerals), suggesting image-derived numeric cues may surface in LM-space and relate to counting behavior [17]. Consistent with these observations, we find that LVLMs' count evidence often resides in both foreground and *background* visual tokens and is more sensitive to layout, density, and resolution than in text-only models. Our causal patching further shows that, unlike text, the most informative visual region is not always tied to the last object, reflecting pre-decoder global integration in the vision stack.

Causal intervention toolkit and patch-based probes. Methodologically, our study connects to activation-level interpretability: activation/mean/interchange patching, key–value (attention) interventions, and mediation analysis; and to layerwise readout probes such as the original *Logit Lens* and its learned variant *Tuned Lens* [4, 18]. *Patchscopes* unifies patching/inspection configurations and leverages stronger models to explain internal representations in natural language, enabling expressive cross-layer analyses [10]. We introduce **CountScope**, a lightweight target-context probe tailored to numerical decoding, enabling precise, layerwise and token/patch-level localization of latent count states in both LLMs and LVLMs. Beyond corroborating prior observations (e.g., layerwise emergence, shortcutting on separators), our causal experiments uncover (a) *internal latent counters* that update across items and transfer across contexts, (b) *type-specific counters* that can *reset* in polytypic lists, and (c) a *max-latent-count* effect whereby the model's final prediction often reflects the maximum latent count present in context rather than a simple sum.

B. Task Details

The textual dataset is built from simple lists of item names and short counting questions. Items are sampled uniformly from a fixed vocabulary of common fruits (apple, orange, peach, fig, mango, pear, coconut, cherry, plum). Lists range from length 1 to 9. We use four prompt configurations: monotypic lists, polytypic lists, list-first (also called question-last) prompts, and question-first prompts. Both total-count and type-specific questions are included. The main templates used for our experiments are shown below.

Monotypic • Question-first

```
Question: How many items are there
in the following sentence?
apple, apple, apple, apple, apple
```

Polytypic • Question-first

```
Question: How many apples are there
in the following sentence?
apple, peach, orange, pear, apple
```

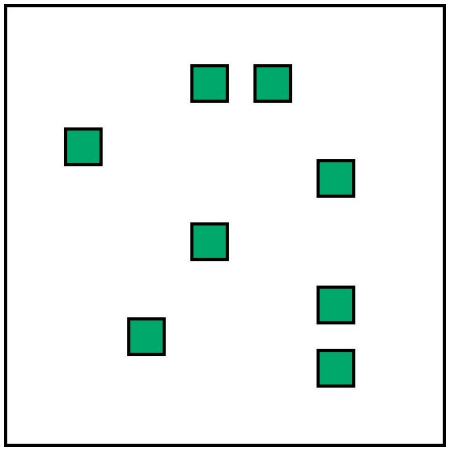
Monotypic • Question-last

```
apple, apple, apple, apple, apple
Question: How many items are there
in the above sentence?
```

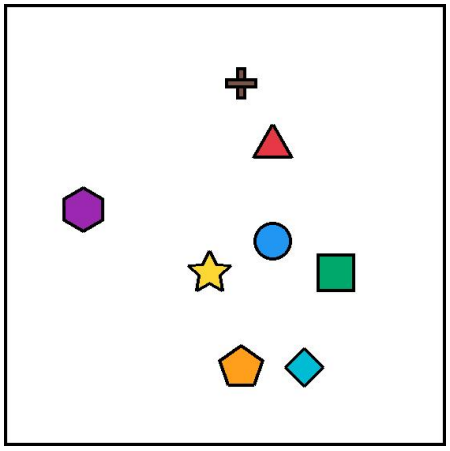
Polytypic • Question-last

```
apple, peach, fig, apple, mango
Question: How many apples are there
in the above sentence?
```

The visual dataset consists of synthetic images containing one or more colored geometric shapes. Shapes are drawn from a fixed set (circle, triangle, square, pentagon, hexagon, star, diamond, cross, heart), and colors are drawn from (blue, green, red, yellow, orange, brown, purple and cyan). Each image contains between 1 and 9 shapes. Objects are placed at random non-overlapping positions on a uniform background using a simple sampling-and-rejection algorithm. Both monotypic and polytypic configurations are created by controlling shape–color combinations. Representative examples for each configuration are shown in Figure 8. For causal analysis, we set the object size equal to the patch size of the model to avoid the confounding effect of multiple-patch aggregation. Since visual tokens always precede prompt tokens in LVLMs, the visual setup typically matches the question-last condition; however, by using a task description in the system prompt, we can approximate the question-first condition in visual experiments.



(a) Monotypic



(b) Polytypic-Unique

Figure 8. Example images from the visual dataset. Monotypic images contain repeated instances of one shape–color pair, while polytypic images contain mixtures. The prompt used for image queries is: "`How many objects are there in the image?`"

C. Behavioral Characterization of Counting

We begin by quantifying the counting accuracy of LLMs and LVLMs across all experimental configurations. Table 5 reports the performance of two LLMs (Qwen2.5, Llama3) and two LVLMs (Qwen2.5-VL, InternVL3.5) on textual counting tasks across category types, ordering conditions, and question types. All models are of similar size (7-8B parameters). Overall, LVLMs match or exceed LLM performance even though the tasks contain no visual input. Across all models, the polytypic-replicate setting is the most difficult, with clear drops in accuracy relative to monotypic and polytypic-unique. The polytypic-unique setting remains stable and often approaches monotypic accuracy. Accuracy differences between question-last and question-

Model	Category	Question-last		Question-first	
		Specific	General	Specific	General
Qwen2.5	Monotypic	91.67	91.67	91.67	91.67
	Polytypic-Replicate	88.13	91.96	88.34	70.90
	Polytypic-Unique	57.71	100.00	89.13	100.00
Llama3	Monotypic	94.44	94.44	94.44	94.44
	Polytypic-Replicate	94.35	6.23	89.85	1.41
	Polytypic-Unique	100.00	99.23	78.38	100.00
Qwen2.5-VL	Monotypic	100.00	100.00	100.00	100.00
	Polytypic-Replicate	92.39	40.21	80.24	48.81
	Polytypic-Unique	100.00	100.00	100.00	100.00
InternVL3.5	Monotypic	100.00	100.00	100.00	100.00
	Polytypic-Replicate	99.36	56.05	99.77	64.58
	Polytypic-Unique	100.00	98.46	100.00	100.00

Table 5. Average accuracy of LLMs and LVLMs on textual counting tasks across category types, ordering conditions, and question types. In the Polytypic-Replicate setting, some items are repeated (e.g., "apple, apple, orange"), while in the Polytypic-Unique setting, each item type appears only once (e.g., "apple, peach, orange").

Model	Category	Question-Last		Question-First	
		Specific	General	Specific	General
Qwen2.5-VL	Monotypic	30.56	41.67	41.67	44.44
	Polytypic-Replicate	76.75	62.26	73.72	65.48
	Polytypic-Unique	83.03	76.15	87.69	81.54
InternVL3.5	Monotypic	94.44	91.67	69.44	55.56
	Polytypic-Replicate	86.89	81.07	71.70	75.19
	Polytypic-Unique	79.21	93.85	79.69	96.15

Table 6. Accuracy of LVLMs on visual counting tasks under different category types, ordering conditions, and question types.

first formats are small. Specific questions show higher accuracy than general questions in polytypic conditions, which is mainly due to the smaller number of valid target types in the specific setting. Among all models, InternVL3.5 and Qwen2.5 show the strongest overall performance.

Table 6 summarizes LVLM behavior on visual counting tasks. Accuracy is lower than in textual tasks for the same numeric range. In contrast to the textual setting, both LVLMs show cases where polytypic inputs, including the replicate condition, outperform monotypic inputs, especially for Qwen2.5-VL. There is no consistent advantage for specific or general questions across models or ordering conditions. InternVL3.5 is stronger overall than Qwen2.5-VL in both question-first and question-last formats. For LVLMs, the question-first setup corresponds to placing the task description in the system prompt, which helps Qwen2.5-VL in several settings but does not produce a uniform pattern across models.

Table 7 evaluates how image resolution, object size, and object density affect visual counting accuracy in InternVL3.5 and Qwen2.5-VL. The two models show opposite trends with respect to resolution: for Qwen2.5-VL, increasing the image size lowers accuracy, while InternVL3.5 benefits from higher resolution. This difference likely reflects model-specific optimal input resolutions. Both models improve substantially as object size increases, indicating that larger items provide clearer visual evidence for counting. Accuracy is consistently higher in the sparse setting than in the dense setting for all object sizes and both resolutions, suggesting that limited background area makes counting more difficult.

Model	Resolution	Size 14		Size 28		Size 56	
		Sparse	Dense	Sparse	Dense	Sparse	Dense
InternVL3.5	280×280	46.20	43.58	67.22	64.38	74.22	67.69
	560×560	53.85	57.18	73.50	74.51	80.17	76.47
Qwen2.5-VL	280×280	55.43	46.09	70.28	74.52	75.92	69.08
	560×560	33.24	30.94	66.10	63.87	69.55	66.95

Table 7. Accuracy by image resolution, object size, and density of objects for InternVL3.5 and Qwen2.5-VL.

Model	Category	Question-last				Question-first			
		Various	Less	More	None	Various	Less	More	None
Qwen2.5	Monotyp.	8.33	55.56	8.33	97.22	8.33	13.89	16.67	100
	Poly.-Rep.	5.53	8.29	11.95	27.43	6.32	8.04	16.75	31.61
	Poly.-Uni.	17.69	22.30	76.57	75.38	32.30	27.69	61.21	76.15
Llama3	Monotyp.	16.67	8.33	55.56	19.44	11.11	11.11	44.44	25.00
	Poly.-Rep.	1.95	0.66	11.02	0.63	0.41	0.14	0.84	0.00
	Poly.-Uni.	40.00	14.62	76.15	30.77	43.08	3.85	93.85	6.15

Table 8. Performance of Qwen2.5 and Llama3 under different separator conditions, including Various (some commas replaced with random separators), Less (some commas deleted), More (some commas repeated), and None (no separators).

We finally assess how different separator conditions impact counting accuracy in Table 8. In all configurations, counting accuracy drops significantly when separators are altered or removed. This effect is observed in both question-last settings, where models must infer task instructions from context, and question-first settings, where the task is explicitly defined. The Polytypic-Replicate setting is particularly sensitive to changes in separators, likely due to the more complex composition of items. Interestingly, in the Monotypic setting, Qwen2.5 performs well even without separators (None condition), suggesting it can rely on the inherent structure of repeated items. In contrast, Llama3 shows much lower accuracy in the None condition, highlighting its greater reliance on separators, even when items are repetitive.

	LLM (Textual)		LVLM (Visual)	
Count	Context masked	Question masked	Image masked	Prompt masked
1	0.0000	0.0000	0.9539	0.0187
2	0.0654	0.0002	0.9691	0.0167
3	0.6832	0.0000	0.9535	0.0304
4	0.9806	0.0000	0.8789	0.0617
5	0.5606	0.0000	0.7605	0.0714
6	0.8135	0.0000	0.6363	0.1119
7	0.7917	0.0000	0.5181	0.0617
8	0.9497	0.2563	0.4916	0.0812
9	0.9862	0.0008	0.4515	0.0330

Table 9. Mean drop in the probability of the ground-truth count under offline zero patching. For each count, we report the drop when masking text-only context vs. question tokens in the LLM, and image patches vs. prompt tokens in the LVLM. Higher values indicate that masking removes more count-relevant information from that part of the input.

#Objects	Region	3x3 Patches	6x6 Patches	10x10 Patches
1	Foreground	0.4333	0.4392	0.4715
1	Background	0.4980	0.5828	0.6255
2	Foreground	0.5827	0.5670	0.5481
2	Background	0.5864	0.7141	0.7171
3	Foreground	0.5815	0.5715	0.4979
3	Background	0.4826	0.6746	0.6892
4	Foreground	0.4629	0.3937	0.3219
4	Background	0.3900	0.5568	0.5923
5	Foreground	0.3865	0.3516	0.2993
5	Background	0.3343	0.4744	0.4899

Table 10. Per-object breakdown (1–5 objects) of the average CountScope probability for foreground and background patches across different patch sizes.

D. Additional Causal Mediation Analysis

Here, we provide additional details of the experiments conducted for causal mediation analysis. Table 9 reports the mean drop in the probability of the ground-truth count after offline zero patching of context and question (for LLMs) or image and prompt (for LVLMs). The results confirm that count-related information is primarily stored in the context for both model types. Table 10 provides a per-object breakdown of CountScope probabilities, revealing how the strength of count signals in foreground and background patches varies with the number of objects and patch size. Finally, Figure 9 illustrates how the count information is distributed across layers under offline patch interchange, with background patches showing stronger signals in the early layers, and foreground patches becoming more informative in deeper layers. The diagrams also reveal that count information is mainly stored in the middle-to-late layers.

The *continued counting* hypothesis is evaluated across both visual and textual tasks. Table 11 presents the expected predictions based on this hypothesis for various values of k. Table 12 reports the performance of visual models (InternVL3.5 and Qwen2.5-VL) for different values of k in both "Question First" and "Question Last" setups. For the textual tasks, the results are shown in Tables 13 and

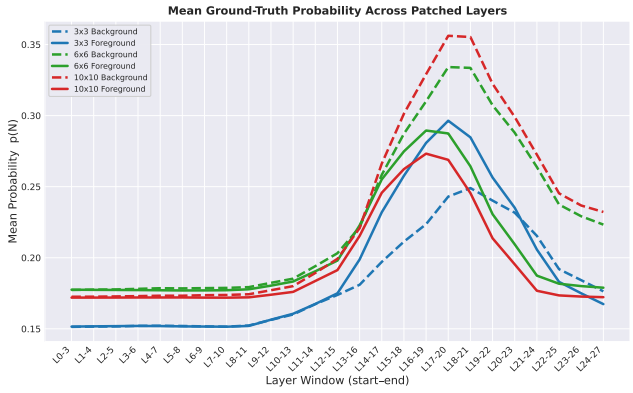


Figure 9. Mean ground-truth probability across layer windows under offline patch interchange. At each step, a window of four consecutive layers was interchanged, and the resulting ground-truth probability was measured. The reported values are averaged over images containing one to five objects. Each color corresponds to a specific image size (3×3, 6×6, and 10×10), while solid and dashed lines distinguish foreground and background interchange, respectively.

src / trgt	2	3	4	5	6	7	8	9
2	3	4	5	6	7	8	9	10
3	4	5	6	7	8	9	10	11
4	5	6	7	8	9	10	11	12
5	6	7	8	9	10	11	12	13
6	7	8	9	10	11	12	13	14
7	8	9	10	11	12	13	14	15
8	9	10	11	12	13	14	15	16
9	10	11	12	13	14	15	16	17

(a) $k = 1$

src / trgt	2	3	4	5	6	7	8	9
2	2	3	4	5	6	7	8	9
3	3	4	5	6	7	8	9	10
4	4	5	6	7	8	9	10	11
5	5	6	7	8	9	10	11	12
6	6	7	8	9	10	11	12	13
7	7	8	9	10	11	12	13	14
8	8	9	10	11	12	13	14	15
9	9	10	11	12	13	14	15	16

(b) $k = 2$

src / trgt	2	3	4	5	6	7	8	9
2	1	2	3	4	5	6	7	8
3	2	3	4	5	6	7	8	9
4	3	4	5	6	7	8	9	10
5	4	5	6	7	8	9	10	11
6	5	6	7	8	9	10	11	12
7	6	7	8	9	10	11	12	13
8	7	8	9	10	11	12	13	14
9	8	9	10	11	12	13	14	15

(c) $k = 3$

Table 11. Expected answer, $\tilde{r}$, under the continued counting hypothesis. The value is computed as $\tilde{r} = N_{\text{source}} + N_{\text{target}} - k$, where N_{source} and N_{target} denote the number of objects in the source and target inputs, respectively, and k is the number of patched items.

		InternVL3.5			Qwen2.5-VL		
Question Type	**K**	**P($\tilde{r}$)**	**P(r')**	**CI**	**P($\tilde{r}$)**	**P(r')**	**CI**
Question-First	1	0.25	0.42	0.24	0.32	0.46	0.32
	2	0.39	0.15	0.43	0.48	0.40	0.42
	3	0.45	0.08	0.50	0.53	0.27	0.49
Question-Last	1	0.36	0.54	0.29	0.24	0.56	0.23
	2	0.56	0.16	0.57	0.49	0.38	0.43
	3	0.53	0.11	0.59	0.54	0.27	0.49

Table 12. Average probabilities $\Pr(\tilde{r})$, $\Pr(r')$, and CI scores for the continued-counting hypothesis across different values of K and prompt structures in LVLMs. In the question-first setting, the counting task is specified in the system prompt.

		Llama3			Qwen2.5		
K	**Patch Type**	**P($\tilde{r}$)**	**P(r')**	**CI**	**P($\tilde{r}$)**	**P(r')**	**CI**
1	Both	0.03	0.26	0.33	0.00	0.54	0.23
	Separators	0.01	0.88	0.00	0.00	1.00	0.00
	Elements	0.03	0.26	0.33	0.00	0.54	0.23
2	Both	0.70	0.01	0.78	0.82	0.00	0.91
	Separators	0.41	0.37	0.45	0.23	0.74	0.24
	Elements	0.17	0.19	0.42	0.58	0.02	0.78
3	Both	0.86	0.19	0.67	1.00	0.28	0.72
	Separators	0.69	0.26	0.55	0.57	0.69	0.30
	Elements	0.35	0.30	0.36	0.81	0.29	0.62

Table 13. Average probabilities $\Pr(\tilde{r})$, $\Pr(r')$, and CI scores for the continued-counting hypothesis and question-first setting across different values of K and patch types in LLMs.

14, where Llama3 and Qwen2.5 are evaluated with different patch types (both, separators, and elements) under "Question-First" and "Question-Last" conditions. These tables report the average probability of the predicted answer ($\tilde{r}$), the average probability of the incorrect answer (r'), and the CI score for each configuration.

The *maximum latent count hypothesis* is tested across both visual and textual tasks. Table 15 presents the expected predictions based on this hypothesis for varying values of k. Table 16 reports the performance of visual models (InternVL3.5 and Qwen2.5-VL) across different values of k and in both "Question First" and "Question Last" configurations. For textual tasks, the results are shown in Tables 17 and 18, where Llama3 and Qwen2.5 are evaluated with various intervention types (both, separators, and elements). These tables present the average probability of the predicted answer ($\tilde{r}$), the average probability of the incorrect answer (r'), and the CI score for each configuration.

To assess the generalizability of the linear additivity effect across tasks, we apply position-difference vectors

		Llama3			Qwen2.5		
K	**Patch Type**	**P($\tilde{r}$)**	**P(r')**	**CI**	**P($\tilde{r}$)**	**P(r')**	**CI**
1	Both	0.01	0.01	0.22	0.16	0.29	0.11
	Separators	0.03	0.47	0.00	0.05	0.40	0.00
	Elements	0.01	0.01	0.22	0.16	0.29	0.11
2	Both	0.00	0.00	0.21	0.10	0.10	0.17
	Separators	0.17	0.06	0.26	0.17	0.22	0.14
	Elements	0.01	0.01	0.21	0.37	0.22	0.24
3	Both	0.12	0.12	0.15	0.23	0.17	0.17
	Separators	0.29	0.16	0.22	0.30	0.24	0.16
	Elements	0.13	0.13	0.15	0.40	0.24	0.21

Table 14. Average probabilities $\Pr(\tilde{r})$, $\Pr(r')$, and CI scores for the continued-counting hypothesis and question-last setting across different values of K and patch types in LLMs.

src / trgt	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
2	2	2	3	4	5	6	7	8
3	3	3	3	4	5	6	7	8
4	4	4	4	4	5	6	7	8
5	5	5	5	5	5	6	7	8
6	6	6	6	6	6	6	7	8
7	7	7	7	7	7	7	7	8
8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9

(a) $k = 1$

src / trgt	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
2	2	2	2	3	4	5	6	7
3	3	3	3	3	4	5	6	7
4	4	4	4	4	4	5	6	7
5	5	5	5	5	5	5	6	7
6	6	6	6	6	6	6	6	7
7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9

(b) $k = 2$

src / trgt	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
2	2	2	2	2	3	4	5	6
3	3	3	3	3	3	4	5	6
4	4	4	4	4	4	4	5	6
5	5	5	5	5	5	5	5	6
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9

(c) $k = 3$

Table 15. Expected answer, $\tilde{r}$, under the maximum latent count hypothesis. The value is computed as $\tilde{r} = \max\big(N_{\text{source}},\ N_{\text{target}} - k\big)$, where N_{source} and N_{target} denote the number of objects in the source and target inputs, respectively, and k is the number of patched items.

learned (averaged over dataset) from a fruit-counting task to an animal-counting task. Table 19 reports the CI scores and accuracies for this transfer experiment, showing that the

		InternVL3.5			Qwen2.5-VL		
Question Type	**K**	$\mathbf{P}(\tilde{r})$	$\mathbf{P}(r')$	**CI**	$\mathbf{P}(\tilde{r})$	$\mathbf{P}(r')$	**CI**
Question First	1	0.39	0.17	0.42	0.74	0.18	0.76
	2	0.49	0.08	0.54	0.80	0.02	0.87
	3	0.58	0.04	0.61	0.86	0.01	0.90
Question Last	1	0.57	0.25	0.53	0.64	0.25	0.66
	2	0.73	0.10	0.73	0.77	0.02	0.84
	3	0.75	0.07	0.76	0.83	0.02	0.88

Table 16. Average probabilities $\Pr(\tilde{r})$, $\Pr(r')$, and CI scores for the maximum latent count hypothesis across different values of K and prompt structures in LVLMs. In the question-first setting, the counting task is specified in the system prompt.

		Llama3			Qwen2.5		
K	**Patch Type**	$\mathbf{P}(\tilde{r})$	$\mathbf{P}(r')$	**CI**	$\mathbf{P}(\tilde{r})$	$\mathbf{P}(r')$	**CI**
1	Both	0.28	0.48	0.31	0.59	0.38	0.60
	Separators	0.03	0.85	0.00	0.00	1.00	0.00
	Elements	0.28	0.48	0.31	0.59	0.38	0.60
2	Both	0.66	0.01	0.74	0.91	0.00	0.95
	Separators	0.49	0.16	0.58	0.34	0.44	0.45
	Elements	0.26	0.42	0.34	0.52	0.43	0.54
3	Both	0.80	0.01	0.82	0.95	0.00	0.97
	Separators	0.51	0.10	0.63	0.43	0.34	0.54
	Elements	0.24	0.37	0.36	0.53	0.36	0.58

Table 17. Average probabilities $\Pr(\tilde{r})$, $\Pr(r')$, and CI scores for the maximum latent count hypothesis and question-first setting across different values of K and patch types in LLMs.

		Llama3			Qwen2.5		
K	**Patch Type**	$\mathbf{P}(\tilde{r})$	$\mathbf{P}(r')$	**CI**	$\mathbf{P}(\tilde{r})$	$\mathbf{P}(r')$	**CI**
1	Both	0.15	0.44	0.04	0.13	0.33	0.09
	Separators	0.12	0.49	0.00	0.05	0.43	0.00
	Elements	0.15	0.44	0.04	0.13	0.33	0.09
2	Both	0.34	0.11	0.32	0.31	0.13	0.28
	Separators	0.42	0.17	0.33	0.32	0.17	0.26
	Elements	0.11	0.38	0.07	0.13	0.30	0.11
3	Both	0.31	0.08	0.32	0.33	0.11	0.30
	Separators	0.36	0.14	0.32	0.32	0.15	0.28
	Elements	0.12	0.30	0.11	0.21	0.24	0.18

Table 18. Average probabilities $\Pr(\tilde{r})$, $\Pr(r')$, and CI scores for the maximum latent count hypothesis and question-last setting across different values of K and patch types in LLMs.

additivity pattern holds when applying the learned vectors to the animal counting task, with consistent performance across different position-difference values.

K	**CI (Animals)**	**Acc (Animals)**
1	0.56 ± 0.23	0.54 ± 0.23
2	0.64 ± 0.16	0.58 ± 0.12
3	0.81 ± 0.18	0.78 ± 0.20
4	0.87 ± 0.21	0.81 ± 0.29
Avg	0.72 ± 0.20	0.68 ± 0.21

Table 19. Linear additivity under task transfer. CI scores and accuracies when position-difference vectors are estimated from a fruit-counting task and then applied to an animal-counting task. We report mean $\pm$ standard deviation over animal types for different position differences K.

Category	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Monotypic	0.26	0.54	0.83	1.00	1.00	0.97	1.00
Polytypic	0.52	0.53	0.88	1.00	1.00	1.00	1.00

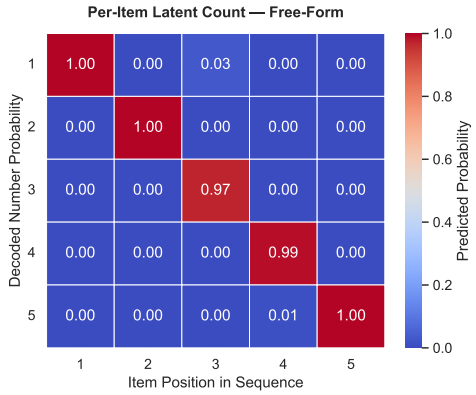
Table 20. Target Probability Drop for Causal Patching of Separator Activations. The drop in target probability is calculated when the activations of the first separator are patched and transferred to subsequent separators across different counts, in both monotypic and polytypic settings.

Table 20 shows the target probability drop when the activations of the first separator are causally patched across all layers and transferred to subsequent separators, comparing monotypic and polytypic settings for different counts.

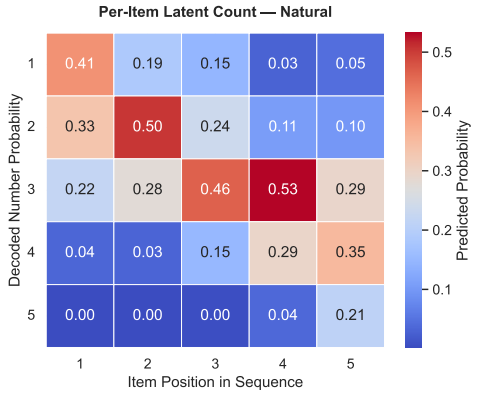
E. Natural Data Analysis

In this paper, we focus on synthetic data samples for causal analysis, as they provide a controllable setting and reliable evidence, which is standard in mechanistic interpretability studies. To test generalization beyond synthetic settings, we conduct additional experiments following Section 4.3 for both modalities. For text, we use unstructured contexts with free-form sentence templates containing multiple occurrences of target words. These contexts follow a short naturalistic prose style, such as notes, reports, or transcripts, where repeated target words are embedded in fluent sentences rather than presented as a plain list. A representative template is: "Short report: Keyword: [WORD]. Repeat: [WORD]. Confirm: [WORD]. Finalize: [WORD]. Close: [WORD]." where `[WORD]` is replaced with the name of an object. For vision, we filter real images from the MS COCO dataset [16] that contain repeated natural object occurrences in diverse scenes.

In both modalities, we apply CountScope to test whether items encode positional count information. We treat item positions as ground truth (GT) and measure the probability assigned to the GT number versus non-GT numbers.



(a) Free-form text



(b) Natural image

Figure 10. Per-item latent count for natural data, decoded by CountScope. Each row shows the probability of decoding the count at different sequence positions for separators in (a) unstructured text and (b) real-world images from MS COCO dataset [16].

For text, the average probabilities for GT and non-GT are 0.81 and 0.02, respectively. For vision, the corresponding probabilities are 0.38 and 0.15, consistent with the standard setting. Figure 10 shows the corresponding per-item latent count decoding results for both modalities. These results indicate that the internal counter mechanism generalizes to free-form text and real-world images.

We next present qualitative examples of decoded counting for real-world images in Figure 11. We observe a consistent pattern in which the model processes and counts objects in a left-to-right and top-to-bottom order. This behavior aligns with the positional encoding of visual tokens and suggests that the internal counter follows a structured spatial traversal of the image.

F. Layer-Wise Representational Analysis

This section provides layer-wise visualizations of representational structure for both LLMs and LVLMs. Figure 12 shows PCA projections of input tokens, and Figure 13 presents PCA of generated responses. Figure 14 reports the corresponding trajectories for the LVLM. Figure 15 shows cosine similarity patterns across layers for element and separator tokens.

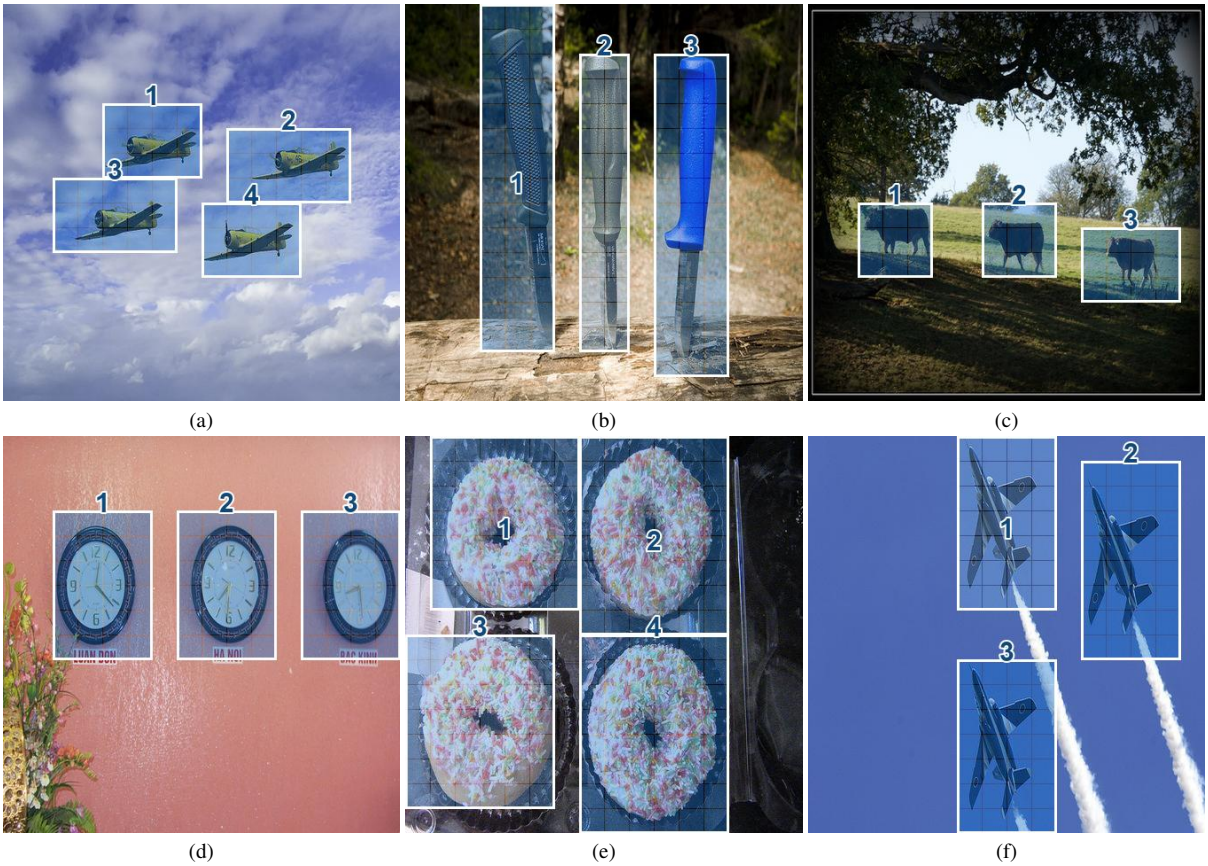
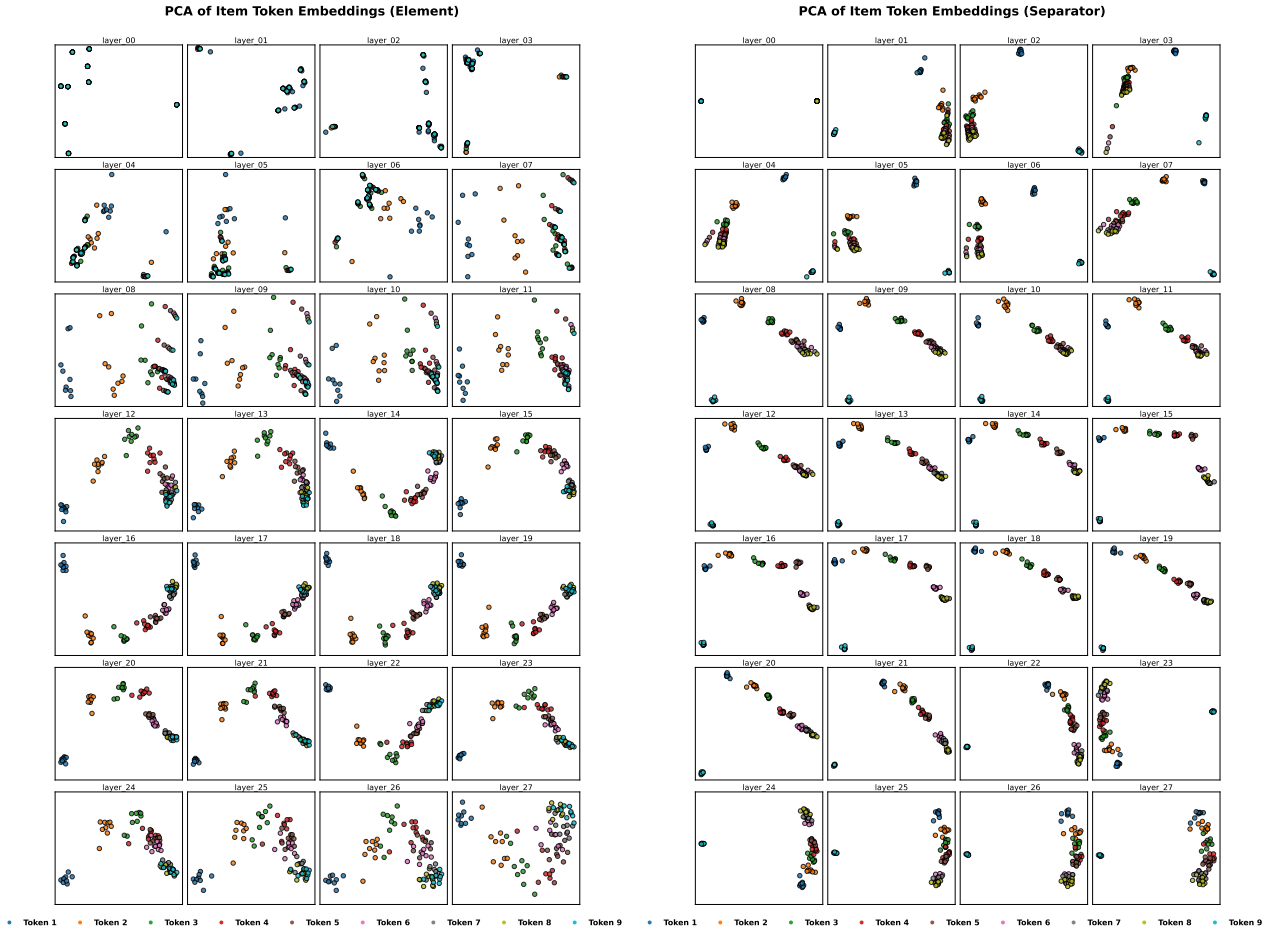


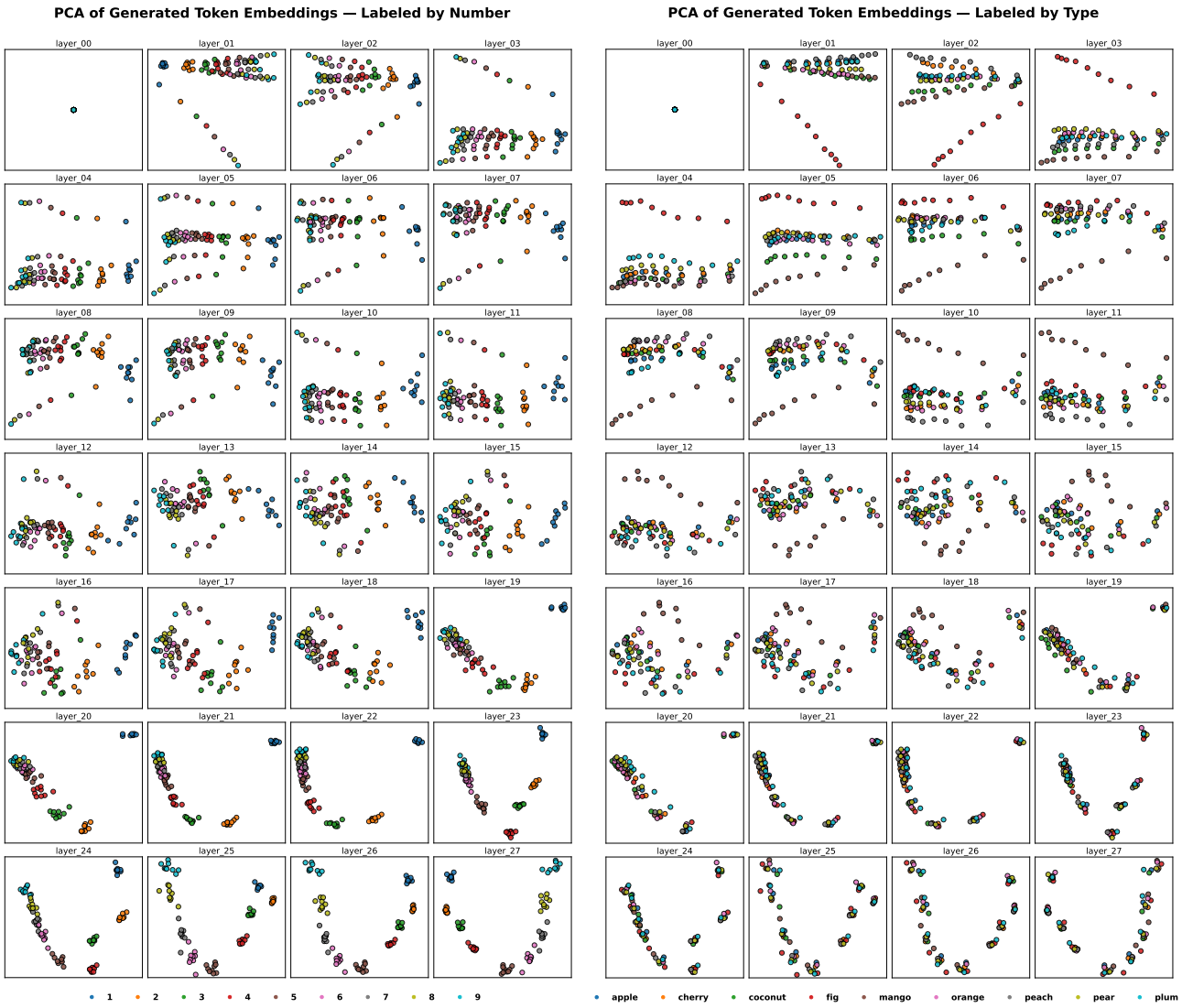
Figure 11. Qualitative results on natural images. Each image contains multiple objects. For each object, its region is patched into CountScope, and the predicted latent count is shown. The overlay heatmap indicates the confidence of the prediction.



(a) PCA of element-token embeddings.

(b) PCA of separator-token embeddings.

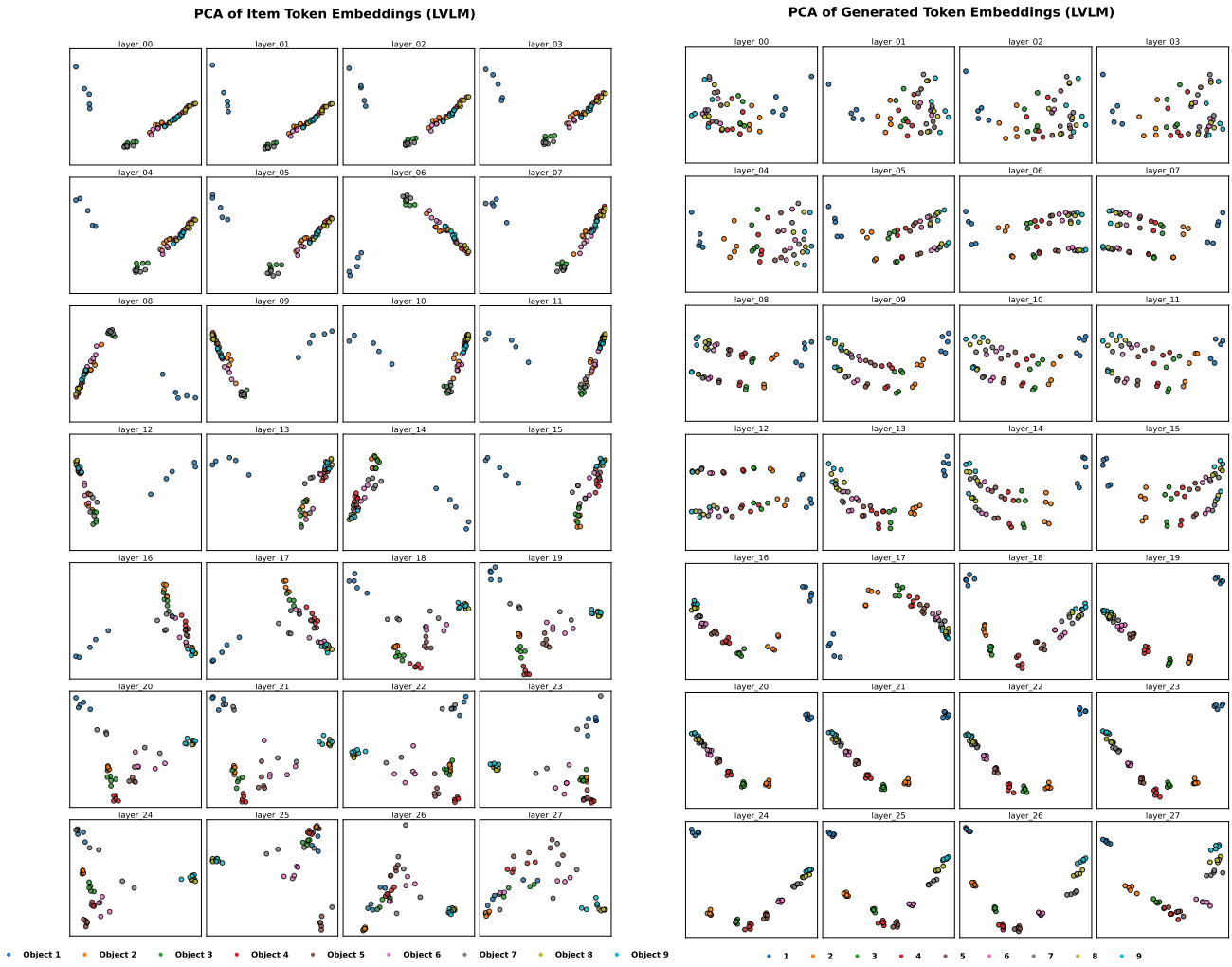
Figure 12. **Layer-wise PCA of Qwen2.5 input-token representations.** PCA trajectories across layers for (a) element tokens and (b) separator tokens in the monotypic, question-first setting.



(a) PCA of generated responses, colored by predicted number.

(b) PCA of generated responses, colored by item type.

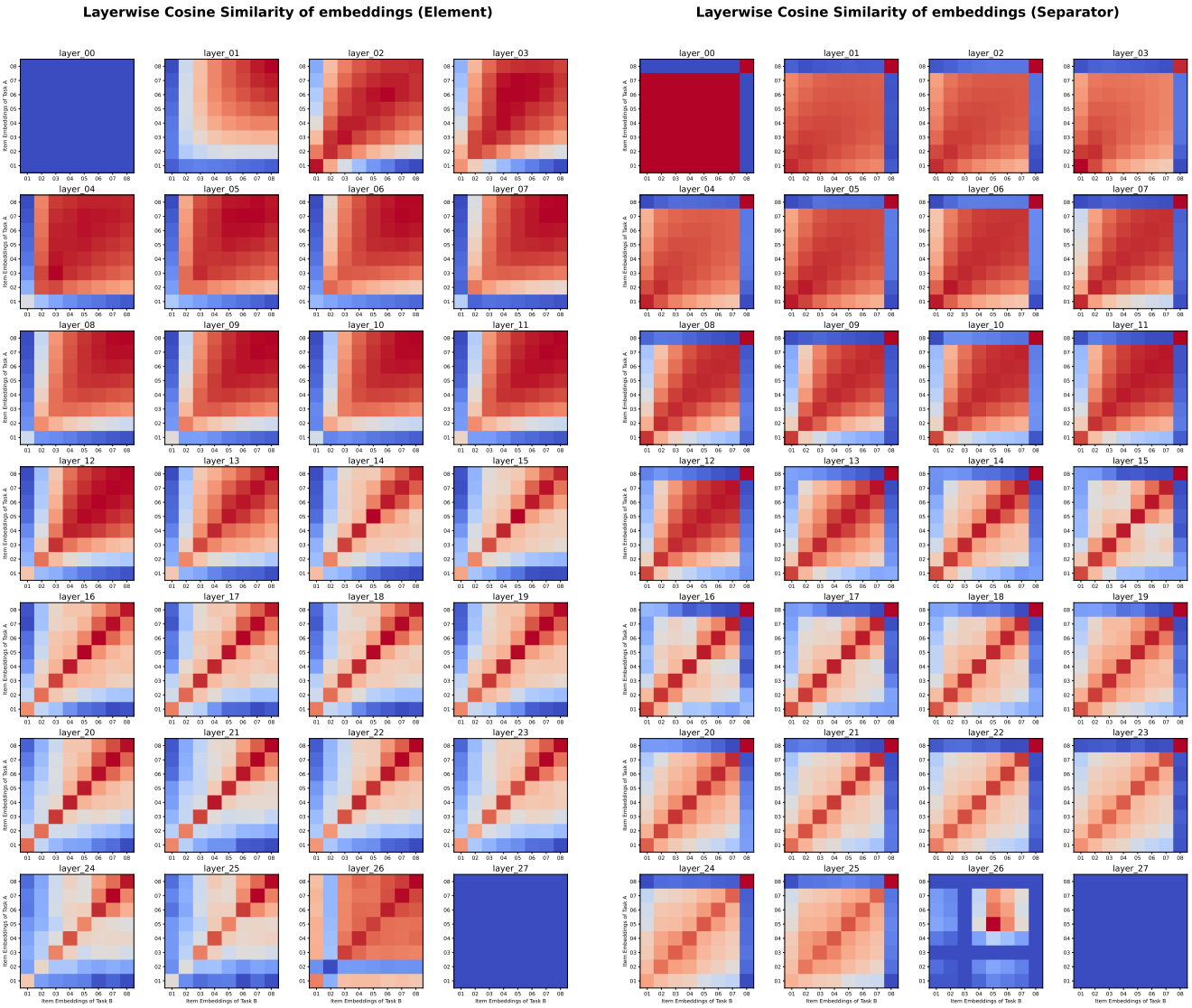
Figure 13. **Layer-wise PCA of Qwen2.5 output representations.** PCA embeddings across layers for generated numerical responses, colored by (a) predicted count and (b) item type, in the monotypic, question-first setting.



(a) PCA of input visual tokens.

(b) PCA of generated response.

Figure 14. **Layer-wise PCA of Qwen2.5VL embeddings.** PCA trajectories across layers for (a) input item embeddings and (b) generated output embeddings in the monotypic setting.



(a) Cosine similarity for element tokens.

(b) Cosine similarity for separator tokens.

Figure 15. **Layer-wise cosine similarity of Qwen2.5 representations.** Cosine similarity matrices across layers for (a) element tokens and (b) separator tokens in the monotypic, question-first setting. Cosine similarities are computed across different tasks with different item types and then averaged over the dataset.