Structure Tax: How Structured Output affects LLMs Performance

Vineet Kumar **Kanishka*** **Bhuvanesh Mandora**
PayPal Artificial Intelligence
PayPal, Bengaluru India
vkumar32@paypal.com, kanishka21dhar@gmail.com, bmandora@paypal.com

Abstract

Deploying large language models in production often requires constraining outputs to structured formats such as JSON or XML, and prior work treats the resulting accuracy loss as an inherent 'structure tax'. We re-examine this claim by evaluating a battery of models, datasets and schemas, measuring task accuracy, confidence calibration, and hidden-state geometry. The tax turns out to depend on schema design rather than on structure per se: reasoning-first field ordering matches or exceeds free-form accuracy, while answer-first ordering causes steep drops, particularly in smaller models. Format sensitivity scales inversely with a task's own structural constraints, and schemas that preserve reasoning order also improve calibration with CKA showing greater separability between correct and incorrect representations in middle transformer layers. Our findings indicate that properly designed structured formats can match or exceed free-form performance, reframing the critical question from 'whether to structure' to 'how to structure' for optimal reasoning preservation.

1 Introduction

Large language models are now deployed across diverse domains, including mathematical reasoning, knowledge retrieval (Rathore et al., 2026), and code generation (Kumar et al., 2025). In practice, their utility depends not only on producing correct answers, but also on how outputs are structured and how reliably models communicate uncertainty. As a result, structured output formats such as XML tags, JSON schemas etc. have become common prompting strategies for improving reliability and downstream usability. However, their impact on model behavior remains underexplored. Do structured formats merely alter surface form, or do they change how models internally process and represent problems? And if structure affects reasoning, does it also influence confidence calibration?

LLMs are known to be sensitive to a wide range of prompt features separator tokens, casing, template choice, few-shot ordering and this brittleness has been documented across model scales and instruction-tuning regimes (Sclar et al., 2024; He et al., 2024). Within this broader landscape, output format constraints are distinctive on two counts. First, they are the part of a prompt that practitioners actually ship to production: free-form responses are rarely viable in deployed systems, so the output schema is not an optional choice but a structural commitment with downstream consequences. Second, unlike surface-token brittleness whose magnitude is empirical and whose direction is not predictable a priori the effect of output structure on reasoning admits a causal hypothesis grounded in the architecture itself. We therefore focus on field ordering as a mechanistically tractable sub-axis of format sensitivity, and ask whether the apparent cost of structured outputs reflects a property of structure per se or of a specific design choice within it.

In this work, we investigate **Structure Tax**: the systematic effect of output format constraints on task accuracy, calibration, and internal representations. Across five models and four benchmarks, we show that the apparent cost is not inherent to structure but contingent on a single design choice – field ordering with second-order consequences for calibration and hidden-state geometry.

A key design dimension we examine is *field ordering*: whether the reasoning chain (R) is placed before or after the final answer (A) within a structured schema. This distinction matters because autoregressive generation is strictly causal; each token conditions on all preceding tokens, but not on those yet to be produced. A *reasoning-first* schema (R,A) forces the model to produce its reasoning trace before committing to an answer, allowing the

*Work done while at PayPal.

Proceedings of the 15th International Joint Conference on Natural Language Processing and the 5th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 4620–4644
November 6-10, 2026 ©2026 Association for Computational Linguistics

answer token to be informed by a fully elaborated chain of thought. An *answer-first* schema (A,R) inverts this dependency: the model commits to a specific answer token before any reasoning is generated, after which the reasoning field can only rationalize a decision already made. Figure 1 illustrates both orderings alongside a free-form baseline, making the structural difference concrete.

If field ordering matters because of autoregressive causality, then (R,A) schemas should preserve or improve reasoning quality while (A,R) schemas should degrade it, with the magnitude of degradation scaling with how much a task depends on multi-step inference. We confirm this prediction empirically. (R,A) schemas match or exceed free-form accuracy across the large majority of configurations, while (A,R) causes severe degradation on multi-step tasks, with the magnitude scaling inversely with a task's inherent structural organisation: low-structure tasks (knowledge QA, logical reasoning) benefit most; high-structure tasks (text-to-SQL) are largely format-invariant. This inverse-structure gradient holds at three levels of analysis — task accuracy, confidence calibration, and hidden-state geometry via CKA and linear probes and structured formatting simultaneously reduces expected calibration error, establishing field ordering as both an accuracy and a calibration lever.

2 Related Work

Tam et al. (2024) report that JSON and XML output constraints can degrade reasoning accuracy, motivating the notion of a behavioural "structure tax"; Long et al. (2025) broaden this to fifteen formats across four categories and propose mitigation through format-aware prompting and fine-tuning. A parallel body of work documents input-side sensitivity: Sclar et al. (2024) show that semantically irrelevant surface features like separators, casing, spacing can swing accuracy with brittleness persisting across model scales and instruction tuning, and He et al. (2024) report a 40-point spread on GPT-3.5 from template variation alone. These studies treat each format as atomic asking *which* format to use and lack a causal mechanism for the observed effects. We isolate a within-format axis, the relative position of reasoning and answer fields, where the direction of the effect is predictable from autoregressive generation: a token emitted at position t can only condition on positions strictly less than t, so an answer token committed before its rationale cannot be revised by it.

The benefits of placing reasoning before the answer are established for chain-of-thought prompting (Wei et al., 2022). Most closely related, Fu et al. (2026) compare answer-direct and reasoning-then-answer prompts on multiple-choice tasks and find that reasoning-first increases verbalised confidence regardless of correctness. Our work differs in three ways: we span a structural-density continuum rather than MCQ alone, we separate the format-family axis (JSON vs XML) from the field-ordering axis, and we trace the mechanism into hidden-state geometry. In a complementary fine-tuning setting, Chen et al. (2025) argue that post-thinking distillation (answer before rationale) is more robust to rationale errors for small models whether our prompted-generation finding persists under such training is an open question. Our results also establishes the connection to chain-of-thought faithfulness (Lanham et al., 2023): plausible-sounding reasoning can still be post-hoc if the answer token precedes it in the generation order.

Calibration is commonly assessed using Expected Calibration Error (Minderer et al., 2021) and reliability diagrams (Guo et al., 2017). Hidden states are known to encode correctness signals beyond surface outputs (Servedio et al., 2025; Orgad et al., 2025; Zhang et al., 2025), motivating combined behavioural-representational analysis. We compare layer representations using Centred Kernel Alignment (Kornblith et al., 2019), mindful of documented interpretive caveats (Davari et al., 2023; Cui et al., 2022), and verify CKA directions against linear probes and accuracy. To our knowledge, no prior work jointly examines how output format constraints affect task performance, calibration, and correctness-related representational geometry; our study addresses this gap.

3 Experimental Design

We investigate three complementary aspects of format constraints task accuracy, calibration, and internal representations detailed in §3.2.

3.1 Dataset

We select four datasets that sample a continuum of intrinsic task structure the degree to which the task itself imposes compositional or sequential constraints on a valid answer, independent of any output format we impose. This continuum framing is central to our analysis: if format scaffolding

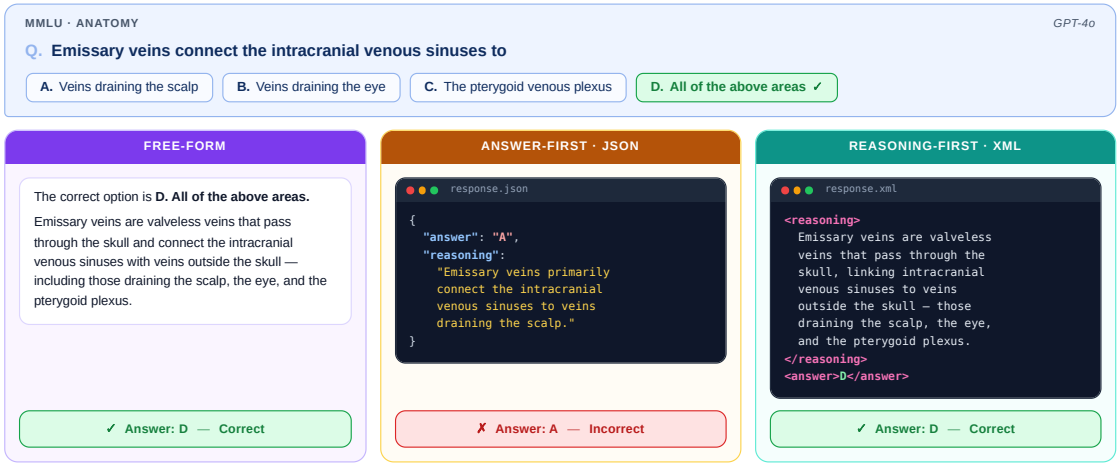


Figure 1: Three response formats elicited from GPT-4o on the same MMLU Anatomy question. **(Left)** *Free-form*: natural-language reasoning arrives at the correct answer **D**. **(Middle)** *Answer-first JSON*: the model commits to `"answer": "A"` before generating a post-hoc rationale that cites only one of three valid targets, yielding an incorrect answer. **(Right)** *Reasoning-first XML*: rationale unrolled inside `<reasoning>` precedes the commitment, recovering the correct `<answer>D</answer>`.

interacts with intrinsic structure, the magnitude of any format effect should scale with where a task sits on the continuum. **Spider** (Yu et al., 2018) anchors the high-structure end, a large-scale text-to-SQL dataset where SQL grammar imposes rigid compositional constraints on valid outputs. **GSM8K** (Cobbe et al., 2021) occupies a middle position: grade-school arithmetic word problems require multi-step reasoning that provides sequential scaffolding but no syntactic enforcement. **MMLU** (Hendrycks et al., 2021), a 57-subject multitask benchmark, and **LogicBench** (Parmar et al., 2024), designed to evaluate logical reasoning across reasoning types and complexity levels, sit at the low-structure end: the multiple-choice format is externally imposed and the reasoning path is otherwise unconstrained. JSON and XML are the two formats we evaluate; together they cover the principal axis a practitioner faces key-ordered serialisation (JSON) versus tag-bounded markup (XML) and span the formats most commonly shipped in production pipelines.

3.2 Experiment protocol

Our experiment is three-pronged. We first measure task **accuracy under structured output** formats (JSON and XML, each in answer-first and reasoning-first orderings) against a free-form baseline. We then assess **calibration** by analyzing both verbalized confidence scores and token-level probability signals to determine whether format constraints shift the alignment between model confidence and empirical accuracy. Finally, we probe layer-wise hidden states using Centered Kernel Alignment (CKA) to characterize how format constraints alter the **internal geometry** of correct versus incorrect representations across model layers.

Why include the answer-first (A,R) ordering? We include (A,R) for three reasons. First, it mirrors a common production pattern where developers place the answer before the rationale because it reads more naturally to human reviewers (e.g., *"rate this text and provide a rationale"*). Second, while the direction of the effect is intuitive, its *magnitude* and *model dependence* reqquires empirical characterization. Third, (A,R) isolates the effect of field ordering from the broader format-family distinction (JSON vs. XML), allowing us to attribute performance differences specifically to *when* the answer is committed rather than to surface syntax.

3.3 Models

We evaluate five LLMs spanning both proprietary and open-weight families. The frontier proprietary models: **GPT-4o** (OpenAI, 2024), **Gemini-Flash-2.0** (Google, 2025), and **Claude-Sonnet-4.5** (Anthropic, 2025) – span the three major frontier providers, reducing single-provider bias in our conclusions. The open-weight models – **Llama-3.1-8B** (Grattafiori et al., 2024) and **Gemma3-4B** (Ka-

math et al., 2025) span two scales (4B and 8B parameters) and two architecture families, which lets us check whether the representational patterns we observe (in Section §6) generalise across architectures rather than depending on a single model design. The open-weight subset also enables reproducible, large-scale ablations under practical compute constraints.

3.4 Evaluation Metrics

Let $\hat{y}_i$ denote the model prediction and p_i the confidence for input x_i with ground truth y_i.

- **Accuracy** measures exact-match fraction; for Spider (Yu et al., 2018) we use **Execution Accuracy (EX)**, which checks whether executing the predicted SQL yields the same result as the reference query.
- **ECE** quantifies calibration by comparing confidence to empirical accuracy across M bins:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{N} \left|\text{acc}(B_m) - \text{conf}(B_m)\right|$$

- **Confidence–Correctness Correlation** (ρ) is the Spearman rank correlation between p_i and $\mathbf{1}[\hat{y}_i = y_i]$, capturing whether higher confidence predicts correctness.
- Brier Score and Entropy, used in Appendix C.2 and C.3, are defined there.

4 Structure Tax

We ask whether imposing a structured output schema incurs a universal accuracy penalty a *structure tax* or whether its effect depends on how the schema is designed.

Setup. Each model receives a task-specific system prompt specifying the required output schema (JSON or XML; R,A or A,R); all other prompt content is held constant across conditions. Responses are collected at `temperature = 0` via each provider's API. Evaluation uses exact-match accuracy for GSM8K, MMLU, and LogicBench, and execution accuracy (EX) for Spider. Prompt templates are reproduced in Appendix A.1.

Results (Table 1). Reasoning-first formats dominate. XML (R,A) or JSON (R,A) wins the best-format cell in 13 of 20 model-dataset pairs and never collapses on tasks where A,R wins. The A,R penalty is strongest on GSM8K XML (A,R) falls to 10.98% for both **Llama-3.1-8B** and **Gemma3-4B**, against 83-87% under (R,A); confirming that committing the answer token before the reasoning chain breaks multi-step arithmetic entirely. On Spider (the high-structure task), freeform remains competitive for three of five models, consistent with SQL generation providing its own intrinsic structure.

Analysis. The severity of the ordering effect scales with task reasoning depth: the R,A advantage is largest on GSM8K, moderate on MMLU and LogicBench, and negligible on Spider. Within (R,A) XML outperforms JSON (margin up to 12.2 pp on LogicBench/Claude-Sonnet-4.5), suggesting that explicit tag boundaries scaffold the reasoning-answer transition more reliably than JSON key ordering alone. Smaller models are more fragile: Llama-3.1-8B and Gemma3-4B show the largest collapses under A,R and higher run-to-run variance, whereas frontier models are sensitive to ordering but do not catastrophically fail. Crucially, R,A formats simultaneously improve accuracy and reduce variance.

Robustness to surface prompt variation. A natural concern is whether the (R,A) advantage is an artefact of the specific prompts used. We test this by perturbing instructional phrasing, schema casing, and whitespace across three variants per condition on GSM8K with Llama-3.1-8B and GPT-4o (full setup in Appendix F). The (R,A) > (A,R) gap is preserved in every (model, format-family, variant) cell: the smallest gap across all 24 combinations is +37 points (GPT-4o JSON), while within-cell spread under perturbation is at most 11.5 points. Field ordering's effect on accuracy is qualitatively invariant to surface prompt changes, distinguishing it from the format-template brittleness documented by (Sclar et al., 2024) and (He et al., 2024).

> **KEY TAKEAWAY**
>
> Field ordering, not structure itself, governs the structure tax: reasoning-first schemas (R,A) consistently match or nearly match the best-performing formats, while answer-first schemas (A,R) may cause severe and in some settings catastrophic degradation on reasoning-heavy tasks.

Dataset	Model	Free Form	Answer-first (A,R)		Reasoning-first (R,A)	
			JSON	XML	JSON	XML
GSM8K	GPT-4o	88.78 ±1.02	55.34 ±1.58	56.86 ±2.71	95.60 ±1.01	**95.60** ±0.17
	Gemini-Flash-2.0	79.38 ±0.28	63.08 ±0.53	54.90 ±3.64	**95.53** ±0.59	95.45 ±0.60
	Claude-Sonnet-4.5	86.50 ±1.13	96.13 ±0.88	62.35 ±2.29	87.79 ±0.98	**97.65** ±0.39
	Llama-3.1-8B	79.98 ±1.10	31.01 ±1.58	10.98 ±2.66	80.21 ±0.45	**83.53** ±2.58
	Gemma3-4B	78.82 ±1.46	27.45 ±4.70	10.98 ±2.13	83.53 ±1.96	**87.06** ±2.34
LogicBench	GPT-4o	79.00 ±0.96	83.80 ±1.29	83.00 ±1.29	83.20 ±0.73	**84.20** ±1.93
	Gemini-Flash-2.0	72.80 ±2.11	76.80 ±1.64	84.77 ±2.34	82.40 ±1.75	**85.40** ±1.16
	Claude-Sonnet-4.5	87.60 ±2.17	56.60 ±2.70	**88.40** ±0.55	75.00 ±0.77	87.20 ±1.35
	Llama-3.1-8B	57.20 ±0.79	**76.00** ±1.05	70.34 ±1.36	74.60 ±2.05	73.00 ±2.50
	Gemma3-4B	61.20 ±0.71	**69.40** ±1.33	67.40 ±1.87	66.00 ±1.61	66.20 ±2.22
MMLU	GPT-4o	79.95 ±0.84	82.50 ±0.83	81.58 ±1.36	**87.46** ±0.70	86.94 ±0.78
	Gemini-Flash-2.0	56.24 ±0.78	84.52 ±0.61	83.96 ±0.94	84.26 ±1.13	**86.14** ±0.76
	Claude-Sonnet-4.5	85.37 ±0.98	90.84 ±0.65	90.66 ±0.63	90.92 ±0.32	**92.03** ±0.64
	Llama-3.1-8B	62.83 ±0.80	64.14 ±1.71	55.98 ±0.75	**71.26** ±0.70	68.06 ±0.86
	Gemma3-4B	59.76 ±1.08	60.29 ±0.83	58.72 ±1.08	**63.10** ±1.03	62.12 ±1.54
Spider	GPT-4o	69.15 ±1.78	69.05 ±1.32	71.76 ±1.57	71.47 ±0.93	**73.02** ±1.40
	Gemini-Flash-2.0	**79.40** ±0.84	77.37 ±1.33	78.20 ±1.09	76.69 ±1.34	77.18 ±1.33
	Claude-Sonnet-4.5	77.47 ±1.60	77.47 ±1.23	**78.05** ±1.20	77.37 ±1.36	77.66 ±1.52
	Llama-3.1-8B	**64.99** ±1.85	58.03 ±0.86	64.31 ±1.15	54.84 ±2.71	62.19 ±1.60
	Gemma3-4B	**70.21** ±1.35	66.15 ±1.48	68.38 ±0.89	67.50 ±0.97	63.35 ±0.57

Table 1: Impact of output structure constraints on accuracy across benchmark tasks. We compare *Answer-first* (answer before reasoning) and *Reasoning-first* (reasoning before answer) orderings under JSON and XML formats, against an unconstrained Free Form baseline. Values are mean accuracy over 5 independent runs (standard deviation in gray). Best format per model–dataset pair is **bolded** and highlighted.

5 Calibration and Confidence Estimation

Accuracy gains from structured formatting are only meaningful if models also communicate reliable uncertainty. We ask whether output format affects calibration i.e. alignment between stated confidence and empirical accuracy. Confidence is elicited as a verbal 1–10 integer score and calibration is measured via ECE; elicitation strategy and logit-verbal alignment are explained in Appendix C. Because the accuracy experiments in Section 4 identify XML with reasoning-first ordering, (R,A), as the recommended structured format, we evaluate the calibration of this configuration against free-form generation. The results show that the recommended XML (R,A) format also reduces expected calibration error relative to free-form output.

5.1 Output Format as a Calibration Mechanism

Figure 2 reports $\Delta\text{ECE} = \text{ECE}_{\text{freeform}} - \text{ECE}_{\text{XML}}$; positive values indicate freeform is *worse* calibrated. The pattern closely mirrors the accuracy results: format sensitivity is highest on tasks lacking inherent structure and negligible where intrinsic scaffolding already exists. LogicBench shows consistent gains under XML across all models; MMLU shows gains for four of five models, the exception being GPT-4o where freeform is marginally better calibrated. GSM8K is effectively format-invariant — arithmetic provides its own sequential discipline that format cannot augment. Spider (the high-structure task) is the dataset where freeform is better calibrated for some model, again consistent with SQL imposing its own structural constraints. These results establish XML formatting not merely as an accuracy intervention but as a *calibration mechanism*: it regularises confidence expression alongside reasoning quality.

5.2 Calibration Profiles Under Structured Output

Table 2 reports ECE, overconfidence rate (OvConf), and Spearman ρ between confidence and correctness under XML prompting. The task-level ordering is stable across models: GSM8K is best

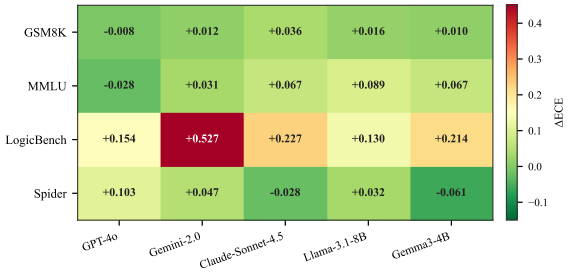


Figure 2: ECE increase under freeform relative to XML ($\Delta\text{ECE} = \text{ECE}_{\text{freeform}} - \text{ECE}_{\text{XML}}$). Positive values (red) indicate freeform is worse calibrated; negative (green) indicates the reverse. LogicBench shows the largest format sensitivity; GSM8K is format-invariant.

calibrated, Spider is worst, with LogicBench and MMLU in between. This ordering reflects how much intrinsic task structure guides confidence. Arithmetic answers are either right or wrong in a verifiable chain, while SQL execution correctness is opaque to the model at generation time Across all settings, the confidence-accuracy correlation ρ remains low, indicating that while XML improves aggregate calibration, per-example discrimination is limited. Appendix Figures 4 and 5 visualize the confidence–accuracy gap and per-bin calibration across all model–dataset conditions.

Dataset	Model	ECE ↓	OvConf ↓	Corr (ρ) ↑
GSM8K	GPT-4o	0.050 ±0.005	0.051 ±0.005	0.169 ±0.028
	Gemini-Flash-2.0	0.042 ±0.002	0.043 ±0.002	0.134 ±0.084
	Claude-Sonnet-4.5	**0.020** ±0.004	**0.022** ±0.004	0.264 ±0.063
	Llama-3.1-8B	0.136 ±0.008	0.138 ±0.009	0.189 ±0.053
	Gemma3-4B	0.109 ±0.028	0.111 ±0.029	**0.350** ±0.146
LogicBench	GPT-4o	0.097 ±0.018	0.197 ±0.018	0.253 ±0.078
	Gemini-Flash-2.0	0.097 ±0.017	0.131 ±0.022	**0.291** ±0.049
	Claude-Sonnet-4.5	**0.060** ±0.009	**0.108** ±0.012	0.251 ±0.037
	Llama-3.1-8B	0.103 ±0.016	0.244 ±0.020	0.222 ±0.064
	Gemma3-4B	0.243 ±0.015	0.337 ±0.015	0.256 ±0.042
MMLU	GPT-4o	0.073 ±0.005	0.132 ±0.006	0.289 ±0.023
	Gemini-Flash-2.0	0.104 ±0.010	0.131 ±0.011	0.282 ±0.040
	Claude-Sonnet-4.5	**0.041** ±0.007	**0.082** ±0.004	**0.396** ±0.026
	Llama-3.1-8B	0.208 ±0.014	0.321 ±0.012	0.173 ±0.021
	Gemma3-4B	0.292 ±0.016	0.365 ±0.014	0.214 ±0.026
Spider	GPT-4o	0.201 ±0.018	0.258 ±0.017	0.247 ±0.025
	Gemini-Flash-2.0	0.210 ±0.011	**0.211** ±0.010	0.115 ±0.032
	Claude-Sonnet-4.5	**0.185** ±0.016	0.235 ±0.015	0.299 ±0.038
	Llama-3.1-8B	0.273 ±0.009	0.351 ±0.009	0.171 ±0.018
	Gemma3-4B	0.283 ±0.006	0.326 ±0.008	**0.331** ±0.050

Table 2: Calibration under XML-structured output. ECE and OvConf are lower-is-better; Corr (ρ) is higher-is-better. Values are mean over 5 runs (std. in gray). Best model per metric–dataset pair is **bolded**.

Elicitation strategy and logit-verbal alignment. Absolute elicitation generally outperforms distributional prompting and is used as the default (Appendix C.2). Verbal scores correlate moderately with logits only in stronger models; smaller models show little alignment, with large KL divergence between verbal and token-level uncertainty signals (Appendix C.3). Thus, for smaller models, verbalized confidence is better viewed as a task-conditioned heuristic than a reflection of internal uncertainty.

> **KEY TAKEAWAY**
>
> Output format acts as a calibration mechanism, not just an accuracy intervention: XML (R,A) reduces ECE alongside improving reasoning quality, with gains following the same inverse-structure gradient observed in Section 4.

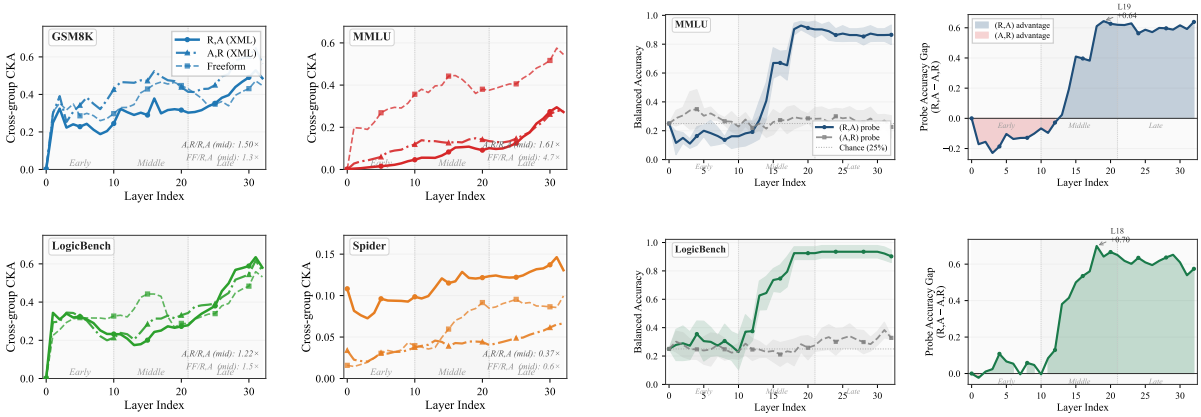
6 Internal Representations and Mechanism

The accuracy and calibration results establish *what* field ordering does; this section asks *why*. We open the model's intermediate computations with two complementary tools. **CKA** (Kornblith et al., 2019) acts as a representational separability thermometer: given hidden states for correct and incorrect predictions at each layer, lower cross-group CKA means the model has built more geometrically distinct internal representations for the two outcomes. **Linear probes** act as an information readout: a linear classifier trained on hidden states at layer ℓ tells us whether the answer identity is already decodable at that depth, before it is generated. A third lens reasoning length connects the surface output to the representational evidence. All three converge on the same account of how field ordering shapes computation.

6.1 Representational Separability: CKA

We apply CKA to **Llama-3.1-8B** & **Gemma3-4B** under XML (R,A), XML (A,R), and freeform across all four datasets; extraction details are in Appendix D.1.

Figure 3 (left) shows that format sensitivity tracks the same inverse-structure gradient seen in accuracy and calibration. On **MMLU** the task with the least inherent structure freeform cross-group CKA rises steadily from early layers and stays elevated, whereas XML (R,A) keeps it below 0.15 through the middle layers: the model is building representations that separate right from wrong only



(a) Cross-group CKA per layer

(b) Linear probe balanced accuracy per layer

Figure 3: Internal representations for **Llama-3.1-8B** on MMLU and LogicBench. **Left**: Cross-group CKA under XML (R,A) (solid), XML (A,R) (dash-dot), and Freeform (dashed) across four benchmarks. Higher CKA = weaker separation between correct and incorrect representations; R,A yields the lowest mid-layer CKA on reasoning tasks while Spider reverses the pattern. **Right**: Linear probe balanced accuracy decoding the predicted answer label from hidden states at each layer. Solid = (R,A); dashed = (A,R); dotted = chance; gap panel shows the (R,A)−(A,R) advantage with peak layer annotated. **Gemma3-4B** replicates both patterns at reduced magnitude (Appendix D.4).

when the output schema forces deliberate reasoning first. **LogicBench** shows a moderate version of the same pattern, converging in late layers as both conditions reach the same reasoning conclusion. **GSM8K** exhibits near-parity across formats arithmetic provides its own sequential scaffold while **Spider** reverses the gradient entirely, the only task where freeform yields lower cross-group CKA than XML, consistent with SQL imposing its own compositional structure that XML tags disrupt. Critically, the direction of the CKA effect correctly predicts the direction of the accuracy effect in all four tasks: Spider is the only task where XML reduces accuracy and the only task where the CKA ratio falls below parity. **Gemma3-4B** replicates this pattern at reduced magnitude (Appendix D.4), confirming findings are not model-specific.

An instructive dissociation appears on MMLU: despite the largest representational suppression under freeform prompting, Llama gains only modest accuracy compared to LogicBench. This highlights the limit of format scaffolding i.e., accuracy remains constrained by the model's underlying knowledge. XML can enforce deliberate ordering, not supply missing facts. Within-group CKA heatmaps further show that XML induces clearer stage-wise block structure in correct representations (Appendix, Figures 6–7).

6.2 Answer Encoding: Linear Probes

While CKA measures whether representations are separable, probes ask a more targeted question: does the model already *know the answer* during the reasoning phase, before the answer token is generated? We train a linear classifier on pre-answer hidden states at each layer to decode the predicted label and compare accuracy under (R,A) vs (A,R) on MMLU and LogicBench (GSM8K and Spider excluded; see Appendix D.3).

The answer is unambiguous (Figure 3, right). Under (R,A), answer identity becomes linearly decodable from middle layers onward, peaking at balanced accuracy of 0.64 (MMLU, L19) and 0.70 (LogicBench, L18) well above chance and sustained across the final third of the network. Under (A,R), the same probe stagnates near chance through early and middle layers: the model has committed to an answer token but has not built a representation of *which* answer it chose during the reasoning phase the answer is present in the output stream. With the answer field first, the model must emit the answer letter without preceding explicit reasoning. The pre-answer-letter state may therefore reflect less task-relevant computation, increasing the risk that commitment precedes deliberate inference. **Gemma3-4B** replicates the directional pattern at reduced magnitude (Appendix D.4), again confirming this is a general property of field ordering, not an artefact of one model family.

6.3 Reasoning Length and the Two Failure Modes

The probe and CKA results describe what happens inside the model; reasoning length reveals the surface signature. We measure word count across 80 cells (5 models $\times$ 4 datasets $\times$ 4 format-orderings) as a proxy for reasoning effort; extraction details and full results appear in Appendix E.

The data reveal two qualitatively distinct failure modes of the (A,R) ordering. The first is **rationalization**: on MCQ tasks, committing the answer token causes the reasoning field to collapse JSON (A,R) produces just 33 words on MMLU against 121 under XML (R,A), a fourfold compression into a brief post-hoc justification rather than deliberate inference. The second is **premature commitment**: on GSM8K, reasoning length is flat across all four orderings (108–127 words), yet accuracy collapses from 0.92 under R,A to 0.55 under JSON (A,R) the model writes as much, but arithmetic that should determine the answer is instead forced to justify one already committed.

Length alone does not predict accuracy ($r = +0.04$ overall); the ordering dimension carries the signal. JSON (A,R)'s correlation of $r = -0.73$ ($p < 0.001$) the only significant result across all four format-orderings (Appendix Table 9) arises because the cells where JSON (A,R) does *not* compress (GSM8K, Spider) are precisely those where accuracy is worst: verbose post-commitment reasoning is worse than compressed post-commitment reasoning, because ordering determines whether effort precedes or follows the decision.

> **KEY TAKEAWAY**
>
> CKA, linear probes, and reasoning-length analysis all support the same conclusion: reasoning-first ordering lets the model form clear answer-related representations before committing to a final response, preserving deliberate reasoning. In contrast, answer-first ordering leads to either compressed post-hoc rationalization in MCQ tasks or incorrect computation despite long reasoning traces in arithmetic tasks. XML (R,A) avoids both by allowing the answer to be generated only after the reasoning process is fully developed.

7 Discussion and Conclusion

Two failure modes: practical diagnostics. The two failure modes share a single underlying cause autoregressive generation forces the answer token to be sampled either before or after the reasoning trace but they call for different diagnostics in deployment. Rationalization on MCQ tasks is detectable from output length alone: a steep drop in reasoning-field token count under (A,R) relative to (R,A) is a reliable signal that the model is producing post-hoc justification rather than deliberate inference. Premature commitment on arithmetic tasks is harder to detect, because reasoning length is preserved (108–127 words across all four orderings on GSM8K); here the diagnostic must check whether the reasoning is consistent with the stated answer, not merely whether reasoning is present. Practitioners auditing structured pipelines for reasoning failures should therefore monitor both the volume of the reasoning field and its semantic alignment with the committed answer.

The "structure tax" attributed to structured output constraints is largely an artefact of answer-first field ordering rather than structure per se. Autoregressive generation is strictly causal, so an answer token emitted before its rationale cannot be revised by it; reasoning-first (R,A) schemas avoid this and consistently match or exceed free-form accuracy across five models and four datasets, while answer-first schemas produce either compressed post-hoc rationalization on MCQ tasks or premature commitment on arithmetic tasks. The format-sensitivity gradient is consistent across accuracy (Table 1), calibration (Figure 2), and hidden-state geometry (Figure 3), with gains largest on low-structure tasks and negligible on high-structure tasks.

Scope and applicability. The gains we report concentrate where the task lacks intrinsic structure and the model is mid-sized: low-structure benchmarks (MMLU, LogicBench) under smaller open-weight models show the largest absolute (R,A) advantages and the most dramatic (A,R) collapses. On high-structure tasks like text-to-SQL, where compositional constraints are imposed by the output language itself, field ordering matters less and the choice between (R,A) and free-form is close to a wash. Frontier models are sensitive to ordering but rarely catastrophic under (A,R), suggesting larger models partially compensate for the causal-ordering constraint. For practitioners, reasoning-first XML is therefore the safer default: it never

meaningfully hurts on the tasks where structure matters, and it preserves accuracy and calibration on the tasks where it does without sacrificing downstream parseability.

Limitations

A recurring pattern across our analyses is that format sensitivity scales inversely with a task's inherent structural organisation Spider at the high-structure end, MMLU and LogicBench at the low-structure end, GSM8K between them. Our four datasets sample this continuum at only four coarse points, which is sufficient to establish the inverse-scaling trend but insufficient to formally characterise a *sweet spot* — the level of intrinsic structural density at which format scaffolding is maximally beneficial. Formally locating such a sweet spot would require systematic variation of structural density within a single task family (e.g., varying constraint density in SQL generation, or step-count and carry-chain length in arithmetic), holding all other factors constant. We flag this as a concrete and important direction for future work.

Representational analysis via CKA and linear probes is restricted to two open-weight models (Llama-3.1-8B and Gemma3-4B), leaving open whether the observed middle-layer patterns generalise across model scales and architectures. We examine only two field orderings (answer-first and reasoning-first); real-world schemas involve finer-grained structure such as planning fields, intermediate steps, and verification stages, and how structural granularity beyond binary ordering affects reasoning remains an open question. All experiments rely on prompting rather than training-time, or constrained decoding so it is unclear whether fine-tuning on structured formats would produce the same representational geometry. Finally, extending the analysis cross-lingually may reveal whether format scaffolding compensates more strongly in settings where the model has weaker internalised task structure.

References

Anthropic. 2025. Claude sonnet 4.5. Accessed: 2026-03-16.

Xiaoshu Chen, Sihang Zhou, Ke Liang, and Xinwang Liu. 2025. Distilling reasoning ability from large language models with adaptive thinking. *Preprint*, arXiv:2404.09170.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Tianyu Cui, Yogesh Kumar, Pekka Marttinen, and Samuel Kaski. 2022. Deconfounded representation similarity for neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*.

MohammadReza Davari, Stefan Horoi, Amine Natik, Guillaume Lajoie, Guy Wolf, and Eugene Belilovsky. 2023. Reliability of CKA as a similarity measure in deep learning. In *The Eleventh International Conference on Learning Representations*.

Tairan Fu, Javier Conde, Gonzalo Martinez, Maria Grandury, and Pedro Reviriego. 2026. Multiple choice questions: Reasoning makes large language models (llms) more self-confident, specially when they are wrong. *IEEE Intelligent Systems*, page 1–10.

Google. 2025. Gemini flash 2.0 model documentation. Accessed: 2026-03-16.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Jia He, Mukund Rungta, David Koleczek, Arshdeep Sekhon, Franklin X Wang, and Sadid Hasan. 2024. Does prompt formatting have any impact on llm performance? *Preprint*, arXiv:2411.10541.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, and 196 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *Proceedings of*

the 36th International Conference on Machine Learning (ICML), volume 97 of *Proceedings of Machine Learning Research*, pages 3519–3529. PMLR.

Vineet Kumar, Ronald Tony, Darshita Rathore, Vipasha Rana, Bhuvanesh Mandora, . Kanishka, Chetna Bansal, and Anindya Moitra. 2025. Genicious: Contextual few-shot prompting for insights discovery. In *Proceedings of the 8th International Conference on Data Science and Management of Data (12th ACM IKDD CODS and 30th COMAD)*, CODS-COMAD '24, page 405–409, New York, NY, USA. Association for Computing Machinery.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, and 11 others. 2023. Measuring faithfulness in chain-of-thought reasoning. *Preprint*, arXiv:2307.13702.

Do Xuan Long, Ngoc-Hai Nguyen, Tiviatis Sim, Hieu Dao, Shafiq Joty, Kenji Kawaguchi, Nancy F. Chen, and Min-Yen Kan. 2025. LLMs are biased towards output formats! systematically evaluating and mitigating output format bias of LLMs. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 299–330, Albuquerque, New Mexico. Association for Computational Linguistics.

Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Ann Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lucic. 2021. Revisiting the calibration of modern neural networks. In *Advances in Neural Information Processing Systems*.

OpenAI. 2024. Hello gpt-4o. Accessed: 2026-03-16.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. 2025. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *The Thirteenth International Conference on Learning Representations*.

Mihir Parmar, Nisarg Patel, Neeraj Varshney, Mutsumi Nakamura, Man Luo, Santosh Mashetty, Arindam Mitra, and Chitta Baral. 2024. LogicBench: Towards systematic evaluation of logical reasoning ability of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13679–13707, Bangkok, Thailand. Association for Computational Linguistics.

Darshita Rathore, Vineet Kumar, Vaibhav Singal, Ankur Vivek Singh, and Anindya Moitra. 2026. Conversational query engine for mixed-modality heterogeneous enterprise data sources. *Preprint*, arXiv:2606.28370.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*.

Giovanni Servedio, Alessandro De Bellis, Dario Di Palma, Vito Walter Anelli, and Tommaso Di Noia. 2025. Are the hidden states hiding something? testing the limits of factuality-encoding capabilities in LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6089–6104, Vienna, Austria. Association for Computational Linguistics.

Zhi Rui Tam, Cheng-Kuang Wu, Yi-Lin Tsai, Chieh-Yen Lin, Hung-yi Lee, and Yun-Nung Chen. 2024. Let me speak freely? a study on the impact of format restrictions on large language model performance. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1218–1236, Miami, Florida, US. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-SQL task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3911–3921, Brussels, Belgium. Association for Computational Linguistics.

Anqi Zhang, Yulin Chen, Jane Pan, Chen Zhao, Aurojit Panda, Jinyang Li, and He He. 2025. Reasoning models know when they're right: Probing hidden states for self-verification. In *Second Conference on Language Modeling*.

Appendix Contents

A Complete Prompt Collection **11**
A.1 Structured Format Evaluation 11
A.2 Calibration Analysis Prompts 13
A.3 Schema violation statistics 13
A.4 Logit Confidence Analysis Prompts 13
B Dataset Summary Statistics **14**
C Calibration: Supplementary Details **14**
C.1 Implementation Details 14
C.2 Confidence Elicitation: Absolute vs. Relative 14
C.3 Verbal Confidence vs. Internal Probabilities 14
C.4 Calibration Visualisations 15
D CKA and Probe: Method Details **16**
D.1 CKA Extraction 16
D.2 Within-Group CKA Heatmaps: Layout and Reading Guide 17
D.3 Linear Probe Setup 23
D.4 Gemma3-4B: Representational Results 23
E Reasoning Length: Measurement and Results **24**
F Prompt-Brittleness Ablation **24**

A Appendix: Complete Prompt Collection

This appendix documents all prompts used in our three experimental conditions: Structured Format Evaluation (§A.1), Calibration Analysis (§A.2), and Logit Confidence Analysis (§A.4).

A.1 Structured Format Evaluation

GSM8K (Mathematical Reasoning).

```
[XML AR] You are a math expert. Solve step-by-step in XML format.
  <answer> final numerical answer </answer>
  <reasoning> step-by-step solution </reasoning>
Problem: {problem}
```

```
[XML RA] You are a math expert. Solve step-by-step in XML format.
  <reasoning> step-by-step solution </reasoning>
  <answer> final numerical answer </answer>
Problem: {problem}
```

```
[JSON AR] You are a math expert. Solve step-by-step in JSON format.
  { "answer": "final answer", "reasoning": "solution" }
Problem: {problem}
```

```
[JSON RA] You are a math expert. Solve step-by-step in JSON format.
  { "reasoning": "solution", "answer": "final answer" }
Problem: {problem}
```

```
[Freeform] You are a math expert. Solve step-by-step.
Problem: {problem}
```

MMLU (Multiple-Choice Knowledge).

```
[XML AR] You are a helpful AI assistant. Analyze and provide your response in XML format.
  <answer> option letter (A-D) </answer>
  <reasoning> analysis </reasoning>
Problem: {problem}  Options: {options}
```

```
[XML RA] You are a helpful AI assistant. Analyze and provide your response in XML format.
  <reasoning> analysis and thought process </reasoning>
```

```
  <answer> option letter (A–D) </answer>
Problem: {problem}  Options: {options}
```

```
[JSON AR] You are a helpful AI assistant. Analyze and provide your response in JSON format.
  { "answer": "option letter (A–D)", "reasoning": "explanation" }
Problem: {problem}  Options: {options}
```

```
[JSON RA] You are a helpful AI assistant. Analyze and provide your response in JSON format.
  { "reasoning": "explanation", "answer": "option letter (A–D)" }
Problem: {problem}  Options: {options}
```

```
[Freeform] You are a helpful AI assistant. Solve this MCQ.
Problem: {problem}  Options: {options}
```

LogicBench (Logical Reasoning).

```
[XML AR] You are a reasoning expert. Answer in XML format.
  <answer> option letter </answer>
  <reasoning> analysis </reasoning>
Context: {context}  Problem: {question}  Options: {options}
```

```
[XML RA] You are a reasoning expert. Answer in XML format.
  <reasoning> analysis </reasoning>
  <answer> option letter </answer>
Context: {context}  Problem: {question}  Options: {options}
```

```
[JSON AR] You are a reasoning expert. Answer in JSON format.
  { "answer": "option letter", "reasoning": "explanation" }
Context: {context}  Problem: {question}  Options: {options}
```

```
[JSON RA] You are a reasoning expert. Answer in JSON format.
  { "reasoning": "explanation", "answer": "option letter" }
Context: {context}  Problem: {question}  Options: {options}
```

```
[Freeform] You are a reasoning expert. Answer the following question.
Context: {context}  Problem: {question}  Options: {options}
```

Spider (Text-to-SQL).

```
[XML AR] You are a SQL expert. Generate query in XML format.
  <answer> SQL query </answer>
  <reasoning> rationale </reasoning>
Schema: {schema}  Question: {question}
```

```
[XML RA] You are a SQL expert. Generate query in XML format.
  <reasoning> rationale </reasoning>
  <answer> SQL query </answer>
Schema: {schema}  Question: {question}
```

```
[JSON AR] You are a SQL expert. Return query in JSON.
  { "answer": "SQL query", "reasoning": "explanation" }
Schema: {schema}  Question: {question}
```

```
[JSON RA] You are a SQL expert. Return query in JSON.
  { "reasoning": "explanation", "answer": "SQL query" }
Schema: {schema}  Question: {question}
```

```
[Freeform] You are a SQL expert. Generate a SQL query.
Schema: {schema}  Question: {question}
```

A.2 Calibration Analysis Prompts

Calibration prompts extend each format with an explicit confidence request on a 1–10 scale. Below we show the XML and Freeform variants; JSON calibration follows the same JSON structure as §A.1 with an additional `"confidence"` field.

XML Calibration. All four datasets share the same XML structure extended by a `<confidence>` field:

```
[XML + Confidence] (role and task instruction as above)
  <reasoning> analysis </reasoning>
  <answer> answer </answer>
  <confidence> integer 1–10 (1 = not confident, 10 = very confident) </confidence>
Question fields: identical to Structured Format prompts above.
```

Freeform Calibration.

```
[Freeform + Confidence — MMLU]
You are a helpful AI assistant. Answer the question directly. Before you finish, state your confidence level on a scale of 1–10 (1 = not confident, 10 = very confident).
Problem: {problem} Options: {options}
```

```
[Freeform + Confidence — GSM8K]
You are a math expert. Solve step-by-step. At the end, state your confidence level on a scale of 1–10.
Problem: {problem}
```

```
[Freeform + Confidence — LogicBench]
You are a reasoning expert. Answer the following question. State your confidence on a scale of 1–10.
Context: {context} Problem: {question} Options: {options}
```

```
[Freeform + Confidence — Spider]
You are a SQL expert. Generate a SQL query. Before you finish, state your confidence on a scale of 1–10.
Schema: {schema} Question: {question}
```

A.3 Schema-violation statistics

Structured-output schema violations were zero for all experiments. The only parsing failures occurred in free-form outputs due to the Gemma3-4B.

Dataset	Empty Extraction Rate
MMLU	**29.8%**
LogicBench	**45.2%**

Table 3: Free-form empty-extraction rates caused by the Gemma3-4B parser bug.

A.4 Logit Confidence Analysis Prompts

Logit-based confidence extraction uses minimal prompts that elicit a single-token answer, enabling direct extraction of confidence from the token probability distribution over {A, B, C, D}.

```
[Logit — MMLU & LogicBench]
Answer the following question with only the single letter of the correct option (A, B, C, or D). Do not include any explanation or punctuation — just the letter.
Problem: {problem} Options: {options} Answer:
```

Confidence extraction. For logit prompts, we record the softmax probability assigned to each option token at the `Answer:` position. For XML/JSON prompts, we parse the `<confidence>` or `"confidence"` field and normalise to $[0, 1]$. For freeform prompts, confidence is extracted via regex (e.g., `"confidence[:\s]+(\d+)"`).

Reproducibility. All experiments use temperature $= 0.0$ to ensure deterministic outputs. Prompts are identical across models; no model-specific tuning was applied.

B Dataset summary statistics

Dataset	Split	Examples	Task Type	Answer Format
GSM8K	Test	1,319	Math reasoning	Numeric
LogicBench	Eval	500	Logical inference	MCQ
MMLU	Validation	1,531	Knowledge QA	MCQ
Spider	Validation	1,030	Text-to-SQL	SQL query

Table 4: Dataset statistics and properties. All splits are held-out evaluation sets; no training data is used. Spider is evaluated via execution accuracy; all other datasets use exact-match accuracy.

C Calibration: Supplementary Details

C.1 Implementation Details

Calibration is measured with Expected Calibration Error (ECE) using $M = 10$ equal-width bins over $[0, 1]$, computed as ECE $= \sum_{m=1}^{10} \frac{|B_m|}{N} |\text{acc}(B_m) - \text{conf}(B_m)|$, where the final bin is closed on both sides. Confidence is elicited as a verbal integer on a 1–10 scale ("1 = not confident, 10 = very confident") and normalized by dividing by 10. Scores are extracted from free-text responses via regex matching on the pattern `confidence[:\s]+(\d+)`. All API calls use `temperature=0`. **Brier Score** measures mean squared error between predicted probabilities and binary correctness: $\frac{1}{N}\sum_i (p_i - y_i)^2$. **Entropy** $H(p) = -\sum_y p(y|x_i) \log p(y|x_i)$ measures predictive uncertainty; used under relative confidence elicitation.

C.2 Confidence Elicitation: Absolute vs. Relative

We compare two elicitation strategies on MMLU and LogicBench (MCQ tasks only, as relative prompting requires a fixed option set). **Absolute** prompting requests a single scalar score. **Relative** prompting asks the model to distribute confidence across the answer options (A/B/C/D), from which we extract the chosen-option probability. Table 5 reports ECE, average confidence, entropy, and Brier score.

Absolute prompting achieves lower or equal ECE in 6 of 8 model–dataset pairs. Relative prompting produces lower average confidence (-0.035 to -0.16) and higher entropy (0.47–1.14), indicating models express greater distributional uncertainty, but this does not translate into better-calibrated scalar estimates for most models. The Brier score under relative prompting is consistently worse for weaker models (**Llama-3.1-8B**: 0.452–0.582), while Claude-Sonnet-4.5 achieves the lowest Brier score under relative prompting on both datasets, suggesting stronger models can exploit distributional elicitation more effectively. **Gemma3-4B** is an exception to the absolute-wins pattern: relative prompting reduces its ECE by $\Delta = +0.138$ (LogicBench) and $+0.083$ (MMLU), the two largest per-model improvements in the comparison, consistent with its tendency toward overconfident scalar responses.

C.3 Verbal Confidence vs. Internal Probabilities

We compare verbal 1–10 scores against logit-based confidence the softmax probability assigned to the selected answer token on MMLU and LogicBench for GPT-4o, **Llama-3.1-8B**, and **Gemma3-4B** (the models for which logits are accessible).

Results show a clear model-size gradient. GPT-4o verbal and logit confidence are moderately correlated ($r_s = 0.44$ on MMLU, 0.67 on LogicBench) with low KL divergence (≈ 0.5), suggesting verbal scores partially reflect internal probability estimates. **Llama-3.1-8B** breaks down on MMLU ($r_s = -0.05$, $p = .07$, KL $= 3.62$): verbal confidence is effectively decoupled from token-level probabilities, yet logit ECE (0.156) is better than verbal ECE (0.191), making logits the more reliable signal for this model. **Gemma3-4B** is the most extreme case: near-zero Spearman on MMLU ($r_s = 0.024$, not significant, KL $= 1.676$), and logit ECE (0.402) is far worse than verbal ECE (0.212) logits are dominated by the token-level distribution rather than calibrated uncertainty, while verbal confidence, though imprecise, is at least weakly correlated with accuracy. On LogicBench, all models improve: GPT-4o reaches $r_s = 0.67$,

Dataset	Model	ECE ↓			AvgConf		Entropy ↓	Brier ↓
		Abs.	Rel.	$\Delta^\dagger$	Abs.	Rel.		
LogicBench	GPT-4o	0.072 ±.010	0.084 ±.017	−0.012	0.883 ±.006	0.741 ±.005	1.141 ±.015	0.297 ±.023
	Gemini-Flash-2.0	0.094 ±.012	**0.052** ±.005	+0.042	0.941 ±.004	0.798 ±.004	0.930 ±.027	0.244 ±.014
	Claude-Sonnet-4.5	**0.045** ±.007	0.066 ±.010	−0.021	0.879 ±.007	0.834 ±.003	0.759 ±.014	**0.200** ±.016
	Llama-3.1-8B	0.166 ±.015	0.161 ±.015	+0.005	0.914 ±.006	0.878 ±.007	**0.496** ±.019	0.452 ±.036
	Gemma3-4B	0.290 ±.012	0.152 ±.017	**+0.138**	0.938 ±.002	0.775 ±.002	1.046 ±.010	0.530 ±.019
MMLU	GPT-4o	0.063 ±.007	0.117 ±.007	−0.053	0.924 ±.002	0.750 ±.005	1.142 ±.010	0.244 ±.011
	Gemini-Flash-2.0	0.103 ±.008	0.078 ±.011	+0.025	0.969 ±.002	0.803 ±.004	0.921 ±.010	0.242 ±.007
	Claude-Sonnet-4.5	**0.024** ±.005	**0.057** ±.005	−0.033	0.933 ±.003	0.864 ±.003	**0.671** ±.002	**0.147** ±.007
	Llama-3.1-8B	0.247 ±.008	0.192 ±.010	+0.055	0.927 ±.001	0.834 ±.004	0.682 ±.004	0.582 ±.018
	Gemma3-4B	0.281 ±.016	0.198 ±.023	**+0.083**	0.948 ±.002	0.744 ±.012	1.093 ±.023	0.591 ±.027

Table 5: Absolute vs. relative confidence elicitation on MCQ datasets. $^\dagger\Delta = \text{ECE}_{\text{Abs}} - \text{ECE}_{\text{Rel}}$; positive = relative prompting improves calibration. Entropy and Brier defined only under relative prompting. Highlighted Δ values mark the largest per-column improvements. Values are mean over 5 runs (std. in gray).

Dataset	Model	Correlation & Divergence			ECE ↓		Brier Score ↓	
		Spearman ρ	Pearson r	Mean KL	Logit	Verbal	Logit	Verbal
MMLU	GPT-4o	0.439	0.128	0.500	0.117	0.116	0.276	0.247
	Llama-3.1-8B	$-0.049^\dagger$	$-0.040^\dagger$	3.617	0.156	0.191	0.496	0.582
	Gemma3-4B	$0.024^\dagger$	$0.002^\dagger$	1.676	0.402	0.212	0.813	0.602
LogicBench	GPT-4o	0.673	0.312	0.423	0.122	0.084	0.274	0.305
	Llama-3.1-8B	0.282	0.230	2.304	0.141	0.162	0.379	0.453
	Gemma3-4B	0.148	0.099	0.757	0.311	0.146	0.627	0.532

Table 6: Logit vs. verbal confidence comparison. Spearman ρ and Pearson r measure scalar agreement on the chosen-option score; Mean KL measures full distributional divergence. ECE and Brier assess calibration against correctness. † $p > 0.05$.

and even **Gemma3-4B** achieves a moderate Spearman (0.148), suggesting task difficulty moderates verbal-logit alignment.

C.4 Calibration Visualisations

Figure 4 plots mean confidence against mean accuracy for each model–dataset combination under freeform (squares) and XML (circles). All models cluster in the high-confidence region ($> 85\%$) regardless of format while accuracy spans a wide range, placing most points well above the diagonal that marks perfect calibration. The format effect appears as a systematic downward-and-rightward shift: XML moves points closer to the diagonal, reducing the confidence–accuracy gap across all conditions. The displacement is largest for weaker models on low-structure tasks, consistent with the ECE reductions in Table 2. GSM8K points cluster near the diagonal under both formats, confirming it as the best-calibrated task irrespective of schema.

Figure 5 shows per-bin reliability diagrams for all 20 model–dataset conditions. Under freeform, high-confidence bins are heavily populated while empirical accuracy lags, producing bars far below the diagonal the visual signature of overconfidence. XML shifts bars toward the diagonal across most configurations, with the largest recalibration on LogicBench and MMLU where freeform calibration is worst. GSM8K bars are near-diagonal under both formats, reinforcing the format-invariance of well-structured tasks. Together, the two figures confirm that the ECE reductions in Table 2 and Figure 2 reflect genuine per-bin recalibration rather than an artefact of aggregation.

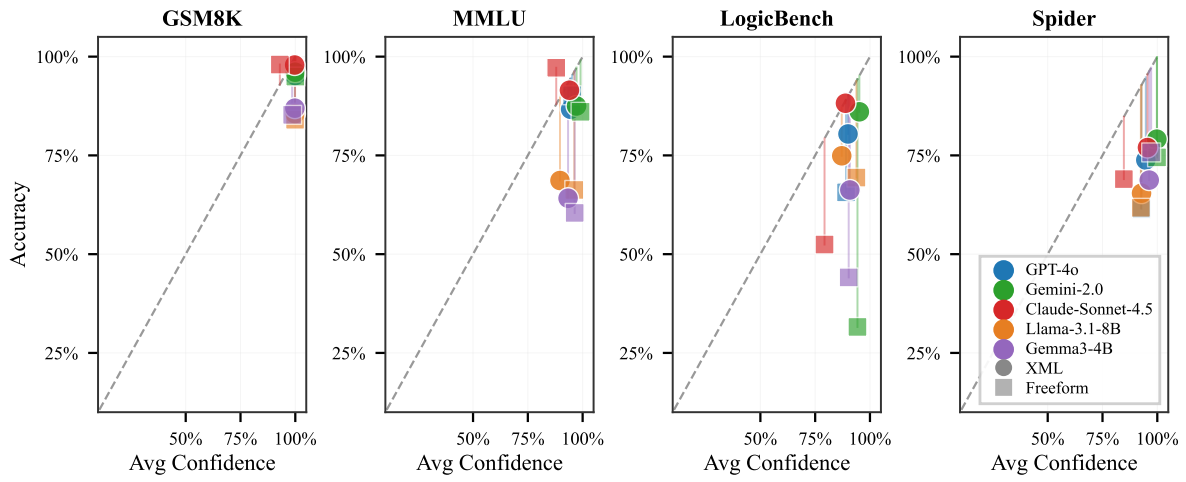


Figure 4: Accuracy vs. confidence scatter for all 5 models across 4 datasets under freeform (squares) and XML (circles) prompting. All models express high confidence ($> 85\%$) regardless of format, while accuracy varies substantially. The gap between confidence and the diagonal is consistently larger under freeform, most dramatically for **Gemma3-4B** on MMLU and LogicBench ($<26\%$ accuracy, $\sim 100\%$ confidence) and **Llama-3.1-8B** on LogicBench (23% accuracy, 93% confidence). XML structured output reduces overconfidence across all models, bringing points closer to the calibration diagonal.

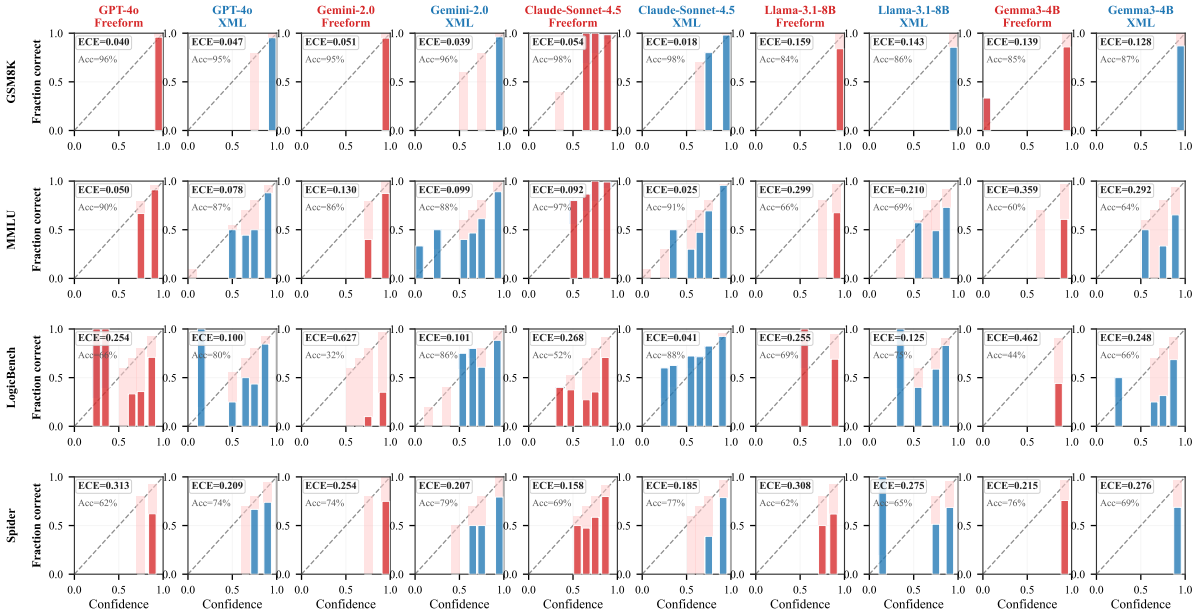


Figure 5: Reliability diagrams comparing freeform and XML calibration across 20 model–dataset combinations (5 models × 4 datasets, each shown in freeform and XML format). XML bars align closer to the diagonal in most configurations, with the largest improvements on LogicBench where freeform produces severe overconfidence (ECE up to 0.703 for **Llama-3.1-8B** and 0.645 for **Gemma3-4B**). **Gemma3-4B** shows the highest overall freeform ECE, particularly on MMLU (0.721). GSM8K remains comparatively well-calibrated under both formats across all models.

D CKA and Probe: Method Details

D.1 CKA Extraction

For each question we run a single forward pass through the frozen model with no generation and extract the *last-token* hidden state at every layer ℓ, yielding representation matrices $\mathbf{X}_{\mathrm{C}}^{\ell}, \mathbf{X}_{\mathrm{I}}^{\ell} \in \mathbb{R}^{n \times d}$ for the correct and incorrect prediction groups respectively ($n = 200$ examples per dataset, sampled uniformly). Inputs

are formatted identically to the main benchmark runs i.e. XML or freeform, ensuring representational differences are attributable to output format, not input variation. We apply this to Llama-3.1-8B ($d = 4096$, 33 layers) and Gemma3-4B ($d = 2560$, 35 layers).

We compute linear CKA via the HSIC estimator:

$$\text{CKA}(\mathbf{X}, \mathbf{Y}) = \frac{\text{HSIC}(\mathbf{X}\mathbf{X}^\top, \mathbf{Y}\mathbf{Y}^\top)}{\sqrt{\text{HSIC}(\mathbf{X}\mathbf{X}^\top, \mathbf{X}\mathbf{X}^\top) \cdot \text{HSIC}(\mathbf{Y}\mathbf{Y}^\top, \mathbf{Y}\mathbf{Y}^\top)}}$$

where $\text{HSIC}(K, L) = \frac{1}{(n-1)^2}\text{tr}(K_c L_c)$ and subscript c denotes centering. This yields three quantities per layer: (i) a per-layer *cross-group* CKA scalar measuring geometric alignment between correct and incorrect representations; (ii) a within-correct inter-layer CKA matrix; and (iii) an analogous within-incorrect matrix. We partition layers into three processing stages Early, Middle, and Late at roughly equal thirds of network depth.

Within-group inter-layer CKA matrices are visualised in Figures 6–7 (Llama-3.1-8B) and Figures 8–9 (Gemma3-4B); see §D.2 for layout and reading guide.

D.2 Within-Group CKA Heatmaps: Layout and Reading Guide

Matrix construction. For each (model, dataset, format) triple we partition the $n = 200$ probe examples into two outcome groups: *correct* (C, examples the model answered correctly) and *incorrect* (I). For each group $g \in \{C, I\}$ we collect the last-token hidden state at every layer $\ell \in \{0, \ldots, L-1\}$, yielding $\mathbf{X}_g^\ell \in \mathbb{R}^{n_g \times d}$. The within-group inter-layer CKA matrix $\mathbf{M}_g \in \mathbb{R}^{L \times L}$ then has entry

$$\mathbf{M}_g[i, j] \;=\; \text{CKA}\big(\mathbf{X}_g^i, \mathbf{X}_g^j\big)\,,$$

i.e. the linear CKA similarity between the population of layer-i and layer-j representations for examples in group g. This produces two $L \times L$ matrices per (format, dataset): one for the correct group and one for the incorrect group. We additionally show their difference, $\mathbf{D} = \mathbf{M}_C - \mathbf{M}_I$, which isolates representational-geometry differences attributable to outcome.

Panel layout. Each figure is a 2×3 grid:

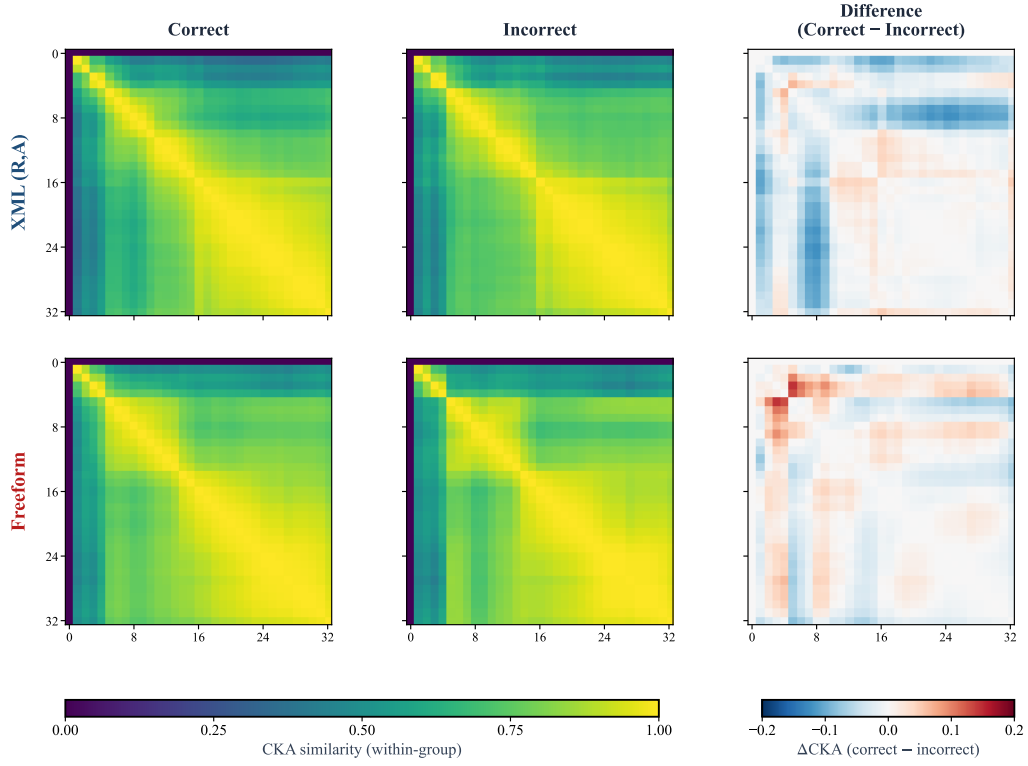
- **Rows:** XML (R,A) (top), Freeform (bottom). Both rows use identically-sampled inputs; only the output format differs.
- **Columns:** (1) within-correct $\mathbf{M}_C$; (2) within-incorrect $\mathbf{M}_I$; (3) difference $\mathbf{D} = \mathbf{M}_C - \mathbf{M}_I$.
- **Colormaps:** columns 1–2 use `viridis` on $[0, 1]$ (CKA is in $[0, 1]$ by construction); column 3 uses a diverging `RdBu_r` colormap symmetric about zero, with limits ± 0.20 for Llama and ± 0.40 for Gemma to accommodate the larger inter-layer variance of the smaller model.
- **Axes:** both axes index layer number, 0 to $L - 1$; the matrices are symmetric about the main diagonal by construction.

Reading guide.

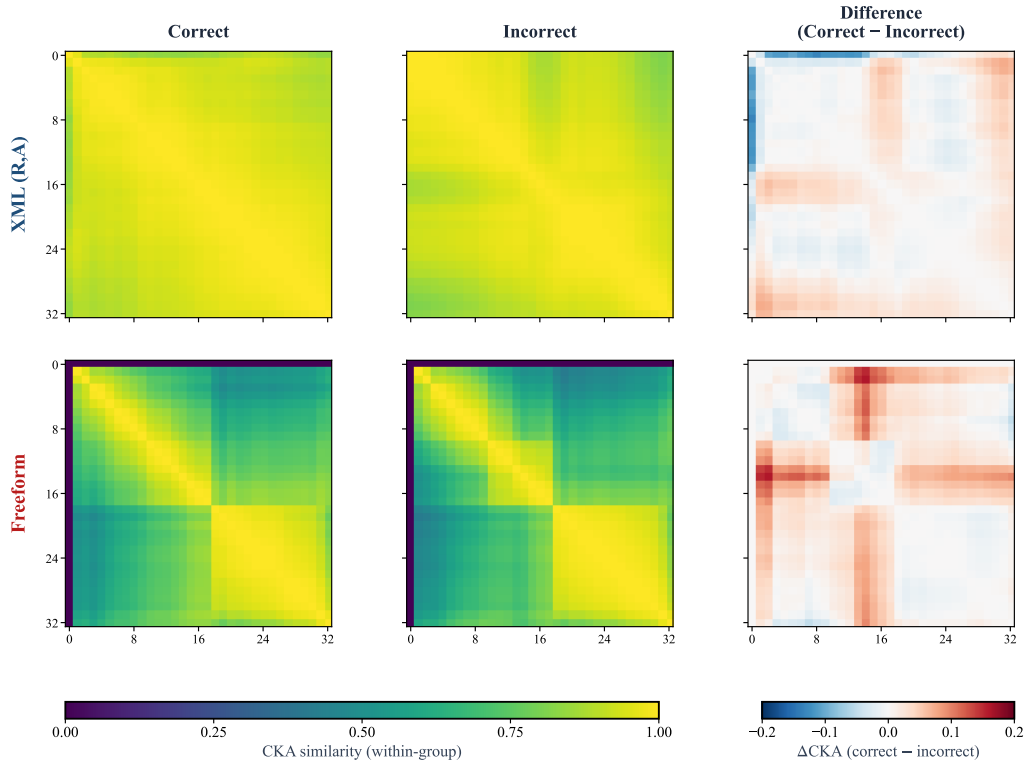
- **Block-diagonal structure** in the within-group panels (cols 1–2) indicates the model organises its forward pass into discrete representational stages: a contiguous run of layers that are mutually similar (a bright block on the diagonal) followed by an abrupt transition to a new regime. Three blocks is typical, corresponding to early (input encoding), middle (task computation), and late (output preparation) stages.
- **Off-diagonal bright patches** indicate two non-adjacent layers maintain similar representations – a residual-stream signature. Conversely, dark off-diagonal patches mark regions where representations have substantially diverged.
- **Difference panel (col 3):** blue cells indicate the correct group has *higher* within-layer similarity than the incorrect group at that (i, j) location; red cells indicate the reverse. Saturation tracks magnitude.

What to attend to.

- **Block-diagonal structure across rows.** Compare the top (XML R,A) and bottom (Freeform) rows in cols 1–2. On low-structure tasks (MMLU, LogicBench), XML (R,A) tends to produce a more crisply delineated three-block pattern in the correct group than Freeform does, consistent with structured prompting inducing cleaner stage-wise processing.
- **Asymmetry between correct and incorrect.** Within a single row, compare cols 1 vs. 2: a prominent block-diagonal pattern in the correct panel than in the incorrect panel suggests that correct answers are produced by more discretely-staged computation, while incorrect answers reflect noisier inter-layer transitions.
- **Difference panel as a summary.** A predominantly-blue difference panel indicates the correct group consistently has higher within-stage coherence than the incorrect group. A predominantly-red or near-white difference panel indicates outcome does not separate the representational geometry.
- **Where the difference concentrates.** A blue patch in the upper-left of $\mathbf{D}$ (early layers) suggests divergence begins early in the network; a mid-matrix concentration localises the differentiation to the task-computation stage.
- **Spider as the high-structure baseline.** On Spider, both formats and both outcome groups exhibit similar, less-pronounced block structure and a near-white difference panel – the task's inherent SQL scaffolding already constrains representations, leaving little room for output format to modulate the internal geometry.

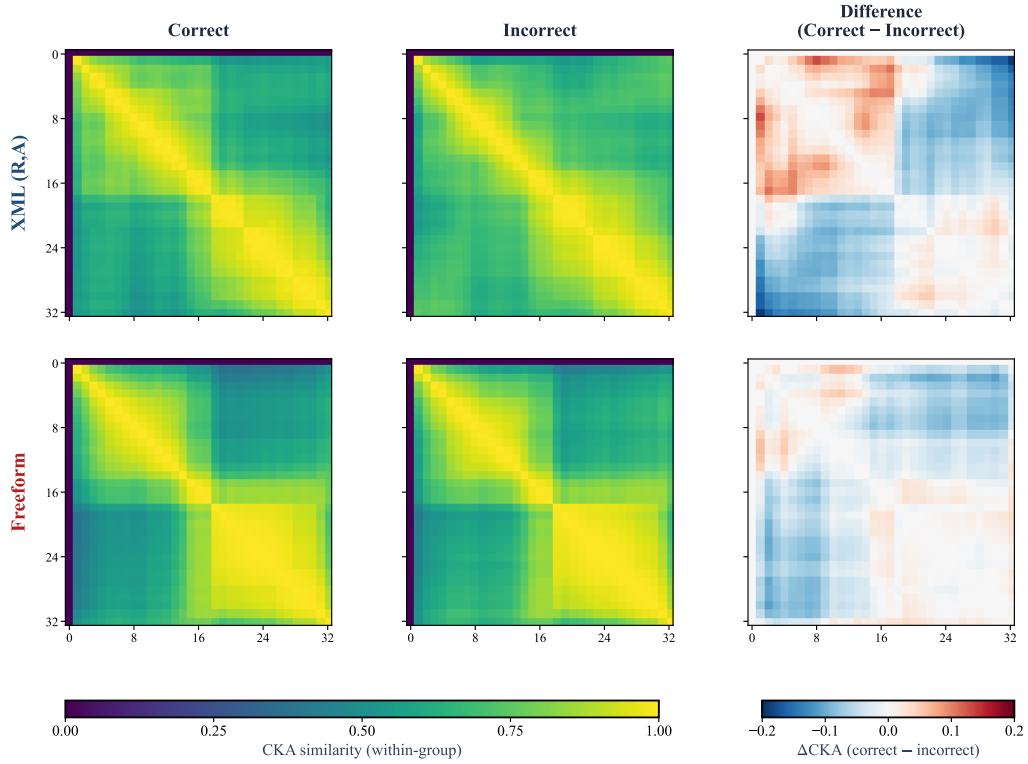


(a) GSM8k

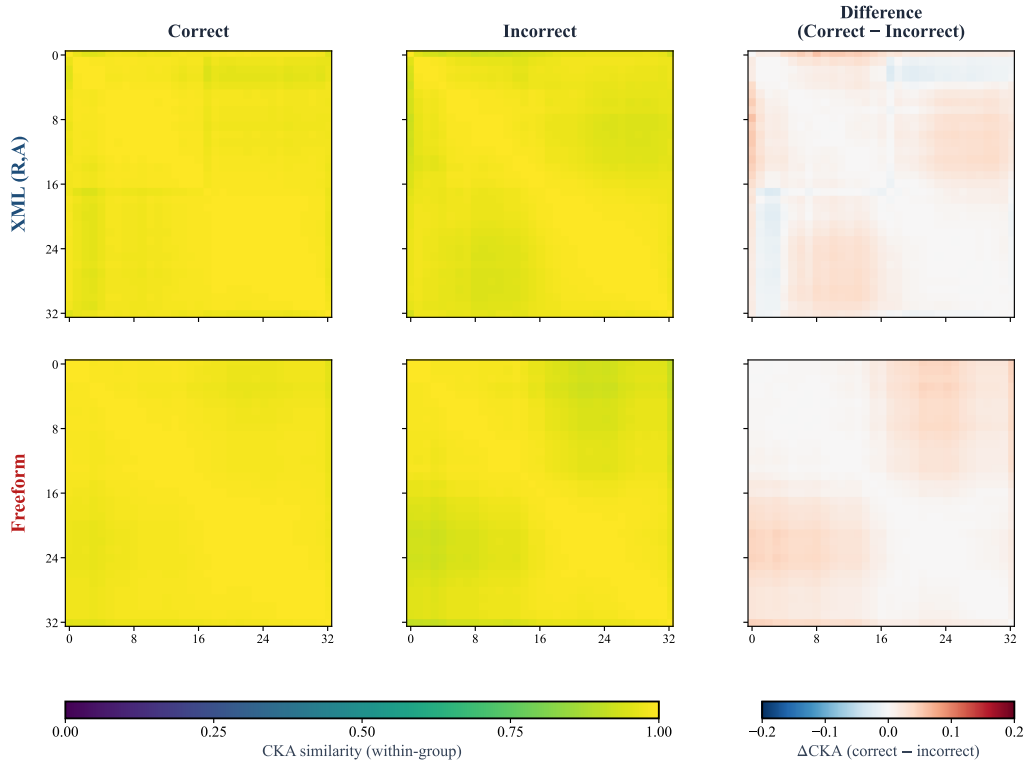


(b) MMLU

Figure 6: Llama-3.1-8B within-group CKA heatmaps for (a) GSM8K and (b) MMLU. Each panel shows XML (R,A) (top row) and Freeform (bottom row) across three views: correct answers only, incorrect answers only, and their difference (correct − incorrect). See §D.2 for construction and reading guide.

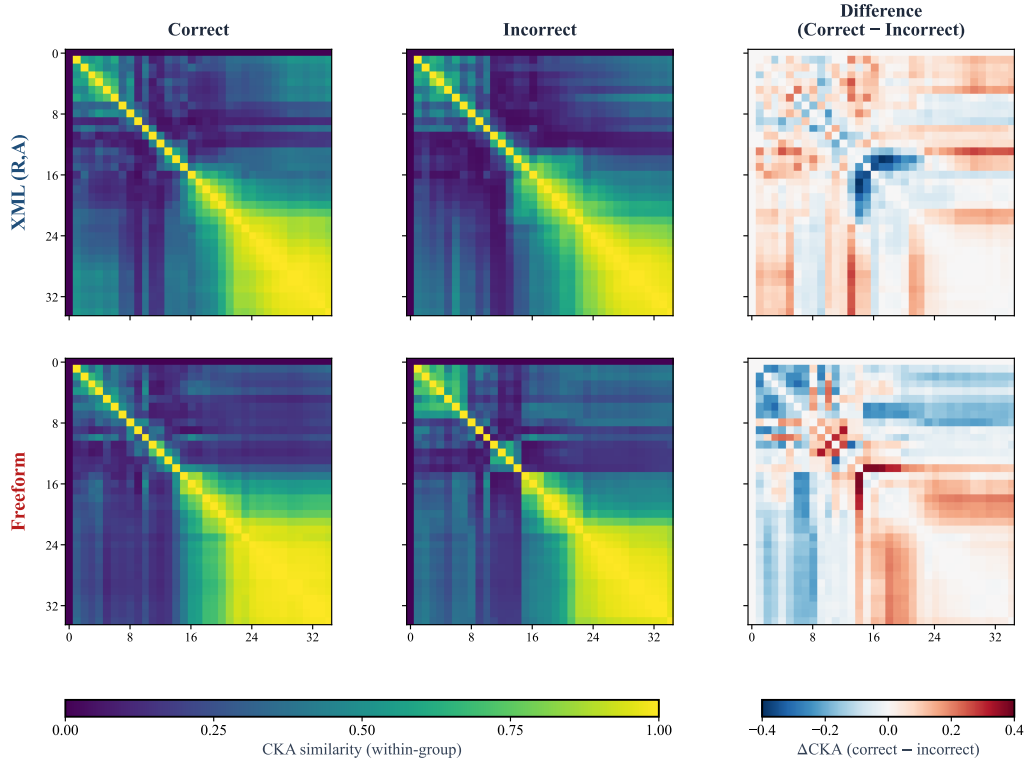


(a) LogicBench

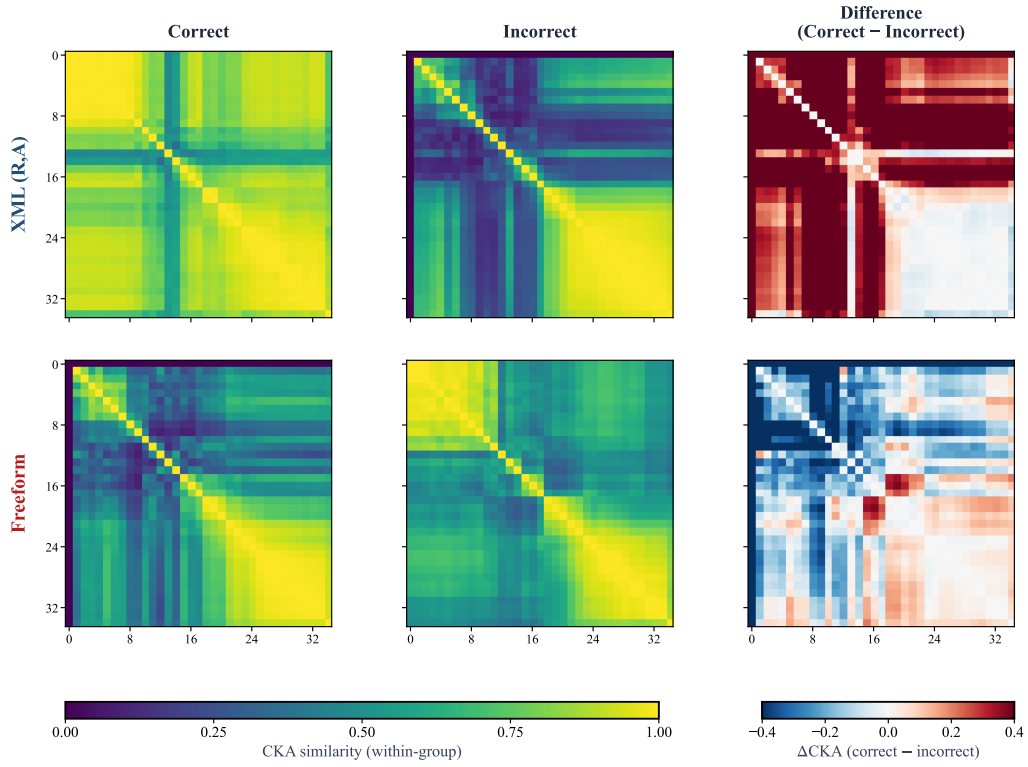


(b) Spider

Figure 7: Llama-3.1-8B within-group CKA heatmaps for (a) LogicBench and (b) Spider. Same layout as Figure 6; see §D.2. Spider shows weak differentiation between outcome groups under both formats, consistent with the high-structure task pattern.

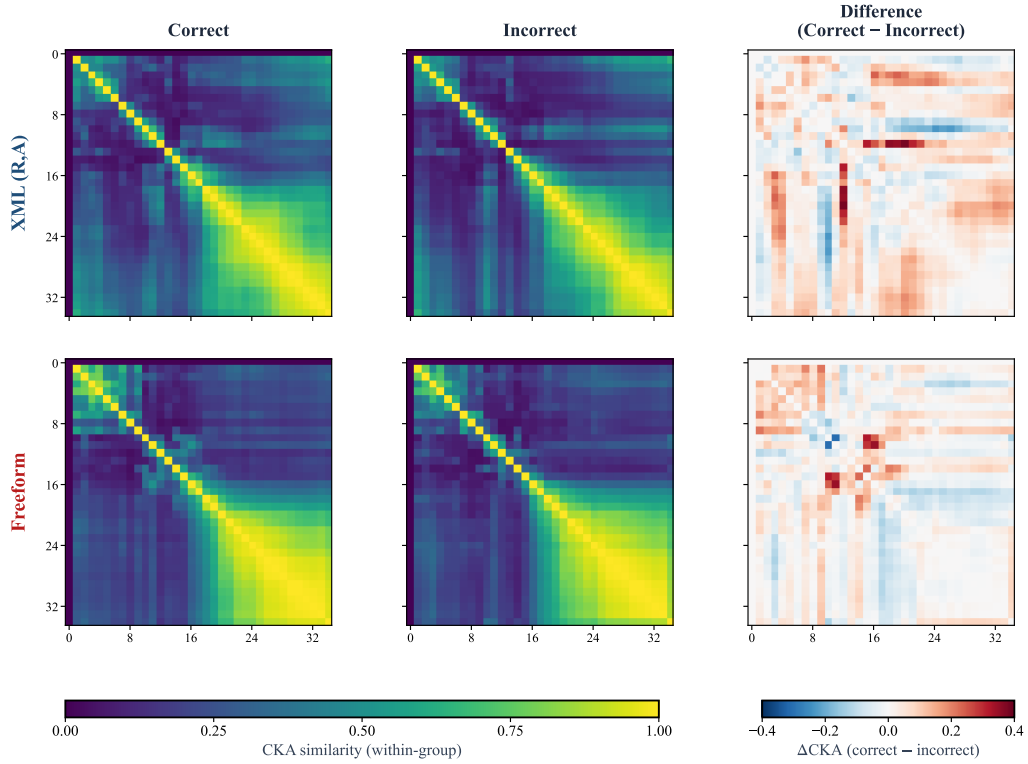


(a) GSM8K

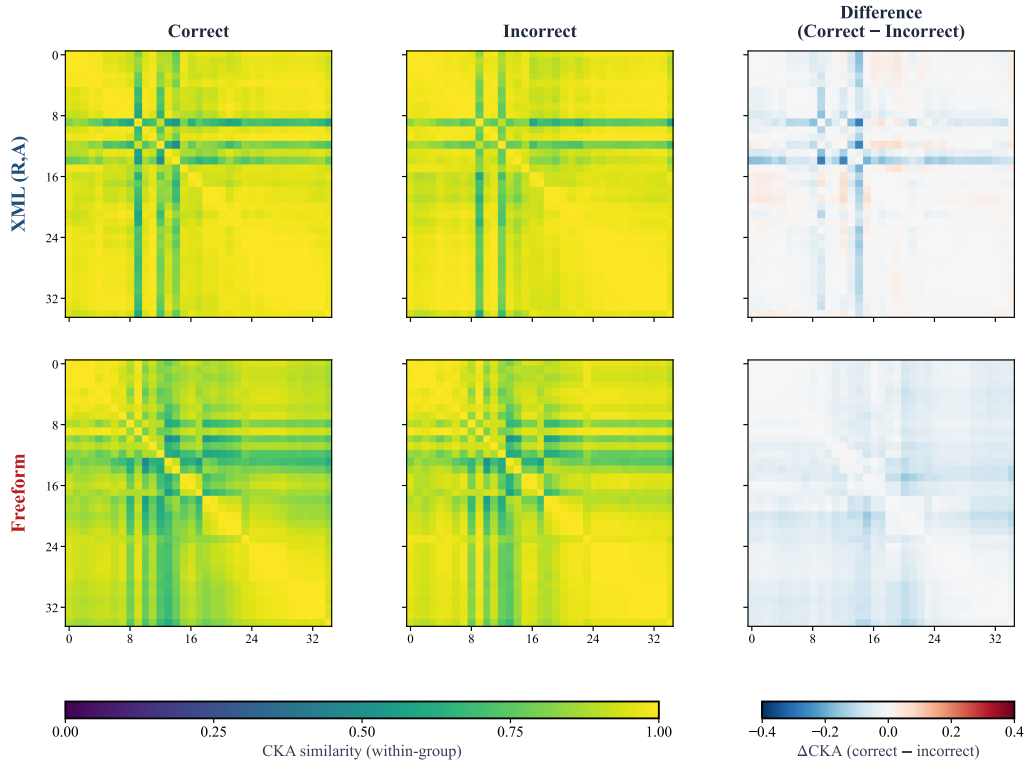


(b) MMLU

Figure 8: Gemma3-4B within-group CKA heatmaps for (a) GSM8K and (b) MMLU. Each panel shows XML (R,A) (top row) and Freeform (bottom row) across correct answers, incorrect answers, and their difference (correct − incorrect). See §D.2 for construction and reading guide.



(a) LogicBench



(b) Spider

Figure 9: Gemma3-4B within-group CKA heatmaps for (a) LogicBench and (b) Spider. Same layout as Figure 8; see §D.2. Spider shows weak differentiation between outcome groups under both formats, consistent with the high-structure task pattern.

D.3 Linear Probe Setup

We train a balanced linear probe (logistic regression with ℓ_2 regularisation, $C = 1.0$, 5-fold cross-validation) on the pre-answer hidden states at each layer to predict the model's own predicted label. Hidden states are extracted from the last token of the reasoning field, immediately before the answer field begins, under both (R,A) and (A,R) orderings.

In both orderings, the probe reads the hidden state at the token immediately preceding the answer letter: the > closing the <answer> opening tag. This is offset -1 relative to the answer-letter token.

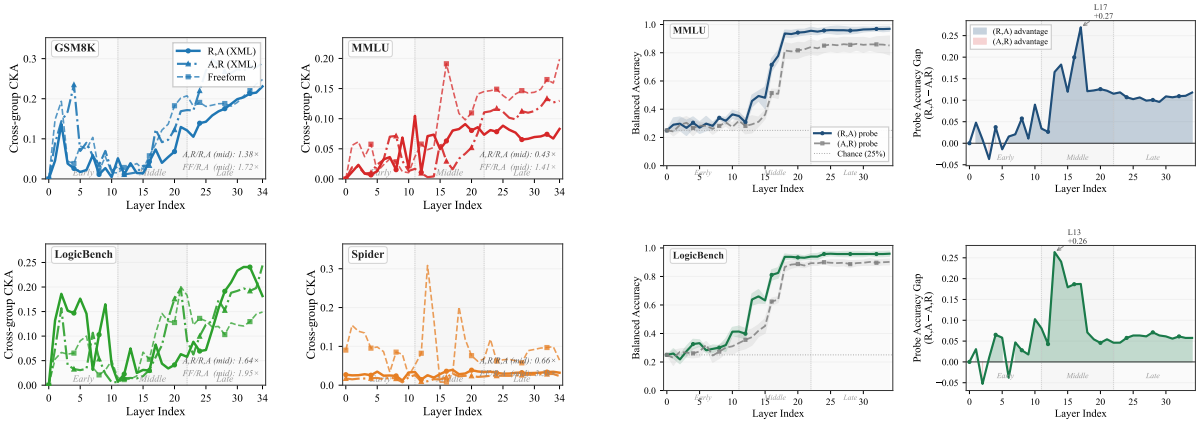
Ordering	Probe Token	Offset	Preceding Context
(R,A)	> in <answer>	-1	Full reasoning block
(A,R)	> in <answer>	-1	No reasoning; answer is the first output field

Table 7: Probe location relative to the answer token under different field orderings.

We report balanced accuracy to account for class imbalance across answer options. Probes are run on MMLU and LogicBench for Llama-3.1-8B and Gemma3-4B; GSM8K is excluded as it requires a binary correct/incorrect probe rather than a label-identity probe, and Spider is excluded due to its open-ended SQL output space. Hidden states for both models are extracted via forward pass on NVIDIA T4 GPUs.

D.4 Gemma3-4B: Representational Results

Figure 10 replicates the **Llama-3.1-8B** analysis of Section 6 for **Gemma3-4B** ($d = 2560$, 35 layers). The inverse-structure gradient in cross-group CKA is preserved: MMLU and LogicBench exhibit lower mid-layer CKA under XML (R,A) than freeform, while Spider again reverses the pattern qualitatively identical to Llama but at reduced absolute magnitude, consistent with Gemma's smaller overall accuracy gap under (R,A) in Table 1. Linear probes tell the same story: the (R,A) advantage in decoding answer identity from intermediate representations peaks at $+0.27$ on MMLU (L17) and $+0.26$ on LogicBench (L13), versus $+0.64$ and $+0.70$ for Llama directionally aligned but attenuated, as expected for a model with fewer parameters and shallower representational depth. The convergence across two architecturally distinct open-weight models strengthens the interpretation that mid-layer answer encoding under (R,A) is a general consequence of field ordering rather than a property of any single architecture.



(a) Cross-group CKA per layer

(b) Linear probe balanced accuracy per layer

Figure 10: Internal representations for **Gemma3-4B**, same layout as Figure 3. **Left**: Cross-group CKA preserves the inverse-structure gradient of **Llama-3.1-8B** MMLU and LogicBench show lower mid-layer CKA under XML (R,A) than freeform; Spider reverses the pattern at reduced overall magnitude. **Right**: Linear probe balanced accuracy on MMLU and LogicBench. The (R,A) advantage is directionally consistent with Llama but smaller in magnitude (peak gap $+0.27$ on MMLU at L17; $+0.26$ on LogicBench at L13), in line with Gemma's smaller behavioural gap under R,A in Table 1.

E Reasoning Length: Measurement and Results

Reasoning length is measured by extracting the reasoning field from each model response and counting words via `str.split()`, which closely tracks subword-token count under standard tokenizers. For XML formats we extract the content of the `<reasoning>` tag; for JSON formats we parse the `"reasoning"` key, with a regex fallback for malformed outputs. For Spider under JSON (A,R), `parsed_reasoning` is empty in the primary data file and is extracted directly from raw response CSVs. This yields 80 cells (5 models $\times$ 4 datasets $\times$ 4 format-orderings); each cell reports median word count and mean accuracy over all examples in that condition.

Tables 8 and 9 report the full per-cell medians and Pearson correlations discussed in §6.3.

Dataset	XML (R,A)	XML (A,R)	JSON (R,A)	JSON (A,R)
GSM8K	126	127	**108**	123
MMLU	121	99	121	**33**
LogicBench	114	**92**	113	96
Spider	**54**	58	62	64

Table 8: Median reasoning-field word count per format–ordering, aggregated over 5 models. **Bold** = shortest per dataset. JSON (A,R) collapses to 33 words on MMLU vs. 121 under XML (R,A); GSM8K is flat across all variants.

By dataset			By format–ordering		
Dataset	r	n	**Format**	r	n
GSM8K	-0.11	20	XML (R,A)	$+0.42$	20
MMLU	$+0.11$	20	XML (A,R)	$+0.28$	20
LogicBench	$+0.06$	20	JSON (R,A)	$+0.25$	20
Spider	$+0.16$	20	JSON (A,R)	-0.73^{**}	20
Overall	$+0.04$	80			

Table 9: Pearson r between median reasoning length and accuracy across 80 cells (5 models $\times$ 4 datasets $\times$ 4 format-orderings). $^{**}p<0.01$. Conditioning on dataset yields no signal; conditioning on format-ordering reveals JSON (A,R) as the sole significant predictor and it is negative.

F Prompt-Brittleness Ablation

Motivation. LLM behaviour is known to be sensitive to surface-level prompt variations. Small changes in instructional phrasing, schema tag casing, or whitespace can shift accuracy by several points without changing semantics as emperically demonstrated by (Sclar et al., 2024; He et al., 2024). A natural concern is whether the (R,A) vs. (A,R) accuracy gap reported in Section 4 is similarly an artefact of the specific system prompts used. We address this concern with a focused robustness study.

Setup. We hold the model, dataset, and field ordering fixed and perturb only surface features of the system prompt. For each of the 5 formats (Free-form, JSON (A,R), JSON (R,A), XML (A,R), XML (R,A)), we construct three perturbed variants:

- **V1 Instructional phrasing.** Paraphrase the role and task sentence ("You are a math expert. Solve the problem step-by-step ..." $\rightarrow$ " Please solve the following mathematics problem carefully. Work through your reasoning ..."). Schema definition is byte-identical to baseline.
- **V2 Surface casing.** For XML, tag names become PascalCase (`<reasoning>` $\rightarrow$ `<Reasoning>`); for JSON, keys become PascalCase (`"answer"` $\rightarrow$ `"Answer"`); for Free-form, instruction-frame headers are uppercased. Instructional sentence is preserved.
- **V3 Whitespace.** For XML, blank lines around the schema are removed (compact form); for JSON, the schema definition is collapsed to a single line; for Free-form, additional blank lines are inserted around the separator. All wording and schema names are preserved.

We evaluate each (model, format, variant) cell on GSM8K dataset, across two models: Llama-3.1-8B (the model with the largest (A,R) collapse in the main results) and GPT-4o (a frontier API model).

Model	Format	Base	V1	V2	V3
Llama-3.1-8B	Free-form	77.5	81.0	78.5	80.0
	JSON (A,R)	32.0	35.0	34.5	36.0
	JSON (R,A)	81.5	86.0	74.5	77.0
	XML (A,R)	10.5	11.5	11.0	11.5
	XML (R,A)	84.0	87.5	82.0	82.0
GPT-4o	Free-form	90.0	86.0	90.0	86.5
	JSON (A,R)	59.0	56.5	55.5	56.5
	JSON (R,A)	96.0	98.0	96.0	97.0
	XML (A,R)	57.0	53.5	55.5	53.5
	XML (R,A)	96.0	97.0	98.0	98.5

Table 10: GSM8K accuracy (%) dataset under three prompt-perturbation variants. **Base** = original prompt used in the main results. Within each (model, format-family) pair, the (R,A) row consistently exceeds the (A,R) row across all four conditions.

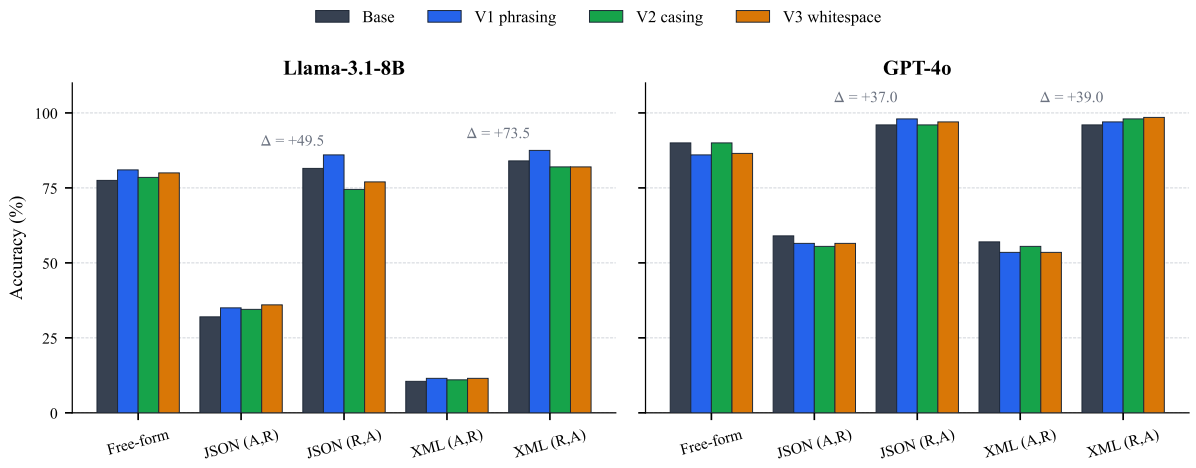


Figure 11: GSM8K accuracy under prompt perturbations. Each group of bars is one (model, format) cell; within-group order is baseline, V1, V2, V3. The (R,A) − (A,R) gap (annotated Δ over each XML / JSON family) is preserved across all perturbations: **the sign of the field-ordering effect is invariant to surface prompt changes**, even though absolute accuracy shifts by a few points.

Findings. **(1) Sign of the gap is preserved.** For every (model, format-family, variant) cell, accuracy under (R,A) exceeds accuracy under (A,R); no perturbation inverts the ordering. The (R,A) − (A,R) gap on Llama XML is +73.5 pts (range across variants: +70.5 to +76.0); on GPT-4o XML it is +39.0 pts (range +39.0 to +45.0); analogous magnitudes hold for JSON. The specific concern that field ordering's effect could disappear under slight prompt changes is empirically rejected.

(2) Absolute accuracy shifts are small relative to the ordering effect. The largest within-cell spread across {Base, V1, V2, V3} is 11.5 pts (Llama JSON (R,A); Table 10); most cells move by under 5 pts. By contrast, the *smallest* (R,A) − (A,R) gap across all (model, family, variant) combinations is +37 pts (GPT-4o JSON baseline).