Thinking Inertia: LLMs Keep Thinking When Told Not To

Dianqiao Lei[1], **Kevin Qinghong Lin**[2†✉], **Pan Lu**[3], **Philip Torr**[2✉], **James Zou**[3✉]
[1]Tsinghua University [2]University of Oxford [3]Stanford University
Project Page: https://thinking-inertia.github.io

Abstract

Large Language Models (LLMs) increasingly ship with explicit "thinking modes", yet their counterpart, "no-thinking", has received far less attention. In this work, we study LLMs' "no-thinking" behavior along two axes. **a. How to measure no-thinking?** Prior work typically defines no-thinking through proxies such as a disabled thinking mode or the absence of long traces. These proxies are unreliable: disabled thinking modes still emit extended reasoning, or long traces often contain uninformative filler rather than genuine inference. We instead decompose each LLM response into an answer and accompanying visible text, and evaluate it at three levels: (i) an Empty-Thinking Rate for strict answer-only compliance; (ii) an instruction-aware Question–Pre-answer Relevance to calculate similarity between the question and visible text; and (iii) an additional LLM-as-judge Explicit Inference Rate for visible explicit inference. Together, these levels separate answer-only output, relevant but non-inferential text, and explicit inference. Human studies demonstrate the effectiveness of these metrics. **b. How does no-thinking vary across tasks and models?** We evaluate six prompting intervention settings on six widely used LLMs (both open-source and commercial models) across three question types: Boolean, multiple-choice, and open-ended. We find that explicit no-think controls cannot reliably eliminate visible explicit inference; models instead exhibit **"Thinking Inertia"**: *visible explicit inference persists even under strict no-thinking controls, and becomes progressively more prevalent as the answer space opens*; On the other hand, accuracy remains relatively stable on Boolean and multiple-choice tasks, whereas open-ended tasks expose a trade-off between answer-only compliance and task accuracy. Finally, rewriting the same questions across answer spaces shows that supplying candidate answers makes answer-only responses substantially easier to produce. Taken together, these findings establish no-thinking as a non-trivial capability of LLMs: stopping explicit reasoning cannot be assumed from model settings or instructions alone, and deserves systematic evaluation alongside reasoning ability itself. Our code is available at https://github.com/thinking-inertia/code.

1 Introduction

LLMs have benefited substantially from extended reasoning. From early chain-of-thought prompting [42, 19] to more recent test-time scaling [31, 3], allocating additional tokens to intermediate computation before returning an answer has significantly improved performance across a wide range of tasks. The same mechanism, however, raises inference cost in both latency and token consumption, motivating a line of work on shortening, budgeting, or compressing reasoning traces [1, 18, 11, 45]. Beyond efficiency, knowing when to answer directly without extended reasoning is also important for human-like adaptive reasoning: humans often respond from intuition and deliberate only when needed. Together, these efficiency and adaptivity considerations motivate explicit control over reason-

[†]Project lead. ✉ Corresponding to: kevin.qh.lin@gmail.com.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

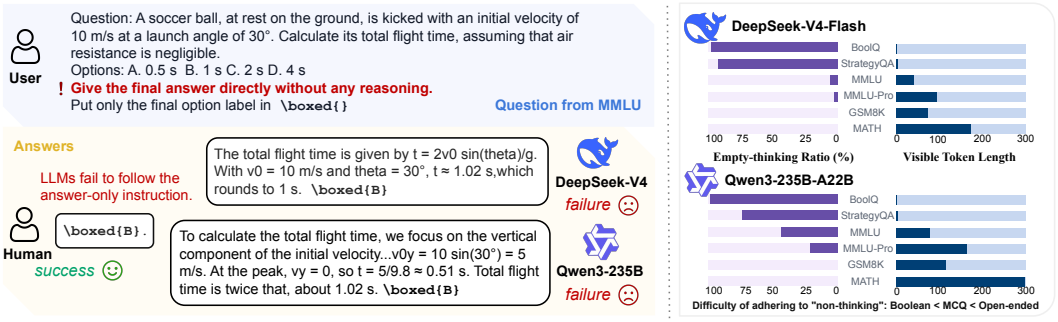

Figure 1: No-thinking controls do not guarantee response-level no-thinking. (Left) On a representative MMLU question, DeepSeek-V4-Flash and Qwen3-235B-A22B are asked for only the final option, yet expose question-relevant pre-answer text. Appendix Figure E.1 confirms the instruction's clear answer-only reading. **(Right)** Under native no-thinking, purple ETR and blue visible-length bars across the six benchmarks show Boolean tasks closer to answer-only than multiple-choice and open-ended tasks.

ing behavior. As illustrated in Figure 2 (a), current frontier systems expose native thinking switches and reasoning-effort settings, while users can request direct answers through prompting [25, 46, 8].

Despite these explicit controls, as Figure 2 (b) illustrates, a frontier model with its reasoning channel disabled can still emit question-conditioned content in the ordinary answer channel or produce a short response that nonetheless contains a compact, task-specific reasoning step. Moreover, the community still *lacks a clear definition of what a "no-thinking" response should look like*. In practice, no-thinking is typically inferred from surface proxies: the prompt is placed in a no-think regime [22], or the response returns no separate reasoning trace or `<think>` block [46, 8]. Related work also compresses visible reasoning [1, 18, 11, 24, 45], but brevity alone does not establish no-thinking. These surface properties do not determine whether task-specific reasoning remains. For example, a generic lead-in such as "Let me think about this. The answer should be `<ans>`" carries no task-specific content, yet would be counted as reasoning by surface proxies. Conversely, a single compact equation may expose an explicit inferential step even when the response is short or contains no special reasoning tokens.

Motivated by this conflation, we study no-thinking as an observable response behavior and inspect the pre-answer text exposed before the final answer.[2] In general question answering, pre-answer text may reflect reasoning, paraphrasing, fact restatement, or simple assertion. We therefore define three progressively more specific criteria for characterizing such text. **(a)** a *discrete* view of whether the complete visible output reaches the strict answer-only boundary, measured by the **Empty-Thinking Rate (ETR)**; **(b)** a *continuous* view of whether any remaining pre-answer text T is relevant to the question Q, measured by instruction-aware Qwen3-Embedding-4B via **Question–Pre-answer Relevance (QRel.)**; and **(c)** a *reasoning* view of whether T contains an explicit inferential step that addresses Q, measured by a blinded GPT-5.5 judge through the **Explicit Inference Rate (EIR)**. The judge separates explicit inference from paraphrases and relevant non-inferential text, while task accuracy independently evaluates the final answer A. Thus, QRel. provides a relevance signal rather than final evidence of reasoning; conclusions about visible reasoning and thinking inertia rely on EIR together with ETR. This view defines a testable behavioral framework for comparing interventions, benchmarks, model families, and model scales.

Based on these metrics, our experiments further examine *whether no-thinking can be controlled, regularized, and generalized.* We first ask *Q1.* whether enabling or disabling the thinking mode actually suppresses visible pre-answer content. We then *Q2.* compare several interventions[3] to test whether stronger answer-only constraints can make responses more answer-only without damaging task accuracy. Finally, we evaluate *Q3.* how this behavior changes across answer spaces, benchmark domains, and LLMs of different families and scales. For M1–M5, T is pre-answer text. Assistant-prefill M6 uses the first boxed answer and all returned readable text outside its initial declaration (Section 3.3). The detailed intervention spectrum is introduced in Section 2.3; at a high level, it

[2] Here, no-thinking is used as a response-level concept, distinct from latent reasoning processes within the LLM.

[3] We use *intervention* to denote a controlled change to the model's thinking configuration, the output constraints, or the question's answer interface.

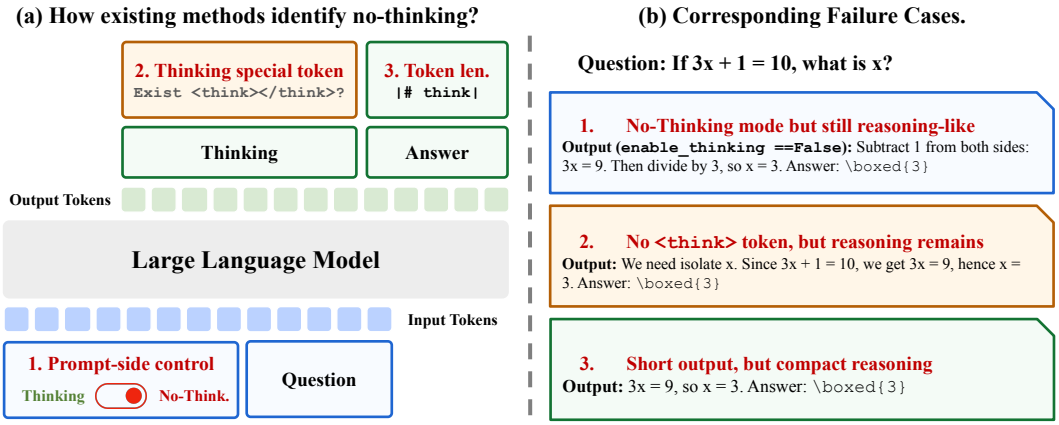

Figure 2: Surface proxies fail to identify response-level no-thinking. (a) Common proxies use prompt-side controls, special reasoning tokens, or output length. **(b)** All three can retain a visible inferential step before the answer, illustrated on the same algebra question.

separates native no-think behavior, reasoning re-elicitation, semantic suppression, and structural prefix forcing. Our human validation supports all three metrics: a manual boundary audit corroborates ETR, the five annotators' relevance judgments align closely with QRel., and their explicit-inference labels achieve 98.8% mean binary agreement with the primary GPT-5.5 judge used to compute EIR. The human annotations also reproduce the main answer-space and intervention trends, further supporting our central conclusions. The EIR results also remain robust when evaluated by an independent judge from a different model family.

Our results highlight four findings. **(i)** Explicit no-think controls are not reliable no-thinking controls: even a frontier LLM such as DeepSeek-V4-Flash (284B) retains visible explicit inference in 99.9% of open-ended responses under its disabled-thinking setting. **(ii)** LLMs exhibit *"thinking inertia"*: visible inference is organized more by the task's answer space than by the switch itself. Under both native and strict no-thinking settings, it remains lowest on Boolean tasks, intermediate on multiple-choice tasks, and highest on open-ended tasks. **(iii)** On open-ended tasks, stronger answer-only instructions reveal what we call a no-thinking–performance trade-off: stronger suppression improves answer-only compliance and reduces visible inference, but can also reduce task accuracy. Specifically, across the five LLMs with fully disabled thinking, strict answer-only prompting raises mean ETR from near zero to roughly 40% while reducing mean accuracy by about 15 percentage points. **(iv)** Answer space matters: for the same question, fact-like and candidate-visible formats make answer-only responses easier to produce than open-generation formats, showing that the pattern is not explained by benchmark difficulty or answer length alone.

In summary, our work makes two contributions. First, we define no-thinking as a measurable response-level behavior and provide a framework for evaluating it. Second, through comprehensive experiments across task families and diverse LLMs, we identify systematic behaviors including thinking inertia, the no-thinking–performance trade-off, and answer-space dependence. Together, these contributions support the broader goal of developing LLMs that, like humans, can adaptively determine when to reason and when to answer directly.

2 Measuring No-Thinking

The failure cases in Figure 2 show why prompt-side heuristics alone are insufficient for studying no-thinking in a controlled way. We therefore ground our analysis in the response-centric framework shown in Figure 3.

2.1 Problem Formulations

Formally, let π denote a language model and Q a task sample. The query is wrapped by an intervention $S(\cdot)$ (*e.g.,* a system prompt such as "please answer <Question> directly") before being passed to the model, which, for the ordinary M1–M5 response order, produces an observable response

$$\pi\big(S(Q)\big) = (T, A) \tag{1}$$

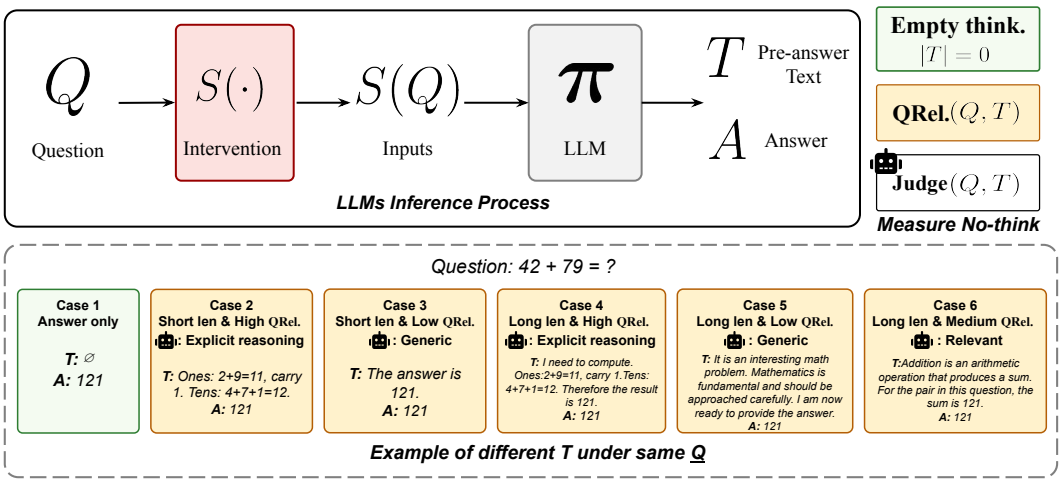

Figure 3: Response-centric metrics for measuring no-thinking. Top: intervention $S(\cdot)$ wraps question Q before model π produces visible text T and answer A. Three levels measure answer-only compliance (ETR), question–text relevance (QRel.), and visible explicit inference (EIR); accuracy evaluates A independently. **Bottom:** six responses to "$42 + 79 =$?" share a final answer but differ in length, relevance, and inference. Appendix C.2 explains their labels; Appendix F.1 gives representative real outputs.

where A is the last parseable final answer and T comprises all readable visible text preceding it. We use (T, A) only as a decomposition of the observable response, without making claims about task-relevant computation in unobserved model states. The strict compliance check additionally scans the complete output, so substantive text after A is not treated as answer-only. Appendix E audits question restatement and answer-first edge cases. Assistant prefilling may alter this order, so we evaluate M6 using its first boxed answer and all residual text (Section 3.3).

Question type and answer space (Q, A). We write each task sample as $Q = (q, c, \mathcal{A})$, where q is the question, c is optional context, and $\mathcal{A}$ is the admissible answer space, with $A \in \mathcal{A}$. For example, $\mathcal{A} = \{\texttt{yes}, \texttt{no}\}$ for Boolean verification, $\mathcal{A} = \{\texttt{A}, \texttt{B}, \texttt{C}, \texttt{D}\}$ for multiple-choice, and $\mathcal{A}$ permits free-form generation for open-ended tasks. This lets us test whether no-thinking behavior changes under different answer interfaces.

Intervention $S(\cdot)$. $S(\cdot)$ denotes an external intervention applied to Q before generation, producing the model input $S(Q)$. We consider two broad classes: **(i)** *question-irrelevant interventions*, which change the instruction, output constraint, or explicit thinking configuration while keeping Q fixed (*e.g.,* prepending "Answer directly without thinking"); and **(ii)** *question-relevant interventions*, which modify the task instance itself, such as its surface form or answer space $\mathcal{A}$ (*e.g.,* converting an open-ended question into a multiple-choice one).

Together, this unified framework allows us to study no-thinking through strict compliance, relevance, and explicit inference rather than through surface proxies alone.

2.2 Three-Level Response Evaluation

Under our formulation, T serves as the primary observable target, while task accuracy separately evaluates the final answer A. We therefore organize the T measurement into three progressively more specific levels (as depicted in Figure 3): strict answer-only compliance, pre-answer relevance, and visible explicit inference. At the response level, Levels 1 and 3 yield binary scores, whereas Level 2 yields a continuous score; their benchmark-level aggregates are defined below.

Level 1: Empty-Thinking Rate. We first ask whether each output is strictly answer-only. For output i, let e_i denote the response-level empty-thinking indicator. For a benchmark with N outputs under a fixed model and intervention, we define the score and **Empty-Thinking Rate (ETR)** as

$$e_i = \mathbf{1}\{|T_i| = 0\}, \qquad \text{ETR} = 100\% \times \frac{1}{N} \sum_{i=1}^{N} e_i. \tag{2}$$

Here, e_i is the response-level score, whereas ETR is its benchmark-level rate. Operationally, $e_i = 1$ only when A_i is parseable and the complete visible output contains no substantive text outside its permitted answer wrapper; otherwise, $e_i = 0$. Thus, empty T_i does not override a malformed final-answer format, and substantive text following A_i also violates strict compliance. ETR gives the cleanest operational limit, but cannot distinguish a generic template, a paraphrase, and a question-specific inference once T_i is non-empty.

Level 2: Question–Pre-answer Relevance. We next ask how strongly the visible pre-answer text T is semantically related to Q. Let ϕ denote an embedding encoder. We use the instruction-aware Qwen3-Embedding-4B [50], encoding Q as a query and T as a candidate passage (Appendix D.1). After ℓ_2-normalizing each embedding, i.e., dividing each vector by its Euclidean norm, we define the response-level **Question–Pre-answer Relevance (QRel.)** score as

$$\mathrm{QRel.}(Q_i, T_i) = \begin{cases} 0, & |T_i| = 0, \\ \cos\big(\phi(Q_i), \phi(T_i)\big), & |T_i| > 0. \end{cases} \tag{3}$$

The first branch assigns empty T_i a score of zero, while the second directly uses the cosine similarity for non-empty T_i. For each model–benchmark–intervention setting, the reported QRel. is the arithmetic mean of these response-level scores over all N outputs, including the zeros assigned to empty T_i. The instruction-aware objective makes relevance to answering Q_i, rather than surface overlap alone, the target relation.

However, QRel. measures relevance rather than inference: a paraphrase or a relevant answer assertion can receive a high score without containing an inferential step. Appendix E audits whether near-verbatim or semantic restatement drives the aggregate results, while Level 3 keeps such text outside the explicit-inference category.

Level 3: Explicit Inference Rate. Level 1 checks a structural rule and Level 2 measures functional relevance; neither directly separates reasoning from relevant non-inferential text. A blinded judge receives only (Q_i, T_i) and classifies non-empty T_i using the five mutually exclusive labels in Table C.2. Appendix C.2 gives the full protocol.

A relevant restatement or answer assertion alone is therefore not reasoning, whereas one concrete inferential step can qualify even when it is short, shallow, or ultimately incorrect. For the metric, let $\mathrm{Judge}(Q_i, T_i)$ denote the binary score induced by the five-way label returned when the blinded judge receives (Q_i, T_i): it is 1 only when the judge selects *Explicit reasoning*, and 0 for the other four labels. Empty T_i is not sent to the judge and is assigned $\mathrm{Judge}(Q_i, T_i) = 0$ by default. For a benchmark with N outputs under a fixed model and intervention, we define the **Explicit Inference Rate (EIR)** as

$$\mathrm{EIR} = 100\% \times \frac{1}{N} \sum_{i=1}^{N} \mathrm{Judge}(Q_i, T_i). \tag{4}$$

Thus, $\mathrm{Judge}(Q_i, T_i)$ is the response-level binary score, whereas EIR is its benchmark-level statistics.

Our blinded human study validates the metrics and their main experimental trends; Appendix D.3 reports the protocol and results.

2.3 Thinking Behavior Control

To cover these setups, we design a six-mode spectrum (Table 1) that organizes modes along a single axis of *no-think intensity*, ranging from $(-)$ for thinking-encouraging modes and (0) for the native no-thinking mode to $(+++)$ for the strongest suppression. All modes share the same task stem and final-answer request; only the thinking switch and the question-irrelevant intervention change.

Thinking variants (THINK-ON, COT). THINK-ON (M1) keeps the official thinking interface enabled, while COT (M3) disables it but adds the instruction *"Think step by step"* to re-elicit reasoning in the ordinary answer channel. Both elicit thinking rather than suppress it, and serve as opposite-direction references for the no-think space (gray rows in Table 1).

No-thinking variants (THINK-OFF, SOFT, STRICT, PREFIX). THINK-OFF (M2) disables `enable_thinking` without any additional output-side control, serving as the natural baseline against which all suppression interventions are measured. SOFT (M4) and STRICT (M5) keep `enable_thinking=False` and add increasingly strong natural-language constraints: SOFT allows

Table 1: Six-mode question-irrelevant prompt intervention spectrum. Modes shown in gray (THINK-ON, COT) still elicit thinking behavior and serve as thinking-side references; the remaining modes (THINK-OFF, SOFT, STRICT, PREFIX) form the no-think intervention space, ranging from the native no-thinking baseline to increasingly strong suppression.

#	Mode	Purpose (no-think intensity)	Thinking enabled	Question-Irrelevant Intervention $S(\cdot)$
M1	THINK-ON	Native `thinking-on` mode $(-)$	✓	None.
M2	THINK-OFF	Native `no-thinking` mode (0)	✗	None.
M3	COT	Explicit reasoning $(-)$	✗	"Think step by step."
M4	SOFT	Soft regularization $(+)$	✗	"A very short lead-in is allowed, but do not explain."
M5	STRICT	Strict regularization $(++)$	✗	"Give the final answer directly without any reasoning."
M6	PREFIX	Prefix forcing $(+++)$	✗	"The answer is `\boxed{`"

a short lead-in but forbids explanation, while STRICT asks for the answer directly without any reasoning. PREFIX (M6) is qualitatively different. It supplies the answer template "The answer is `\boxed{`" as the unfinished assistant response and generates its continuation.

3 Experiments

We organize our experiments around three questions. Starting from the model's native thinking controls, we ask **Q1.** whether enabling or disabling the thinking mode actually suppresses visible pre-answer content. We then focus on the native no-thinking mode and ask **Q2.** how prompt interventions of varying strength affect no-thinking behavior and its relationship with task performance. Building on these analyses, we conduct finer-grained controlled comparisons and ask **Q3.** how no-thinking behavior varies across benchmark domains and question types, and how much of this variation can be attributed to answer space.

3.1 Settings

LLM baselines. We select LLMs that expose explicit thinking-mode controls, so that no-thinking behavior can be tested by intervention rather than inferred from output length or format. Table 2 spans the Qwen3 family (4B, 32B, and 235B), DeepSeek-V4-Flash, and proprietary frontier models from OpenAI and Google, covering both model families and scales. The three Qwen3 sizes support a within-family scaling analysis, while the local 4B and 32B checkpoints provide transparent paired native on/off controls for detailed ablations. All six models are evaluated under M1–M5 from Section 2.3; assistant-prefill M6 covers the three Qwen sizes and DeepSeek.

Table 2: LLM baselines.

Model	Size	Thinking mode
`Qwen3-4B`	4B	On | Off
`Qwen3-32B`	32B	On | Off
`Qwen3-235B-A22B`	235B	On | Off
`DeepSeek-V4-Flash`	284B	High | Off
`GPT-5.4`	N/A	High | Off
`Gemini-3-Flash`	N/A	High | Minimal

Benchmarks. We use three answer spaces, each with an easier and a harder benchmark: Boolean verification (BoolQ [6] and StrategyQA [10]), multiple-choice answering (MMLU [14] and MMLU-Pro [40]), and open-ended answering (GSM8K [7] and MATH [15]). See Appendix B.1 for details.

Metrics. For each model–benchmark–intervention setting, we report **Task Accuracy (Acc.)** and the three no-thinking metrics defined in Section 2.2: the **Empty-Thinking Rate (ETR)**, the **Question–Pre-answer Relevance (QRel.)**, and the **Explicit Inference Rate (EIR)**. Metrics include outputs with both empty and non-empty T. Empty T contributes zero to QRel. and EIR. M1–M5 use the full output set as the denominator. M6 follows the residual-text rule in Section 3.3.

We use Qwen3-Embedding-4B for QRel. and GPT-5.5 for EIR; implementation details and cross-encoder, cross-judge, and human validation appear in Appendix D.

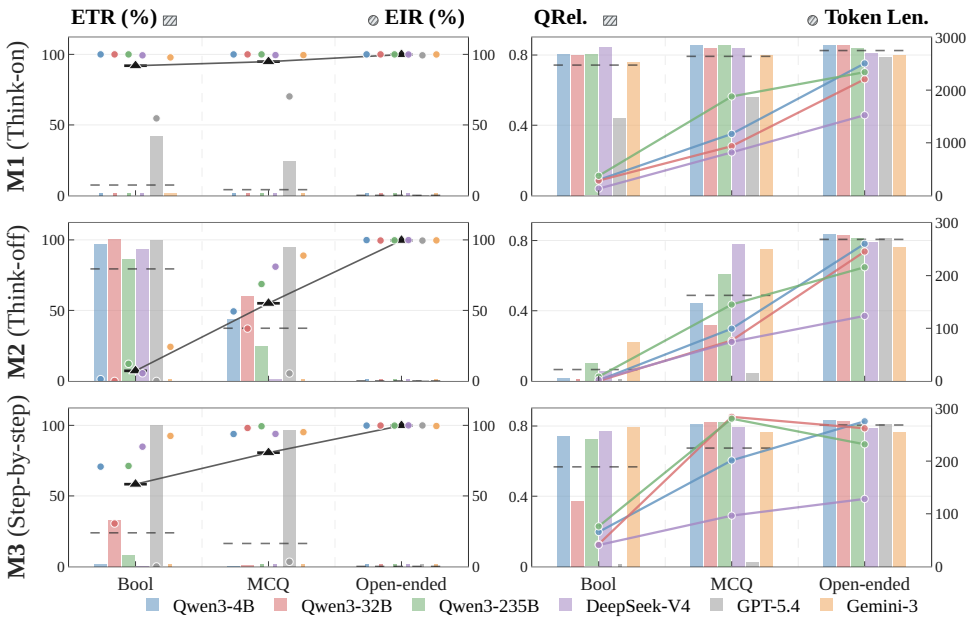

Figure 4: Think-on (M1), Think-off (M2), and Step-by-step (M3) across six models and three answer-space families. Rows show modes; the columns pair ETR bars with EIR points (left) and QRel. bars with visible pre-answer length for the four models with complete length traces (right). Gray dashed segments mark six-model means for ETR and QRel.; black triangles connected by dark lines mark the mean EIR; colored marks follow the model legend.

3.2 Native Think-on vs. Think-off

Figure 4 compares M1 (THINK-ON), M2 (THINK-OFF), and M3 (STEP-BY-STEP) across six models and three answer-space families (Bool, MCQ, and Open-ended). The left panels pair ETR with EIR, while the right panels pair QRel. with visible pre-answer length $|T|$.

Native Think-off becomes less effective as the answer space opens. We first test whether the native thinking-mode control reliably suppresses visible pre-answer content. M1 (THINK-ON) serves as the native thinking-on reference: for most model–task pairs, ETR is near zero, while QRel. and EIR are high. Under M2 (THINK-OFF), most models reach roughly 80–100% ETR on Bool, with sharply reduced QRel. and EIR. MCQ forms an intermediate but heterogeneous regime: ETR ranges from near zero to 95% across models, while the six-model mean QRel. and EIR values fall between Bool and Open-ended. On Open-ended tasks, ETR collapses to nearly zero everywhere, while QRel. remains close to its M1 level and EIR remains high. Across the three metrics, responses become less likely to remain answer-only and more likely to retain question-related, explicitly inferential pre-answer text as the answer space opens. This observable progression is consistent with greater computational demands: constructing an answer in an open space may require more task-relevant intermediate computation than selecting from narrow or candidate-visible alternatives.

We use "thinking inertia" to describe how visible explicit inference persists under no-thinking controls and becomes more common as the answer space opens. The native no-thinking setting therefore does not guarantee answer-only responses: ETR is high mainly on Bool, while the six-model mean EIR is lowest on Bool, intermediate on MCQ, and highest on Open-ended tasks.

Step-by-step prompts can recover visible reasoning. M3 (STEP-BY-STEP) provides the reverse intervention: it asks whether an ordinary step-by-step prompt can re-elicit visible reasoning while the native thinking switch remains off. Under M3, QRel. and EIR often move back toward their M1 levels, including on Bool for many models; the four available length traces show the same recovery. This indicates that an ordinary step-by-step instruction can re-elicit visible explicit inference even while the native thinking switch remains off. Nevertheless, the six-model mean EIR preserves the

same Bool < MCQ < Open-ended staircase under M3, showing that the answer-space dependence characteristic of *thinking inertia* persists under explicit step-by-step prompting. GPT-5.4 is the main exception to this recovery pattern, maintaining high answer-only compliance and correspondingly low EIR on both Bool and MCQ. Thus, the reverse intervention changes the overall level of visible inference but not its answer-space ordering.

3.3 Performance across No-Think Variants

We next ask how no-think interventions affect response behavior and task performance.

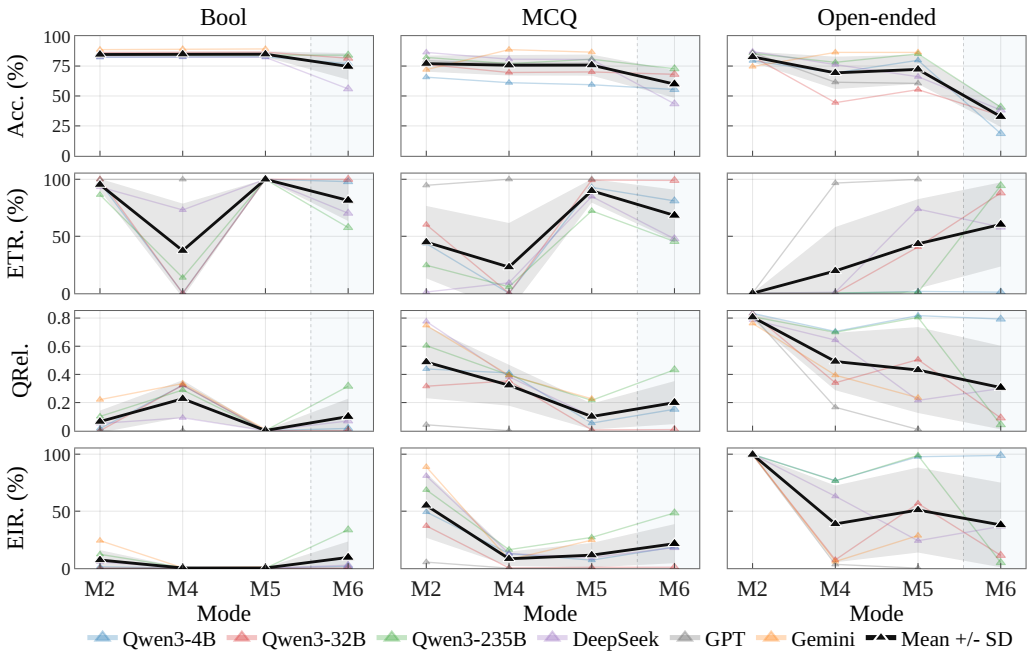

Figure 5: Natural-language suppression and assistant prefilling across answer spaces. Columns show Bool, MCQ, and Open-ended tasks; rows show Acc., ETR, QRel., and EIR. M2/M4/M5 use pre-answer text; M6 uses residual text. Blue shading marks M6's distinct assistant-prefill control. Colored marks show models; black marks and gray bands give their mean and one standard deviation. Appendix E.2 gives evaluation and aggregation details.

Open-ended tasks expose a trade-off between answer-only compliance and accuracy. Across M2, M4, and M5 in Figure 5, Bool accuracy remains nearly flat at around 85%, MCQ accuracy varies modestly, and Open-ended accuracy varies most across modes. On Open-ended tasks, moving from M2 to M5 raises mean ETR from near zero to roughly 40% and reduces mean accuracy by about 15 percentage points across models with a disabled-thinking condition. Over the complete evaluation corpus for these two modes, EIR likewise decreases from 99.69% to 62.00% (Appendix Table D.3). Bool and MCQ show smaller accuracy changes, even as EIR falls from 5.15% to 0.12% on Bool and from 45.57% to 5.39% on MCQ. Thus, stronger natural-language suppression reduces visible explicit inference in every answer space, but Open-ended tasks retain substantial inference and show the clearest accuracy cost.

Evaluating assistant-prefilled responses. M6 starts the assistant response with `The answer is \boxed{` and lets the model continue from that unfinished prefix. We find that M6 responses do not consistently follow the pre-answer (T, A) structure observed under M1–M5. Literal answer-first responses with a text suffix are rare among boxed M1–M5 outputs (Appendix Table E.1). In M6, alongside answer-only responses, we observe (A, T) and (A, T, A'), as well as inference within the initial answer box, schematically `\boxed{T,A}`. The initial box may also be empty or unclosed (Table E.2). To evaluate these responses, **Acc.** scores the answer extracted from the *first prefilled box*, using the same benchmark comparator as M1–M5; a later correction does not replace that

answer. For the behavioral metrics, we remove the first complete `The answer is \boxed{xxx}` declaration, including its entire box, and define T as *all remaining returned readable text*: subsequent explanations, repeated or revised answers, and any separately returned reasoning text. A response counts as answer-only for **ETR** when it contains a parseable first boxed answer and no substantive residual text. **QRel.** and **EIR** use this residual T with the same encoder, judge, and rubric as M1–M5; empty T receives zero for both metrics. These residual-text scores exclude any inference inside the removed box. Appendix E.2 gives protocol details; Appendix J reports user-message-prefix results.

Accuracy is lower under M6 prefilling, with losses increasing from Bool to MCQ to Open-ended tasks. For the three Qwen sizes and DeepSeek, mean accuracy falls from 83.85% under M2 to 74.66% under M6 on Bool, from 77.68% to 59.93% on MCQ, and from 83.78% to 32.76% on Open-ended tasks: decreases of 9.19, 17.75, and 51.01 percentage points, respectively. For each of the four models, the decrease from M2 is smallest on Bool and largest on Open-ended tasks. Compared with M5, mean accuracy decreases by 9.27, 12.75, and 38.84 points, respectively, although individual models do not all follow this ordering. The three behavioral metrics show broadly similar changes from M5 across models on Bool and MCQ, but diverge on Open-ended tasks. On Bool and MCQ, M6 generally lowers answer-only compliance and increases residual relevance and inference, while Qwen3-32B remains almost entirely answer-only. On Open-ended tasks, answer-only compliance rises substantially from M5 for Qwen3-32B and Qwen3-235B, falls for DeepSeek, and remains near zero for Qwen3-4B. Qwen3-4B combines persistent residual inference with low first-box accuracy, whereas Qwen3-235B is usually answer-only but still suffers a substantial accuracy loss.

On open-ended tasks, no-thinking comes at a cost. This pattern is consistent with greater computational demands in Open-ended tasks: compared with native THINK-OFF, strict answer-only prompting lowers EIR but leaves substantial visible explicit inference, while higher answer-only compliance accompanies lower accuracy. For M6, first-answer accuracy losses grow from Bool through MCQ to Open-ended tasks, where answer-only compliance and residual inference vary sharply across models.

Within MMLU, no-thinking behavior also varies across disciplines under the same answer interface; Appendix I.1 reports the comparison.

3.4 Performance under answer-space rewrites

The preceding experiments vary thinking controls across task families. We now keep the underlying math questions fixed and change only the answer interface, allowing a more controlled comparison.

Changing only the answer interface reshapes no-thinking behavior. Figure 6 rewrites the matched numeric-answer subset of MMLU/MMLU-Pro for Qwen3-32B under M2 from its original MCQ form into Boolean verification or open-ended generation. The Boolean rewrite leaves accuracy nearly unchanged, from 77.8% to 78.9%, but raises ETR from 10.5% to 62.5%, lowers QRel. from 0.709 to 0.314, and lowers EIR from 86.95% to 36.90%. By contrast, the open-ended rewrite reduces accuracy to 52.6%, drives ETR to 0%, and raises QRel. to 0.810 and EIR to 90.14%. Thus, changing only how the answer is requested substantially changes response-level no-thinking behavior, even for the same underlying questions. A qualitative example tracing all six modes on one rewritten item appears in Appendix Figure F.1.

Answer presentation shapes direct answering. This controlled comparison clarifies what the earlier answer-space trend reflects. Boolean verification gives the model a proposed answer to check, while open-ended generation requires the model to produce the value without such support. The original MCQ form lies between these cases: it provides options, but the model may still need visible intermediate steps to compare or compute them. The same ordering appears in the three-level view: Boolean verification has the highest ETR and the lowest QRel. and EIR, whereas open-ended generation has no answer-only responses and retains both relevant pre-answer text and explicit inference. Thus, the answer-space pattern in Sections 3.2 and 3.3 appears both across benchmark families and within the same questions when the answer interface changes.

Even for the same question, the answer space shapes whether an LLM answers directly. Candidate-visible formats make answer-only responses easier to produce than open-ended generation, so the broader pattern cannot be reduced to differences between benchmark families.

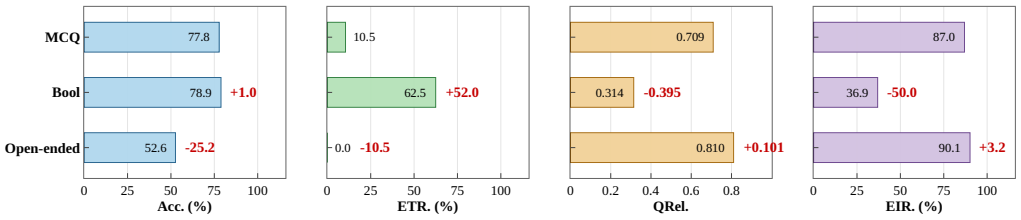

Figure 6: Changing only the answer space reshapes no-thinking behavior on the same questions. We rewrite matched numeric-answer items from MMLU and MMLU-Pro for Qwen3-32B under M2 from MCQ into Boolean or open-ended forms. The four panels report Acc., ETR, QRel., and EIR for these variants; red values give the change from the MCQ form, computed before rounding.

4 Related Work

Efficient and Adaptive Reasoning. Test-time computation improves reasoning [31, 3], motivating selective allocation. Length-conditioned training [1], token-budget prompting [11], budget forcing [23], and concise self-training [24] control visible reasoning; CoT distillation [18] and terse drafts [45] compress intermediate steps. Since useful reasoning length depends on task difficulty and model capability [4, 43, 49], adaptive methods learn when extended reasoning is needed [47, 16]. Selective reasoning has also been studied in vision-language models [36, 21]. Recent work also studies whether concealed reasoning can be reconstructed or exposed. Trace Inversion synthesizes distillation-useful reasoning traces from black-box inputs, answers, and optional summaries without access to hidden CoT [48]. Encrypted reasoning blocks exposed by proprietary APIs can also be transferred to weaker models within the same provider and decoded into plaintext [27]. Prompt repetition offers an input-side technique for non-reasoning models, evaluated through final-answer accuracy, length, and latency [22]. We evaluate native controls, natural-language suppression, and assistant prefilling across answer spaces.

Instruction Following. Instruction tuning [41, 30, 5] and human- and AI-feedback alignment [26, 2] train models to follow open-ended directives, while self-generated and synthetic instruction data [39, 33] broaden the instruction distribution. IFEval [51], FollowBench [17], and complex-instruction benchmarks [13, 28] evaluate output-format and content constraints. We treat "produce no visible reasoning" as an output constraint and separately measure answer-only compliance, question–text relevance, and visible explicit inference, alongside task accuracy. This separation lets us examine whether suppressing visible reasoning preserves task performance. Appendix L discusses these approaches in more detail.

5 Conclusion

We argue that "no-thinking" in LLMs should be evaluated as a response-level property rather than inferred from surface proxies such as missing reasoning tags, no-think settings, or short generations. We measure strict answer-only compliance, visible-text relevance, and visible explicit inference at three levels, using pre-answer text T and final answer A for M1–M5, and first-box A with residual T for M6. Task accuracy evaluates A independently. Across local Qwen3 models, scaled variants, and frontier APIs, we find that explicit no-think controls do not reliably remove visible explicit inference; models exhibit "*thinking inertia*" organized by answer space; stronger answer-only constraints reveal a *no-thinking–performance trade-off* between compliance and accuracy; and compressibility is governed by the structural support of the question itself, not by question difficulty or model capability alone. These findings suggest that building LLMs that can stop thinking on demand requires more than hiding reasoning traces or enforcing shorter outputs: it requires evaluating when direct answering is actually supported by the task.

Acknowledgments and Disclosure of Funding

This work was supported by the UKRI Turing AI Fellowship (EP/W002981/1). This work was also supported by Tsinghua University Initiative Scientific Research Program (Student Academic Research Advancement Program, 20267020002). The authors declare no competing interests.

References

[1] Pranjal Aggarwal and Sean Welleck. L1: Controlling how long a reasoning model thinks with reinforcement learning. *arXiv preprint arXiv:2503.04697*, 2025.

[2] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[3] Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.

[4] Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do NOT think that much for 2+3=? on the overthinking of o1-like LLMs. *arXiv preprint arXiv:2412.21187*, 2024.

[5] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25:1–53, 2024.

[6] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019.

[7] 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. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[8] DeepSeek-AI et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[9] Gongfan Fang, Xinyin Ma, and Xinchao Wang. Thinkless: LLM learns when to think. *arXiv preprint arXiv:2505.13379*, 2025.

[10] Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 2021.

[11] Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware LLM reasoning. *arXiv preprint arXiv:2412.18547*, 2024.

[12] Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. In *Conference on Language Modeling*, 2025.

[13] Qianyu He, Jie Zeng, Wenhao Huang, Lina Chen, Jin Xiao, Qianxi He, Xunzhe Zhou, Jiaqing Liang, and Yanghua Xiao. Can large language models understand real-world complex instructions? *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

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

[15] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. *arXiv preprint arXiv:2103.03874*, 2021.

[16] Lingjie Jiang, Xun Wu, Shaohan Huang, Qingxiu Dong, Zewen Chi, Li Dong, Xingxing Zhang, Tengchao Lv, Lei Cui, and Furu Wei. Think only when you need with large hybrid-reasoning models. *arXiv preprint arXiv:2505.14631*, 2025.

[17] Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. FollowBench: A multi-level fine-grained constraints following benchmark for large language models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

[18] Yu Kang, Xianghui Sun, Liangyu Chen, and Wei Zou. C3oT: Generating shorter chain-of-thought without compromising effectiveness. *arXiv preprint arXiv:2412.11664*, 2024.

[19] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*, 2022.

[20] Tamera Lanham et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

[21] Dianqiao Lei and Lianlei Shan. Think less, act early: Reinforced latent reasoning with early exit in vision-language-action models. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, pages 65421–65436. PMLR, 2026.

[22] Yaniv Leviathan, Matan Kalman, and Yossi Matias. Prompt repetition improves non-reasoning LLMs. *arXiv preprint arXiv:2512.14982*, 2025.

[23] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

[24] Tergel Munkhbat, Namgyu Ho, Seo Hyun Kim, Yongjin Yang, Yujin Kim, and Se-Young Yun. Self-training elicits concise reasoning in large language models. *arXiv preprint arXiv:2502.20122*, 2025.

[25] OpenAI. Reasoning models. https://platform.openai.com/docs/guides/reasoning, 2025. OpenAI API documentation.

[26] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[27] Alexander Panfilov, David Schmotz, Ilia Shumailov, Luca Beurer-Kellner, Joachim Schaeffer, Ameya Prabhu, Jonas Geiping, and Maksym Andriushchenko. Stealing reasoning traces from proprietary LLM APIs. *arXiv preprint arXiv:2608.09867*, 2026.

[28] Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. InfoBench: Evaluating instruction following ability in large language models. *Findings of the Association for Computational Linguistics: ACL 2024*, 2024.

[29] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019.

[30] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, et al. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations (ICLR)*, 2022.

[31] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *International Conference on Learning Representations*, 2025.

[32] Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Hanjie Chen, and Xia Hu. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*, 2025.

[33] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford Alpaca: An instruction-following LLaMA model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[34] Songjun Tu, Jiahao Lin, Qichao Zhang, Xiangyu Tian, Linjing Li, Xiangyuan Lan, and Dongbin Zhao. Learning when to think: Shaping adaptive reasoning in r1-style models via multi-stage RL. *arXiv preprint arXiv:2505.10832*, 2025.

[35] Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems*, 2023.

[36] Jiaqi Wang, Kevin Qinghong Lin, James Cheng, and Mike Zheng Shou. Think or not? selective reasoning via reinforcement learning for vision-language models. In *Advances in Neural Information Processing Systems*, volume 38, pages 123247–123285, 2025.

[37] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

[38] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *arXiv preprint arXiv:2002.10957*, 2020.

[39] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

[40] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*, 2024.

[41] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations (ICLR)*, 2022.

[42] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022.

[43] Yuyang Wu, Yifei Wang, Ziyu Ye, Tianqi Du, Stefanie Jegelka, and Yisen Wang. When more is less: Understanding chain-of-thought length in LLMs. *arXiv preprint arXiv:2502.07266*, 2025.

[44] Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-Pack: Packed resources for general chinese embeddings. *arXiv preprint arXiv:2309.07597*, 2023.

[45] Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. Chain of draft: Thinking faster by writing less. *arXiv preprint arXiv:2502.18600*, 2025.

[46] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin

Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[47] Jiajie Zhang, Nianyi Lin, Lei Hou, Ling Feng, and Juanzi Li. AdaptThink: Reasoning models can learn when to think. *arXiv preprint arXiv:2505.13417*, 2025.

[48] Tingwei Zhang, John X. Morris, and Vitaly Shmatikov. How to steal reasoning without reasoning traces. *arXiv preprint arXiv:2603.07267*, 2026.

[49] Wenyuan Zhang, Shuaiyi Nie, Xinghua Zhang, Zefeng Zhang, and Tingwen Liu. Exploring the system 1 thinking capability of large reasoning models. *arXiv preprint arXiv:2504.10368*, 2025.

[50] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

[51] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

A Appendix Overview

We report the protocols, robustness checks, aggregate results, question-structure controls, and representative outputs underlying the main-text results. Appendix E.2 gives the assistant-prefill M6 protocol and results. We denote the separate user-message-prefix condition by $\mathrm{M6}^{\mathrm{u}}$; its implementation and results are described in Appendix J.

B Experimental Details

B.1 Dataset Sizes

To evaluate no-thinking across answer spaces, we use three answer-space families, each with an easier and a harder benchmark. *(i) Boolean verification.* Each sample requires a yes/no judgment. We use BoolQ [6] and StrategyQA [10]. *(ii) Multiple-choice answering.* Each sample provides a fixed set of candidate options, and the model returns one option label. We use MMLU [14] and MMLU-Pro [40]. *(iii) Open-ended answering.* Each sample requires the model to generate the answer without a provided candidate set. We use GSM8K [7] and MATH [15]. These benchmarks cover answer spaces ranging from binary judgments through fixed-choice selection to open-ended generation, while using the same response-level evaluation metrics.

Table B.1 lists the full benchmark splits. The number of evaluated items depends on the model and intervention.

Table B.1: Full benchmark split sizes, used by the local Qwen3-4B/32B runs under M1–M5 and $\mathrm{M6}^{\mathrm{u}}$.

Dataset	Split	Size used
BoolQ	Validation	3,270
StrategyQA	Test	687
MMLU	Test	14,042
MMLU-Pro	Test	12,032
GSM8K	Test	1,319
MATH	Test	5,000

For M1–M5 and $\mathrm{M6}^{\mathrm{u}}$, the local Qwen3-4B/32B runs use the full benchmark splits. Each of the four API models uses 1,000 sampled items from BoolQ, MMLU, MMLU-Pro, GSM8K, and MATH, together with all 687 StrategyQA items, giving 5,687 outputs per mode and 34,122 outputs across these six conditions. M6 uses 5,687 items per model for all four models; its item manifest and family counts are given in Appendix E.2.

B.2 Models and Inference Settings

For M1–M5 and $\mathrm{M6}^{\mathrm{u}}$, we use the models' public thinking-control interfaces. Appendix E.2 describes the assistant-prefill interfaces and deployment settings for M6.

Table B.2: Model interfaces and decoding settings. API controls and "all modes" budgets refer to M1–M5 and $\mathrm{M6}^{\mathrm{u}}$; M6 deployment settings are given in Appendix E.2.

Model group	Thinking control	Decoding and evaluation scope
Qwen3-4B / Qwen3-32B	Local Qwen chat-template switch: `enable_thinking=true` for Mode 1 and `false` for M2–M5 and $\mathrm{M6}^{\mathrm{u}}$.	Temperature 0.0. Response budgets: 4,096 tokens for M1; 1,024 for M2–M5 and $\mathrm{M6}^{\mathrm{u}}$; 8,192 for M6.
Qwen3-235B-A22B	API reasoning control: high effort for Mode 1 and no-reasoning setting for M2–M5 and $\mathrm{M6}^{\mathrm{u}}$.	Temperature 0.0. Max tokens: 8,192 for all modes.

Continued on next page

Table B.2 (continued)

Model group	Thinking control	Decoding and evaluation scope
DeepSeek-V4-Flash	Provider thinking interface: high effort for Mode 1 and disabled for M2–M5 and $M6^{u}$.	Temperature 0.0. Max tokens: 8,192 for all modes.
GPT-5.4	Reasoning effort: high for Mode 1 and none for M2–M5 and $M6^{u}$.	Temperature 0.0. Max tokens: 8,192 for all modes.
Gemini-3-Flash-Preview	Reasoning effort: high for Mode 1 and minimal for M2–M5 and $M6^{u}$.	Temperature 0.0. Max tokens: 8,192 for all modes. The minimal setting is treated as near-off rather than strict off.

We set temperature to zero wherever the endpoint supports it. The M1–M5 and $M6^{u}$ API runs use an 8,192-token response budget. Local Qwen budgets vary by condition, as listed in Table B.2; M6 uses 8,192 tokens for all four models (Appendix E.2). For Qwen3-235B-A22B, DeepSeek-V4-Flash, and GPT-5.4, truncation affects at most 0.35% of outputs in any M2/M5 answer-space cell under the 8,192-token budget. For M1–M5 and $M6^{u}$, non-text provider reasoning signals enter the strict answer-only decision, and returned readable reasoning contributes to the visible text.

C Evaluation Protocol

Protocol summary. M1–M5 and $M6^{u}$ use the response-level setup from Section 2.2, with pre-answer text T and final answer A. Non-text reasoning signals enter the answer-only check, and returned readable reasoning fields contribute to T. For M6, A is the first boxed answer and T contains the residual text from all returned readable fields (Appendix E.2).

Table C.1: Scope of the M1–M5 and $M6^{u}$ result blocks and question-structure controls. Assistant-prefill M6 results are reported in Appendix E.2.

Result block	Models and task scope	Appendix evidence supported
Local Qwen spectrum	Qwen3-4B and Qwen3-32B on the six-benchmark suite: BoolQ, StrategyQA, MMLU, MMLU-Pro, GSM8K, and MATH, under M1–M5 and $M6^{u}$.	Global local-model summary and Qwen scaling evidence in Figure G.1 and Table H.1.
Frontier model results	Qwen3-235B-A22B, DeepSeek-V4-Flash, GPT-5.4, and Gemini-3-Flash-Preview on the same six-benchmark suite, using each provider's public thinking-control interface.	Frontier native no-think and suppression/forcing results in Figures H.1 and G.2, plus the Qwen table in Table H.1.
MMLU domain analysis	Qwen3-4B, Qwen3-32B, DeepSeek-V4-Flash, and Qwen3-235B-A22B, grouped by MMLU supercategory and, where applicable, subject. Acc./ETR use the full split; QRel./EIR use the full split for the local models and 1,000 items for each API model.	Domain and subject-level analyses in Figure H.2, Table I.2, and Table I.3.
Surface and candidate-visibility interventions	Qwen3-4B and Qwen3-32B on MMLU variants with choice shuffling, numeric labels, and boolean verification with a proposed answer shown.	Question-interface controls in Tables I.4 and I.5.
Numeric-source open-generation intervention	Qwen3-4B and Qwen3-32B on matched numeric-source items, comparing MCQ, boolean verification, and open numeric generation.	Candidate-removal stress test in Table I.6.

Run aggregation. Dataset-level aggregates use M1–M5 as defined in Section 2.3 and $M6^{u}$ for the user-message-prefix condition. The grouped metric tables pool items within each answer family. In Figure 5, local Open-ended accuracy for M2, M4, and M5 is the mean of the two benchmark

accuracies. M6 pools items within each model and family, then averages the model rates with equal weight. The frontier M1–M5 and $\mathrm{M6}^{\mathrm{u}}$ aggregates use the 8,192-token runs with seed 20260502. For these pre-answer conditions, QRel. uses the complete visible text before A, including readable reasoning fields and ordinary pre-answer content. GPT-5.5 scores all outputs in the M1–M5 plus $\mathrm{M6}^{\mathrm{u}}$ corpus. The domain analysis groups the MMLU portion of these outputs by subject area. These EIR scores appear in the global, frontier, scaling, and domain analyses; the question-structure controls examine relevance and consistency.

Visible pre-answer text T. T contains all readable visible content before the final answer A, whether it appears in a special reasoning block, the ordinary answer channel, or a short lead-in. It may include an explicit derivation, a compact reasoning step, a paraphrase, a relevant assertion, or a generic template such as "The answer is". These cases motivate evaluating the content of T rather than equating every non-empty T with reasoning.

C.1 Surface proxies and response-level evaluation

No-thinking in LLMs does not have a concrete definition. Three surface properties are relevant here, as summarized on the left side of Figure 2: (**i**) *prompt-side control*, where the model is run in a no-think setting or is not prompted to think step by step [22]; (**ii**) *visible-format control*, where no-thinking is inferred from the absence of an explicit `<think>` block or a separate reasoning trace [46, 8]; and (**iii**) *length-based control*, where generations are shortened through length control or chain-of-thought compression [1, 18]. These proxies are convenient and easy to monitor, but they characterize the response by its surface prompting condition, visible format, or length, rather than by the actual content of the LLM response (such as the question-conditioned information it actually exposes).

Each of these formulations, however, has a simple failure case, illustrated on the right side of Figure 2. (**i**) A no-think instruction can still yield explanatory, question-dependent text before the final answer; disabling a thinking mode therefore does not by itself remove visible reasoning. (**ii**) Removing explicit `<think>` tags only removes a particular format; reasoning may still appear as ordinary language in the answer channel. (**iii**) Making an output short does not make it no-thinking, since a compact response can still encode the essential intermediate step needed to answer the question. Thus, the problem is not that these proxies are useless, but that none of them defines the behavioral object we want to study. We therefore operationalize no-thinking at the response level and progressively distinguish strict answer-only compliance, question relevance, and visible explicit inference. This observable definition does not make claims about latent computation inside the model, because extending the analysis to latent reasoning would substantially broaden the scope and make the boundary of no-thinking difficult to define clearly.

C.2 Blinded judge instruction and rubric

The judge receives only (Q_i, T_i); model identity, intervention mode, final answer A_i, correctness, and QRel. are hidden. It returns one of the five mutually exclusive labels in Table C.2, together with an exact supporting span from T_i. The span records the evidence for the selected label and makes the decision auditable, but does not enter the metric. Table C.2 presents the judge instruction and rubric together.

The six illustrative responses in Figure 3 answer the same question, "$42 + 79 =$?", while varying in length, relevance, and inferential content. Case 1 is answer-only (empty T), whereas Cases 2–6 use the non-empty-T labels from the rubric: Cases 2 and 4 are explicit reasoning, Cases 3 and 5 are generic/off-topic, and Case 6 is relevant non-inferential. The rubric additionally defines paraphrase-only and unclear; representative real outputs appear in Appendix F.1.

Table C.2: Blinded LLM-judge instruction and rubric for non-empty pre-answer text T. Empty T is handled separately and counts as no visible explicit inference.

Prompt for the LLM Judge

Judge whether the `PRE-ANSWER TEXT` contains reasoning that addresses the `QUESTION`, select the corresponding rubric label, and record an exact supporting span.

Index	Label	Criterion
1	Generic/off-topic	Generic lead-in, formatting text, or off-topic content with no question-specific inference.
2	Paraphrase-only	Restates Q or supplied facts without connecting them through an inference step.
3	Relevant non-inferential	Gives a relevant fact, definition, description, or answer assertion without an inference step.
4	**Explicit reasoning**	Contains deduction, computation, comparison, elimination, rule application, evidence-to-conclusion, or causality addressing Q.
5	Unclear	Cannot be classified reliably from the visible text.

D Validation of the Three-Level Evaluation

For QRel., we use the instruction-aware `Qwen3-Embedding-4B` [50] with last-token pooling and a maximum sequence length of 32,768, following the query–passage setup in Section 2.2. **For EIR**, we use `GPT-5.5` at temperature 0 as the primary blinded judge, following the input and blinding protocol in Section 2.2. This appendix tests Level 2 with alternative widely used encoders and replicates Level 3 with Claude Opus 4.8 across all six models and six benchmarks under M1–M5 and $\mathrm{M6}^{\mathrm{u}}$.

The full-corpus comparisons validate the pre-answer protocol for M1–M5 and $\mathrm{M6}^{\mathrm{u}}$.

D.1 Level 2: Encoder Robustness

We instantiate ϕ with the instruction-aware Qwen3-Embedding-4B [50], because its query instruction lets us specify the intended relation between Q and T: Q is encoded as the query under the instruction "Given a question, retrieve pre-answer text that contains reasoning relevant to answering the question," while T is encoded as the candidate passage. Here, $\phi(Q)$ denotes the resulting instruction-conditioned query embedding and $\phi(T)$ the candidate-passage embedding produced by the same encoder.

Qwen3-Embedding-4B is our primary Level 2 encoder because its instruction-aware retrieval interface lets the query specify the intended relation between Q and T. We test whether the relevance pattern depends on this representation by rescoring the complete corpus with Qwen3-Embedding-4B and all-MiniLM-L6-v2; Table D.1 reports the central M2/M5 comparison.

Table D.1: Full-corpus Level 2 comparison. Mean relevance scores use all outputs, with empty T assigned zero. Both encoders were applied to outputs from all six models, six benchmarks, and the M1–M5 plus $\mathrm{M6}^{\mathrm{u}}$ conditions; the table reports the central M2/M5 contrast.

	M2: Think-Off			M5: Strict		
Encoder	**Bool**	**MCQ**	**Open**	**Bool**	**MCQ**	**Open**
all-MiniLM-L6-v2	0.046	0.366	0.701	0.001	0.041	0.455
Qwen3-Embedding-4B	**0.047**	**0.399**	**0.817**	**0.001**	**0.044**	**0.527**

We also compare MiniLM, BGE-base-en-v1.5, and E5-base-v2 on a fixed, balanced 3,600-output sample [38, 29, 44, 37]. The sample contains 50 outputs from each of 72 cells: six models, three answer spaces, and M1/M2/M3/M5. We use each encoder's recommended pooling and query/passage prefixes and assign a score of zero to empty T. Their retrieval interfaces differ from Qwen3-Embedding-4B's reasoning-focused query setup; this comparison tests encoder sensitivity.

Table D.2: Level 2 sensitivity to commonly used encoders on the balanced 3,600-output sample. Absolute cosine scales are encoder-specific; the relevant comparisons are the directions within each row.

	M2: Think-Off			M5: Strict		
Encoder	**Bool**	**MCQ**	**Open**	**Bool**	**MCQ**	**Open**
all-MiniLM-L6-v2	0.119	0.460	0.677	0.052	0.132	0.439
BGE-base-en-v1.5	0.284	0.424	0.759	0.154	0.262	0.518
E5-base-v2	0.263	0.431	0.890	0.251	0.365	0.638

Across both comparisons, every encoder preserves Bool $<$ MCQ $<$ Open under M2 and M5 $<$ M2 in every answer space.

D.2 Level 3: Cross-Family Judge Robustness

GPT-5.5 is the primary Level 3 judge. We apply Claude Opus 4.8 to the complete corpus with the same blinded input, prompt, five-way rubric, and supporting-span requirement, covering all six models and benchmarks under M1–M5 plus $\mathrm{M6}^{\mathrm{u}}$. For each non-empty T, both judges receive only (Q, T); empty T is assigned no visible inference.

Table D.3: Cross-family replication of the main EIR result. Values are all-output explicit-inference rates; GPT-5.5 is the primary judge and Claude Opus 4.8 is the independent cross-family judge.

	M2: Think-Off			M5: Strict		
Judge	**Bool**	**MCQ**	**Open**	**Bool**	**MCQ**	**Open**
GPT-5.5	5.15%	45.57%	99.69%	0.12%	5.39%	62.00%
Claude Opus 4.8	4.24%	40.73%	98.55%	0.10%	5.11%	61.02%

Separating paraphrases from explicit inference. The five-way GPT-5.5 rubric distinguishes semantic paraphrases from explicit inference, including paraphrases with little lexical overlap with the question. Table D.4 reports the M2/M5 rates using all outputs as the denominator for both categories. Across M1–M5, GPT-5.5 classifies 477 of 477,240 outputs (0.100%) as paraphrase-only. Appendix J reports the corresponding total with $\mathrm{M6}^{\mathrm{u}}$ included. Only the explicit-reasoning label contributes to EIR; paraphrases and relevant non-inferential text contribute zero.

Table D.4: Explicit inference and paraphrase-only rates under the primary GPT-5.5 judge. Both rates use all outputs in the corresponding setting as the denominator; empty T contributes to neither numerator.

Mode	**Answer space**	**Explicit reasoning**	**Paraphrase-only**
M2	Bool	5.15%	0.03%
	MCQ	45.57%	0.11%
	Open	99.69%	0.02%
M5	Bool	0.12%	0.01%
	MCQ	5.39%	0.03%
	Open	62.00%	1.00%

Both judges give Bool $<$ MCQ $<$ Open under M2 and M5, and M5 $<$ M2 in every answer space. On Open-ended tasks under M5, Opus identifies visible explicit inference in 61.02% of outputs, compared with GPT-5.5's 62.00%.

Appendix J reports agreement between the two judges over the full M1–M5 and $\mathrm{M6}^{\mathrm{u}}$ corpus.

D.3 Human Validation

We report the metric- and conclusion-level results of the blinded human study, followed by its annotation protocol and sampling procedure.

Metric validation. At **Level 1**, the strict-compliance decision underlying ETR agrees with the manually inspected empty-T indicator on 565 of 600 outputs (94.2%); the remaining 35 discrepancies reflect the additional requirement that the final answer be parseable in the permitted format. At **Level 2**, the primary instruction-aware Qwen3-Embedding-4B QRel. scores reach a mean item-level Spearman correlation of $\rho = 0.829$ with the five annotators' relevance judgments, compared with 0.809 for all-MiniLM-L6-v2 (Table D.6).

Table D.5: Human and LLM-judge explicit-inference rates across the six sampled M2/M5 answer-space cells. All entries use the all-output denominator within each cell; the Human mean row averages the five annotator-specific rates.

	M2: Think-Off			M5: Strict		
Evaluator	**Bool**	**MCQ**	**Open**	**Bool**	**MCQ**	**Open**
GPT-5.5	4.0%	29.6%	100.0%	0.0%	11.8%	57.1%
Claude Opus 4.8	0.0%	29.6%	100.0%	0.0%	11.8%	53.6%
Human mean	**4.0%**	**30.4%**	**100.0%**	**0.0%**	**11.8%**	**57.1%**

At **Level 3**, the human explicit-inference labels achieve 98.8% mean binary agreement with the primary GPT-5.5 judge and 96.7% with the independent Claude Opus 4.8 judge across all 600 outputs. Table D.5 further shows that the corresponding all-output rates closely track the human mean across the six sampled M2/M5 answer-space cells; Appendix D.3 provides the per-annotator agreement and reliability statistics.

Table D.6: Human relevance calibration on the unified 600-output sample. Each entry is an item-level Spearman correlation over all 600 outputs; empty T receives relevance score 1 and QRel. 0, and the final column reports the mean across annotators.

	A1	**A2**	**A3**	**A4**	**A5**	**Mean**
Qwen3-Embedding-4B QRel.	0.828	0.831	0.830	0.833	0.823	**0.829**
all-MiniLM-L6-v2	0.794	0.811	0.814	0.827	0.798	0.809

Conclusion validation. The mean human rates in Table D.5 preserve both experimental trends. Under M2 and M5, explicit inference rises from Bool to MCQ to Open-ended; relative to M2, M5 lowers the rate in every answer space, including from 100.0% to 57.1% on Open-ended tasks.

These direct annotations therefore reproduce the two conclusions drawn in Sections 3.2 and 3.3: visible explicit inference becomes harder to suppress as the answer space opens and remains common on Open-ended tasks even under the strictest natural-language instruction.

Human judgments support the three-level evaluation and its central conclusions. The boundary audit corroborates ETR, relevance judgments correlate with QRel., and explicit-inference labels agree strongly with the primary and independent LLM judges. The same annotations reproduce the answer-space staircase and persistent visible inference on Open-ended tasks under M5.

Study design. We complement the original pre-answer evaluation with a blinded human study that considers two aspects: *first*, whether the three metrics align with human judgments; and *second*, whether the main conclusions drawn from them persist under direct human annotation. **(i) Metric validation.** A manual boundary audit checks the strict visible boundary underlying ETR, while five annotators independently provide graded relevance judgments for comparison with QRel. and explicit-inference labels for comparison with the Level 3 judge. Additionally, we use a second LLM judge from a different model family to check that the Level 3 results are not specific to the primary judge. **(ii) Conclusion validation.** Building on this agreement, we test whether the main answer-space and intervention trends persist under direct human annotation. The annotation protocol follows.

Study sample. Our unified blinded human study samples 600 outputs, 100 from each benchmark, while preserving each benchmark's full-corpus empty/non-empty-T proportion; the empty-T boundary is manually audited for all outputs. Five annotators independently review all 402 non-empty pre-answer texts and assign both a five-point question-relevance score and one Level 3 reasoning label; the 198 empty cases receive relevance score 1 and no visible inference by definition. The annotation view exposes only (Q, T); model identity, intervention mode, final answer, correctness, and automated evaluator scores are hidden.

D.3.1 Annotation protocol

The human study evaluates 600 outputs from M2 and M5.

Sampling and blinding. The sample contains 100 outputs from each benchmark. After rounding, the proportions of empty and non-empty T match those in that benchmark's full corpus. We manually audit whether T is empty for all 600 outputs. The annotation view exposes only (Q, T) and hides dataset, model, mode, final answer, correctness, QRel., and judge labels. Empty T receives relevance score 1 and no visible inference by definition. Five annotators were recruited from the authors' friends, colleagues, and classmates and participated voluntarily. They independently review the same 402 non-empty texts.

Level 2 relevance annotation. For every non-empty T, annotators assign one ordinal score: **1**, no relevant content (generic, off-topic, or unrelated text); **2**, weak relevance (mostly boilerplate or only loosely related text); **3**, on-topic but non-evidential content (a relevant restatement, definition, or assertion without question-specific support); **4**, question-specific support (at least one relevant piece of evidence, comparison, computation, or inference); or **5**, substantive multi-step support (multiple connected question-specific steps or a developed derivation). This score evaluates graded pre-answer relevance and is kept separate from the Level 3 reasoning decision.

For annotator a, let h_{ai} be the human score and q_i the QRel. score for output i. We compute item-level Spearman correlation

$$\rho_a = \mathrm{Corr}(\mathrm{rank}(h_{ai}), \mathrm{rank}(q_i)) \,, \tag{5}$$

where ties receive their average rank. Table D.6 reports every ρ_a and their arithmetic mean $\bar{\rho} = \frac{1}{5}\sum_{a=1}^{5} \rho_a$.

Level 3 reasoning annotation. Independently of the relevance score, annotators choose exactly one of the same five mutually exclusive labels used by GPT-5.5 and Opus 4.8: generic/off-topic, paraphrase-only, relevant non-inferential, explicit reasoning, or unclear. The criteria are identical to Table C.2; in particular, only deduction, computation, comparison, elimination, rule application, evidence-to-conclusion, or causality enters the explicit-reasoning category. A restatement or relevant assertion without an inferential step therefore does not enter the EIR numerator. Annotators also record a supporting span from T.

Aggregation and agreement. All rates in Table D.5 use the full number of sampled outputs in each M2/M5 answer-space cell, including empty T, and the human row is the arithmetic mean of the five annotator-specific rates. We compute per-example five-way agreement using exact rubric-label matches and binary agreement by collapsing the rubric into explicit reasoning versus all other labels. Krippendorff's α is computed across the five annotators on the 402 independently reviewed non-empty texts: nominal distance for the five-way and binary reasoning labels, and ordinal distance for the five-point relevance scores.

Table D.7: Per-annotator relevance–QRel. calibration on the 600-output human sample. Values are item-level Spearman correlations.

Encoder	A1	A2	A3	A4	A5	Mean
Qwen3-Embedding-4B	0.828	0.831	0.830	0.833	0.823	**0.829**
all-MiniLM-L6-v2	0.794	0.811	0.814	0.827	0.798	0.809

Table D.8: Per-annotator agreement with the blinded Level 3 judges on all 600 outputs. Five-way agreement requires the same rubric label; binary agreement compares explicit-reasoning versus other labels.

Judge / agreement	A1	A2	A3	A4	A5	Mean
GPT-5.5 / Five-way	98.50%	97.83%	98.17%	97.33%	97.50%	**97.87%**
GPT-5.5 / Binary	98.67%	98.83%	99.00%	98.83%	98.67%	**98.80%**
Opus 4.8 / Five-way	96.00%	95.67%	95.83%	95.00%	95.17%	**95.53%**
Opus 4.8 / Binary	96.50%	97.00%	96.83%	96.67%	96.50%	**96.70%**

Table D.9: Inter-annotator reliability on the 402 non-empty pre-answer texts.

Annotation variable	Krippendorff's α
Five-way reasoning label (nominal)	0.873
Explicit reasoning vs. other (nominal)	0.948
Five-point relevance score (ordinal)	0.919

Table D.10: Per-annotator all-output explicit-inference rates for the six sampled M2/M5 cells. The human mean in Table D.5 is the column-wise mean.

Mode	Answer space	A1	A2	A3	A4	A5	Mean
M2	Bool	4.0%	4.0%	4.0%	4.0%	4.0%	**4.0%**
M2	MCQ	29.6%	29.6%	33.3%	29.6%	29.6%	**30.4%**
M2	Open	100.0%	100.0%	100.0%	100.0%	100.0%	**100.0%**
M5	Bool	0.0%	0.0%	0.0%	0.0%	0.0%	**0.0%**
M5	MCQ	11.8%	11.8%	11.8%	11.8%	11.8%	**11.8%**
M5	Open	57.1%	57.1%	57.1%	57.1%	57.1%	**57.1%**

The evaluation artifacts include the per-example annotations and fixed sample IDs. Across the six cells, the ranking by mean human rate matches the rankings by mean instruction-aware QRel. ($\rho = 1.000$) and GPT-5.5 EIR ($\rho = 1.000$), and nearly matches the ranking by Opus 4.8 EIR ($\rho = 0.986$). Judge–annotator agreement is computed item by item over all sampled outputs.

E Measurement Robustness Audits

We audit response order in 477,240 M1–M5 outputs from six models and six datasets, then examine the 22,748 assistant-prefill M6 outputs under the residual-text protocol.

E.1 Response Order and Restatement under M1–M5

Near-verbatim question repetition is rare under a lexical audit. We check M1–M5 outputs for near-verbatim repetition of the question. The detector compares the question stem, without appended option labels or instruction wrappers, with four fields. These are `extracted_t` and `content`, together with `reasoning` and `reasoning_content`. It removes complete boxed-answer spans, lowercases the text, and tokenizes ASCII letters and numbers. A record is flagged if any component has at least four tokens, differs from the question's token length by at most a factor of 1.5, and shares at least 80% of the distinct consecutive four-grams in *both* directions. This flags 176 of 477,240 records (0.037%); even assigning all flagged QRel. values to zero changes the corpus mean by at most $176/477{,}240 = 0.00037$. Including $\mathrm{M6}^{\mathrm{u}}$, the same detector flags 310 of 572,688 outputs (0.054%; Appendix J).

Semantic paraphrases do not enter the inference numerator. The primary judge classifies 477 M1–M5 outputs (0.100%) as paraphrase-only. Only the explicit-reasoning label contributes to EIR; paraphrases and relevant non-inferential text do not. Table D.4 gives the central M2/M5 breakdown.

Answer-first responses with a suffix. We audit 445,180 M1–M5 outputs with a parseable `\boxed{}` or `\fbox{}` in the canonical answer-view content. The strict lexical check flags a response when the

first box begins after at most three letter/number tokens and at least three such tokens follow it. It finds 62 records (0.014% of boxed outputs; Table E.1). The low rate supports the pre-answer protocol for boxed M1–M5 responses; this check captures the literal answer-first format defined above. ETR checks the complete returned output, including any substantive suffix. For QRel. and EIR, extraction ends at the selected answer, leaving subsequent explanations outside T. M6 begins with a supplied answer declaration and uses the residual-text protocol below.

Table E.1: M1–M5 format audit. Copy uses the four-gram detector and all responses as its denominator. Answer-first-plus-tail uses boxed answer-view outputs and the strict lexical rule defined above.

Model	Responses	Copy	Answer-first-plus-tail
Qwen3-4B	181,750	117 (0.064%)	23/164,895 (0.014%)
Qwen3-32B	181,750	28 (0.015%)	27/170,686 (0.016%)
Qwen3-235B-A22B	28,435	3 (0.011%)	10/28,273 (0.035%)
DeepSeek-V4-Flash	28,435	10 (0.035%)	1/28,272 (0.004%)
GPT-5.4	28,435	0 (0.000%)	0/28,417 (0.000%)
Gemini-3-Flash-Preview	28,435	18 (0.063%)	1/24,637 (0.004%)
All	**477,240**	**176 (0.037%)**	**62/445,180 (0.014%)**

QRel. separates relevance from verbosity. In the right panels of Figure 4, pre-answer length under M1 (THINK-ON) grows sharply from Bool to Open-ended tasks for all four models with complete length traces. Yet QRel. changes much less across these answer spaces, indicating that it is not merely a proxy for length. QRel. and EIR remain complementary: question-related T may be a paraphrase or relevant assertion, whereas EIR identifies whether it contains an explicit inferential step. Our M1–M5 audit flags 176 of 477,240 outputs (0.037%) as near-verbatim repetitions. Setting all their QRel. values to zero changes the mean over this corpus by at most 0.00037. An audit of the stored Level 3 labels identifies 477 M1–M5 outputs (0.100%) as paraphrase-only; these do not enter the EIR numerator. Appendix E, with Tables E.1 and D.4, provides the definitions and breakdowns.

E.2 Assistant-Prefill M6: Structure and Evaluation

Cohort and implementation. The M6 evaluation contains 22,748 generations: 5,687 per model for Qwen3-4B, Qwen3-32B, Qwen3-235B-A22B, and DeepSeek-V4-Flash. Each model uses the same item manifest: 1,000 items from each of BoolQ, MMLU, MMLU-Pro, GSM8K, and MATH, plus all 687 StrategyQA items. Thus, the per-model family counts are 1,687 Bool, 2,000 MCQ, and 2,000 Open-ended. The three Qwen models run locally; generation continues from an unfinished assistant response containing `The answer is \boxed{`. DeepSeek uses a prefix-continuation interface with model version 0731. All four models use an 8,192-token response budget, temperature zero, and a configuration that requests thinking to be disabled. GPT-5.4 and Gemini-3-Flash-Preview are omitted because we could not verify an assistant-continuation interface for these endpoints. Relative to M2/M4/M5, M6 uses smaller samples and a longer budget for Qwen3-4B and Qwen3-32B (1,024 tokens under M2–M5), local inference for Qwen3-235B in place of its API endpoint, and a different DeepSeek version.

Observable response structures. Table E.2 partitions all generations by the first prefilled box and subsequent visible structure. The answer-only test covers all returned readable fields, while the later-box counts use the content channel alone. Separately returned reasoning fields have no assumed position relative to that channel. A response with a usable first box is answer-only when its residual text is empty. Otherwise, it has no later content-channel box, later boxes that all match the initial box, or at least one later box that differs. Box comparisons ignore whitespace and letter case and compare literal text; a difference need not change the answer's meaning. Explicit inference in the residual text is assessed separately by the judge. The unusable category takes precedence and contains 877 empty boxes, 532 unclosed boxes, and 447 complete boxes from which the benchmark answer normalizer extracts no answer. Of these, 1,854 occur in DeepSeek and two in Qwen3-32B. There are also 648 outputs with empty residual T but no usable first answer; these receive ETR=0.

Table E.2: Response structures under assistant-prefill M6. The five categories are mutually exclusive; parentheses give percentages of all outputs in the row. Residual text includes all returned readable text outside the first declaration. Later boxes are counted in the content channel only.

Model / family	N	Answer-only (ETR)	Residual, no later box	Later boxes all same	A later box differs	Unusable first box
Qwen3-4B	5,687	3,293 (57.90%)	511 (8.99%)	864 (15.19%)	1,019 (17.92%)	0 (0.00%)
Qwen3-32B	5,687	5,427 (95.43%)	42 (0.74%)	149 (2.62%)	67 (1.18%)	2 (0.04%)
Qwen3-235B	5,687	3,771 (66.31%)	1,594 (28.03%)	252 (4.43%)	70 (1.23%)	0 (0.00%)
DeepSeek	5,687	3,302 (58.06%)	194 (3.41%)	60 (1.06%)	277 (4.87%)	1,854 (32.60%)
All	22,748	15,793 (69.43%)	2,341 (10.29%)	1,325 (5.82%)	1,433 (6.30%)	1,856 (8.16%)
Bool	6,748	5,497 (81.46%)	769 (11.40%)	5 (0.07%)	9 (0.13%)	468 (6.94%)
MCQ	8,000	5,466 (68.33%)	1,219 (15.24%)	286 (3.57%)	99 (1.24%)	930 (11.62%)
Open-ended	8,000	4,830 (60.38%)	353 (4.41%)	1,034 (12.93%)	1,325 (16.56%)	458 (5.73%)

First-answer accuracy and residual-text metrics. We parse the first prefilled box with balanced braces and pass only that isolated box to the same benchmark answer normalizer and comparator used for M1–M5. Accuracy depends only on the first box, regardless of later corrections. An empty, unclosed, or unparseable first answer receives Acc.=0 and ETR=0; ETR requires a complete box with a nonempty normalized prediction. We remove the entire initial declaration when its box is complete, including all text inside the box. If the box never closes, we retain the complete content. We join nonempty `reasoning`, `reasoning_content`, readable nonsummary `reasoning_details`, and residual content in that order, retaining the first occurrence of each part after whitespace normalization and case folding for duplicate comparison. We then append readable reasoning summaries, applying the same rule to the merged text and each summary as whole parts. We trim and collapse whitespace in the remaining content and the merged text. A string consisting only of a single full stop (ASCII or ideographic) is treated as empty. We pass residual T to the Qwen3-Embedding-4B encoder and GPT-5.5 five-way judge, using the same query instruction and judge rubric as for pre-answer text. Empty T receives QRel.=0 and EIR=0 without a model call. Removing the first box also removes any equations or prose inside it, so residual-text scores exclude inference within that box. Later answer assertions and corrections remain in T and are visible to the judge.

Denominators and aggregation. Acc., ETR, and QRel. use all 22,748 outputs. The GPT-5.5 judging API returned content-filter errors for 30 outputs, which are excluded only from EIR, leaving 22,718 outputs. Table E.3 reports the family-level values and per-group denominators N_{EIR}. Within a model, tasks are item-weighted. The means and population standard deviations in Figure 5 give equal weight to each available model. Gemini is excluded from the M2/M4/M5 ETR means and standard deviations in Figure 5. M6 differs from M2/M4/M5 in model coverage and evaluation scope.

Table E.3: M6 with first-box accuracy and evaluation of residual text from all returned readable fields. Acc., ETR, and EIR are percentages. N_{EIR} excludes only judge-filtered records; all other columns use N.

Model	Family	N	Acc.	ETR	QRel.	EIR	N_{EIR}
Qwen3-4B	Bool	1,687	77.00	97.93	0.016	1.54	1,686
Qwen3-4B	MCQ	2,000	55.40	81.05	0.152	18.62	1,992
Qwen3-4B	Open-ended	2,000	18.60	1.00	0.793	98.90	2,000
Qwen3-32B	Bool	1,687	81.56	100.00	0.000	0.00	1,687
Qwen3-32B	MCQ	2,000	68.10	99.00	0.008	0.80	2,000
Qwen3-32B	Open-ended	2,000	33.55	88.00	0.091	11.25	2,000
Qwen3-235B	Bool	1,687	84.11	57.68	0.316	33.61	1,684
Qwen3-235B	MCQ	2,000	72.75	45.45	0.433	48.49	1,986
Qwen3-235B	Open-ended	2,000	40.65	94.45	0.043	4.90	2,000
DeepSeek	Bool	1,687	55.96	70.24	0.070	2.61	1,687
DeepSeek	MCQ	2,000	43.45	47.80	0.205	18.09	1,996
DeepSeek	Open-ended	2,000	38.25	58.05	0.300	36.70	2,000

Examples and interpretation. Table E.4 illustrates several M6 response structures. The opening declaration includes the supplied prefix, and a bracketed ellipsis marks omitted text. The pear example follows (A, T). The crustacean example compares sizes before revising "yes" to "no", yielding (A, T, A'). In the arithmetic example, the computation appears inside the first box and is therefore removed before residual-text evaluation.

Table E.4: Examples of M6 response structure. Text is reproduced from the stored content field with line breaks normalized. Q235 denotes Qwen3-235B; DS denotes DeepSeek.

Source / structure	Observed content
Qwen3-32B, MMLU 10039; answer-only	The answer is $\boxed{\mathrm{C}}$.
Q235, StrategyQA 206; (A, T)	The answer is $\boxed{\mathrm{no}}$. A pear typically floats in water because its density is less than that of water.
Q235, StrategyQA 168; (A, T, A')	The answer is $\boxed{\mathrm{yes}}$. [...] Since the crab's leg span is significantly larger than the dimensions of the mattress, it would not be able to stretch out completely on a king-sized mattress. Correction: The answer is $\boxed{\mathrm{no}}$.
Q235, GSM8K 723; computation inside box	The answer is $\boxed{170 + 23 - 20 - 13 = 160}$.
DS, BoolQ 2373; empty first box	The answer is $\boxed{\ }$ no, because Hawaii is a U.S. state, not a territory.
DS, MMLU 12206; unclosed first box	`The answer is \boxed{B`

The answer-only instruction has a clear literal reading. We ask GPT to explain the instruction used in Figure 1: *"Give the final answer directly without any reasoning."* It interprets the request as returning only the result, without explanation, and gives answer-only examples (Figure E.1). This interpretation matches the answer-only behavior targeted by M5 and measured by ETR.

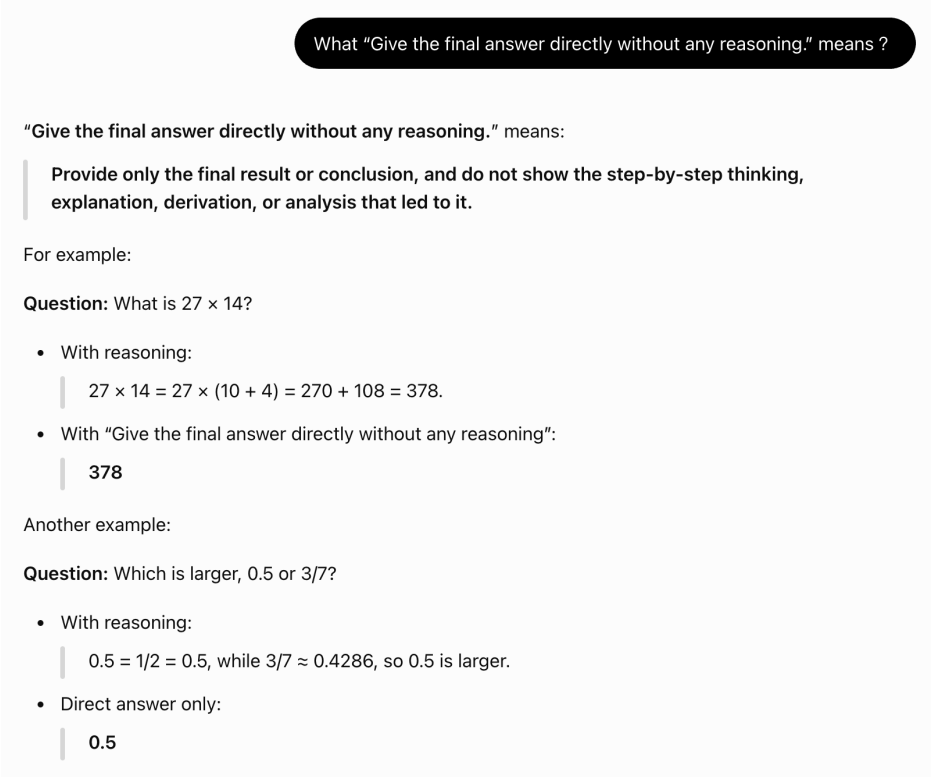

Figure E.1: Interpretation of the answer-only instruction. Complete visible exchange asking GPT to interpret the instruction in Figure 1. The response distinguishes a worked solution from the requested direct-answer format.

F Representative Case Studies

F.1 Representative Responses by Length and Relevance

Table F.1 gives five observed responses that illustrate how response length and question relevance can vary independently. All final answers are correct. Cases 1–4 use the same sampled MMLU item, so their differences reflect the model and intervention with the question held fixed. Case 5 uses a StrategyQA item because none of the evaluated model–mode pairs produced a long, low-QRel. response on the shared MMLU item. The regime names describe these examples without defining additional classification thresholds.

Table F.1: Observed responses with different lengths and degrees of question relevance. $|T|_{\mathrm{w}}$ is the visible pre-answer word count. QRel. is computed by Qwen3-Embedding-4B between the question and complete visible T under the instruction in Appendix D.1; empty T is assigned zero. All five final answers are correct.

Case	Regime	Model and mode	Record	$\lvert T\rvert_{\mathrm{w}}$	QRel.
1	Answer only	Qwen3-235B-A22B, M5	MMLU, `mmlu-2663`	0	0.000
2	Short length, high QRel.	DeepSeek-V4-Flash, M2	MMLU, `mmlu-2663`	23	0.896
3	Short length, low QRel.	Qwen3-235B-A22B, M2	MMLU, `mmlu-2663`	4	0.310
4	Long length, high QRel.	Gemini-3-Flash-Preview, M3	MMLU, `mmlu-2663`	144	0.831
5	Long length, low QRel.	Gemini-3-Flash-Preview, M1	StrategyQA, `strategyqa-227`	137	0.276

Shared input for Cases 1–4. *As of 2017, the share of GDP spent on the military by the United States is about: A. 1%, B. 3%, C. 6%, or D. 10%.* The gold option is B.

Case 1: answer only. *Qwen3-235B-A22B, M5 (Strict).*
Visible pre-answer T: $\emptyset$
Final answer A: $\boxed{\text{B}}$

Case 2: short length, high QRel. *DeepSeek-V4-Flash, M2 (Think-Off).*
Visible pre-answer T: The United States military spending as a share of GDP was around 3% to 3.5% in 2017. Thus, the correct option is B.
Final answer A: $\boxed{\text{B}}$

Case 3: short length, low QRel. *Qwen3-235B-A22B, M2 (Think-Off).*
Visible pre-answer T: The correct answer is:
Final answer A: $\boxed{\text{B}}$

Case 4: long length, high QRel. *Gemini-3-Flash-Preview, M3 (CoT).*
Visible pre-answer T: To find the share of GDP spent on the military by the United States in 2017, we can look at data from several reliable sources:
1. **World Bank and SIPRI (Stockholm International Peace Research Institute):** According to these organizations, the United States' military expenditure as a percentage of its Gross Domestic Product (GDP) in 2017 was approximately **3.31%**.
2. **U.S. Government Data:** In 2017, the U.S. GDP was approximately \$19.48 trillion, and the total national defense spending was approximately \$646 billion. Calculating the percentage, $(\$646 \text{ billion} / \$19.48 \text{ trillion}) \times 100 \approx$ **3.31%**.
Comparing this value to the given options: A. 1% is too low; B. 3% is closest to 3.31%; C. 6% is too high and was last seen in the 1980s; and D. 10% is too high and was last seen during the Korean War. The most accurate option is 3%.
Final answer A: $\boxed{\text{B}}$

Input for Case 5. *Was Eve involved in an incestuous relationship?* The gold answer is yes.

Case 5: long length, low QRel. *Gemini-3-Flash-Preview, M1 (Think-On).*
Visible pre-answer T: When considering the origins of a population from a perspective that prioritizes the long-term survival and flourishing of humanity, the moral value of an action is determined by its contribution to the aggregate well-being of all sentient beings. In a scenario where the continuation of the species depends on reproduction between the only existing individuals, the preservation of the human race—and the billions of lives and potential happiness that follow—represents the highest possible utility.
If the foundational narrative requires that the initial propagation of humanity occur through unions between closely related individuals, the immense benefit of human existence would outweigh the biological or social concerns typically associated with such relationships in larger populations. Within the internal logic of this account, these unions were a necessary means to achieve the greatest good: the existence of the human species.
Final answer A: $\boxed{\text{yes}}$

The displayed text reproduces the complete visible T and final A from each record; only Markdown list and emphasis markers are converted to LaTeX typography.

F.2 The same MMLU question under three answer spaces

Same Question, Different Answer Space

Boolean verification	Multiple choice	Open-ended
The length of a rectangle is twice its width. Given the length of the diagonal is $5\sqrt{5}$, find the area. Proposed answer: 50. Is it correct? Answer yes or no.	**The length of a rectangle is twice its width. Given the length of the diagonal is $5\sqrt{5}$, find the area. Options: A 2500, B 2, C 50, D 25.**	**The length of a rectangle is twice its width. Given the length of the diagonal is $5\sqrt{5}$, find the area of the rectangle.**
M1 THINK-ON *Prompt:* Put the final answer in `\boxed{}`.		
<think> Okay, let's see. The length is twice the width, and the diagonal is $5\sqrt{5}$. We need to check whether the proposed answer 50 is correct. [...] Let width be w, length $2w$. Then $w^2 + (2w)^2 = (5\sqrt{5})^2$, so $5w^2 = 125$, $w = 5$, length = 10, area = 50. Final Answer: \boxed{*yes*}. </think> [...] The proposed answer is correct. \boxed{*yes*}.	<think> Okay, so I need to find the area where length is twice the width and the diagonal is $5\sqrt{5}$. [...] Let width be w; then length is $2w$, and area is $2w^2$. Since $(5\sqrt{5})^2 = 125$, $(2w)^2 + w^2 = 125$. [...] Thus $w = 5$, area = 50. Therefore, the answer is C. \boxed{*C*}. </think> [...] Final answer: \boxed{*C*}.	<think> Okay, so I need to find the area of a rectangle where the length is twice the width and the diagonal is $5\sqrt{5}$. [...] Denote the width by w, so length is $2w$. By the Pythagorean theorem, $w^2 + (2w)^2 = 125$. [...] Therefore $5w^2 = 125$, $w = 5$, length = 10, and area = 50. The area is \boxed{50}. </think> [...] The area is \boxed{50}.
M2 THINK-OFF *Prompt:* Put the final answer in `\boxed{}`.		
Let the width of the rectangle be w. Then the length is $2w$. The diagonal gives $w^2 + (2w)^2 = 125$, so $5w^2 = 125$ and $w = 5$. The area is $10 \times 5 = 50$. \boxed{*yes*}.	We are given $L = 2W$ and diagonal $5\sqrt{5}$. Using $L^2 + W^2 = d^2$, $(2W)^2 + W^2 = 125$, so $W = 5$, $L = 10$, and area = 50. Final answer: \boxed{*C*}.	We are given $L = 2W$ and diagonal $5\sqrt{5}$. Then $(2W)^2 + W^2 = (5\sqrt{5})^2 = 125$, so $5W^2 = 125$, $W = 5$, $L = 10$, and the area is 50. \boxed{50}.
M3 COT *Prompt:* Think step by step. Put the final answer in `\boxed{}`.		
Step 1: let width be W and length $2W$. Step 2: $(2W)^2 + W^2 = 125$, so $W = 5$. Step 3: area = 50. Final answer: \boxed{*yes*}.	Step 1: let width be x, length $2x$. Step 2: $(5\sqrt{5})^2 = (2x)^2 + x^2$, so $x = 5$. Step 3: area = $2x^2 = 50$. \boxed{*C*}.	Step 1: let width be x, length $2x$. Step 2: use the Pythagorean theorem. Step 3: $x^2 = 25$, $x = 5$. Step 4: area = $2x^2 = 50$. \boxed{50}.
M4 SOFT *Prompt:* A very short lead-in is allowed, but do not explain. Put the final answer in `\boxed{}`.		
\boxed{*yes*}.	Let the width be x. Then the length is $2x$. Using the Pythagorean theorem, $x^2 + (2x)^2 = (5\sqrt{5})^2$, so $x = 5$. Area = 50. \boxed{*C*}. ✓	Let the width be x. Then the length is $2x$. Using the Pythagorean theorem, $x^2 + (2x)^2 = 125$, so $x = 5$. The area is $5 \times 10 = 50$. \boxed{50}.
M5 STRICT *Prompt:* Give the final answer directly without any reasoning. Put the final answer in `\boxed{}`.		
\boxed{*yes*}. ✓	\boxed{*C*}. ✓	Let the width of the rectangle be x. Then the length is $2x$. Using the Pythagorean theorem for the diagonal, $x^2 + (2x)^2 = (5\sqrt{5})^2$, so $5x^2 = 125$ and $x = 5$. So, width is 5 and length is $2 \times 5 = 10$. The area is $10 \times 5 =$\boxed{50}. ✓
M6 PREFIX *Assistant prefill:* `The answer is \boxed{`		
The answer is \boxed{yes}. ✓	The answer is \boxed{C}. ✓	The answer is \boxed{50}. ✓

Figure F.1: The same MMLU question under three answer spaces. Qwen3-32B is evaluated with the six intervention modes on a single MMLU geometry item rewritten as Boolean verification, multiple-choice answering, and open-ended answering. Blue text shows output excerpts, blue boxed text marks the parsed final answer, green checks mark strict answer-only following, and gray [...] marks omitted spans. In M1, `<think>` and `</think>` expose the boundary between the visible thinking trace and the ordinary answer text. In M6, gray text is the supplied assistant prefix and blue text is the full generated continuation.

F.3 Cases behind the aggregate findings

Table F.2 illustrates four aggregate findings with individual outputs: visible computation under native no-think control, re-elicitation, compression cost, and candidate visibility.

Table F.2: Representative cases of native no-think behavior, re-elicitation, compression cost, and candidate visibility.

Case	Record setting	Observed pattern	Interpretation
Native residual information	Qwen3-4B, MATH, item `math-4`, Mode 2	Correct prediction 4, ETR fails, $\|T\| = 84$ words. The visible response rewrites $4 = 2^2$, sets $2^8 = 2^{2x}$, and solves $x = 4$.	Disabling native thinking does not force an open-ended math item into answer-only behavior; the ordinary answer channel still carries computation.
Re-elicitation	Qwen3-235B-A22B, BoolQ, item `boolq-0`, Modes 2–3	Mode 2 returns only `\boxed{no}` and passes ETR. Mode 3 remains correct but expands to $\|T\| = 124$ words and fails ETR.	A step-by-step instruction can recover visible question-conditioned content even when the provider no-think interface remains active.
Compression cost	Qwen3-4B, MATH, item `math-811`, Modes 2 and 5	Mode 2 is correct: prediction -9, $\|T\| = 61$. Mode 5 gives a short direct response but predicts 9 while the gold answer is -9.	Stronger direct-answer pressure can remove the computation needed to check a subtle constraint, here the word "smallest."
Candidate visibility	Qwen3-32B, MMLU abstract algebra, item `mmlu-39`, Mode 2	MCQ form is correct but has $\|T\| = 209$ and fails ETR. Boolean verification of the same content returns direct `yes`/`no` answers for both a true and a false proposal, and both pass ETR.	Changing the answer interface can make the same underlying content much more compressible because the candidate answer is already visible.

G Additional Global Results

Figure G.1 summarizes the local Qwen results for M1–M5 and the user-message-prefix condition $\mathrm{M6}^{\mathrm{u}}$, using the pre-answer definitions of Acc., ETR, QRel., and EIR. Assistant-prefill M6 results appear in Figure 5 and Appendix E.2.

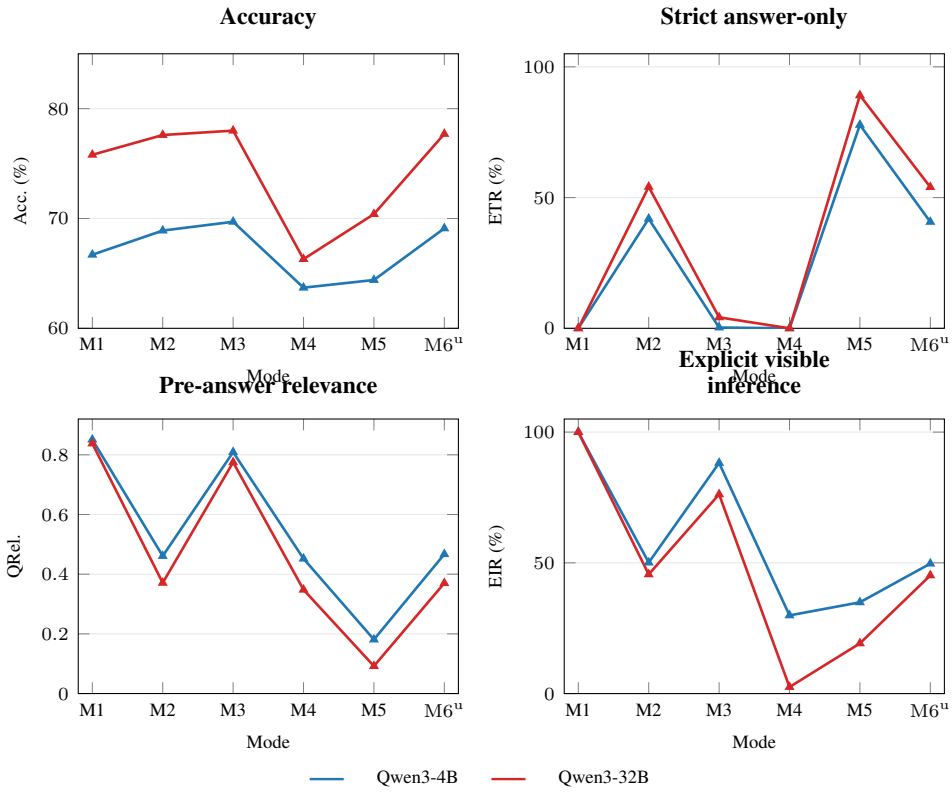

Figure G.1: Local Qwen profiles for M1–M5 and the user-message-prefix condition $\mathrm{M6}^{\mathrm{u}}$. All metrics use the six-benchmark suite. Acc., ETR, and QRel. pool items across benchmarks; EIR first pools items within each answer-space family and then averages the three family rates with equal weight.

M2 raises strict answer-only compliance while leaving question-relevant and often inferential text. M3 can re-elicit this text. For both local models, the behavioral scores under $M6^{u}$ are close to those under M2.

Compact model labels. In the heatmaps below, Q4 denotes Qwen3-4B, Q32 denotes Qwen3-32B, Q235 denotes Qwen3-235B-A22B, DS denotes DeepSeek-V4-Flash, GPT denotes GPT-5.4, and Gem denotes Gemini-3-Flash-Preview. The same abbreviations are reused in the later MMLU domain figure.

Figure G.2 compares the accuracy change from M2 to M5 with the resulting M5 ETR for each dataset.

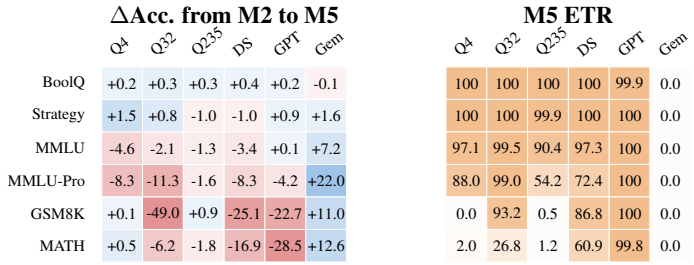

Figure G.2: Dataset-level view of the no-thinking–performance trade-off. The left panel shows the accuracy change from native no-think (M2) to strict answer-only suppression (M5); the right panel shows the resulting M5 ETR. The plotted values are the corresponding dataset-level aggregates.

H Detailed Experiment Tables

This section gives dataset, scaling, and domain results for M1–M5 and the user-message-prefix condition $M6^{u}$ under the pre-answer protocol.

Frontier native no-think behavior. Figure H.1 compares Bool, MCQ, and Open-ended tasks under native no-think control across providers. QRel. and EIR increase in that order for every model. ETR falls in the same order for Qwen3-235B, DeepSeek, and GPT, while Gemini produces no strictly answer-only outputs in any of the three answer spaces. Question-relevant and inferential text therefore persists on MCQ and Open-ended tasks even when the thinking interface is off or minimal.

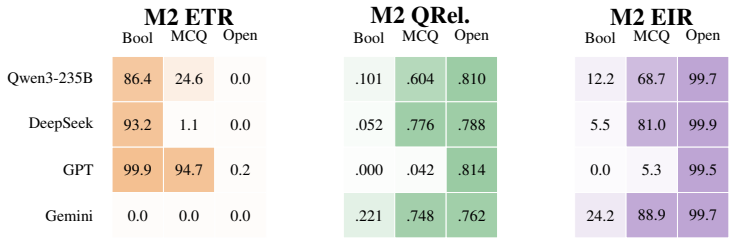

Figure H.1: Native no-think results by answer space across providers. The three panels evaluate the same M2 outputs for strict answer-only compliance (ETR), instruction-aware pre-answer relevance (QRel.), and visible explicit inference (EIR).

Qwen scaling under M1–M5 and $M6^{u}$. Table H.1 compares model size, intervention, and dataset within the Qwen family.

Interpretation of Table H.1. Larger models are generally more accurate, but the answer-space pattern of no-think behavior remains. BoolQ and StrategyQA are easiest to compress, while GSM8K and MATH remain hardest. The $M6^{u}$ results are often closer to M2 than to M5. In Table H.2, EIR follows Bool $<$ MCQ $<$ Open under both M2 and M5 for all three model sizes. M5 lowers or leaves unchanged the EIR observed under M2 in each answer space.

Table H.1: Qwen scaling across datasets under M1–M5 and the user-message-prefix condition $M6^{u}$. Blue, orange, and green shading encode Acc., ETR, and QRel., respectively. Table H.2 reports the corresponding Level 3 EIR comparison.

Mode	Dataset	Qwen3-4B			Qwen3-32B			Qwen3-235B		
		Acc.	ETR	QRel.	Acc.	ETR	QRel.	Acc.	ETR	QRel.
M1 THINK-ON	BoolQ	87.4	0.0	0.800	89.2	0.0	0.795	89.2	0.0	0.794
	StrategyQA	72.2	0.0	0.821	81.5	0.0	0.815	83.3	0.0	0.816
	MMLU	75.2	0.0	0.853	85.3	0.0	0.835	88.3	0.0	0.853
	MMLU-Pro	53.4	0.0	0.861	65.9	0.0	0.845	82.0	0.0	0.857
	GSM8K	87.0	0.0	0.808	93.1	0.0	0.801	94.2	0.0	0.823
	MATH	55.3	0.0	0.873	59.1	0.0	0.869	73.3	0.0	0.859
M2 THINK-OFF	BoolQ	85.9	98.1	0.007	89.4	100.0	0.000	90.4	98.4	0.012
	StrategyQA	64.9	92.3	0.058	71.3	100.0	0.000	75.4	68.9	0.231
	MMLU	73.4	52.2	0.371	82.7	79.2	0.166	86.4	32.3	0.543
	MMLU-Pro	56.7	33.6	0.517	69.1	38.0	0.491	77.9	16.8	0.666
	GSM8K	89.9	0.0	0.774	93.4	0.0	0.774	95.3	0.0	0.775
	MATH	69.1	0.0	0.849	72.3	0.0	0.846	75.9	0.0	0.844
M3 COT	BoolQ	85.6	2.3	0.733	89.1	38.3	0.314	87.3	13.6	0.696
	StrategyQA	70.3	0.4	0.800	75.7	6.0	0.640	78.9	0.6	0.763
	MMLU	74.8	0.0	0.820	84.1	1.2	0.825	87.7	0.0	0.832
	MMLU-Pro	57.5	0.2	0.807	68.6	0.5	0.820	78.6	0.0	0.818
	GSM8K	90.1	0.0	0.766	94.2	0.0	0.759	95.6	0.0	0.770
	MATH	68.6	0.0	0.850	72.5	0.0	0.848	74.9	0.0	0.844
M4 SOFT	BoolQ	85.9	0.2	0.318	89.5	0.0	0.319	90.1	15.3	0.275
	StrategyQA	66.2	0.0	0.347	71.3	0.0	0.342	75.1	11.4	0.311
	MMLU	69.6	0.0	0.372	80.4	0.0	0.346	84.4	4.9	0.358
	MMLU-Pro	51.2	0.0	0.455	56.9	0.0	0.363	69.5	6.5	0.433
	GSM8K	80.7	0.0	0.712	51.8	0.0	0.470	86.4	0.1	0.680
	MATH	57.9	0.1	0.703	36.9	0.0	0.306	69.7	0.5	0.718
M5 STRICT	BoolQ	86.1	100.0	0.000	89.7	100.0	0.000	90.7	100.0	0.000
	StrategyQA	66.4	100.0	0.000	72.1	100.0	0.000	74.4	99.9	0.001
	MMLU	68.8	97.1	0.022	80.6	99.5	0.004	85.1	90.4	0.074
	MMLU-Pro	48.4	88.0	0.092	57.8	99.0	0.008	76.3	54.2	0.358
	GSM8K	90.0	0.0	0.765	44.4	93.2	0.052	96.2	0.5	0.776
	MATH	69.6	2.0	0.831	66.1	26.8	0.625	74.1	1.2	0.833
$M6^{u}$ PREFIX	BoolQ	86.1	99.9	0.000	89.2	100.0	0.000	90.7	93.7	0.041
	StrategyQA	64.9	96.8	0.026	71.5	100.0	0.000	74.1	76.1	0.184
	MMLU	73.4	49.4	0.386	82.7	79.5	0.162	87.5	7.3	0.662
	MMLU-Pro	57.2	32.7	0.520	69.5	37.4	0.492	77.8	2.9	0.746
	GSM8K	91.0	0.0	0.773	93.6	0.0	0.774	96.2	0.0	0.776
	MATH	69.2	0.0	0.849	72.4	0.0	0.844	75.2	0.0	0.845

Note. Qwen3-4B and Qwen3-32B use local native `enable_thinking` controls on all six datasets. Qwen3-235B abbreviates Qwen3-235B-A22B and uses the provider-mediated API reasoning control. M1–M5 follow Section 2.3; $M6^{u}$ supplies the answer prefix in the user message.

Table H.2: Qwen scaling of visible explicit inference. EIR is the GPT-5.5 judge's explicit-inference rate over all outputs; empty T contributes zero. Results cover M2 (native no-think) and M5 (strict answer-only). Table H.1 gives Acc., ETR, and QRel. by dataset.

Model	Mode	Bool	MCQ	Open
Qwen3-4B	M2 THINK-OFF	1.26%	49.30%	99.86%
	M5 STRICT	0.00%	7.05%	97.78%
Qwen3-32B	M2 THINK-OFF	0.00%	37.11%	99.56%
	M5 STRICT	0.00%	0.72%	56.73%
Qwen3-235B	M2 THINK-OFF	12.15%	68.70%	99.70%
	M5 STRICT	0.00%	27.00%	98.90%

Full domain dependence view. Figure H.2 fixes the answer interface to test whether MMLU domain still affects compressibility.

Interpretation of Figure H.2. The native M2 pattern appears across multiple domains: Humanities and Social Sciences are generally easier to compress than STEM. Table I.2 reports the separate user-message-prefix comparison.

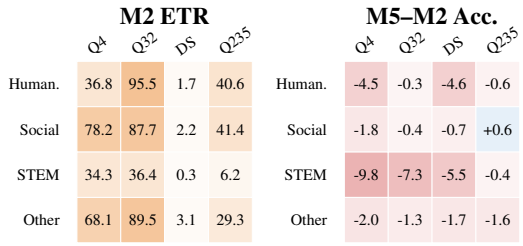

Figure H.2: MMLU domain heatmaps under the same multiple-choice answer interface. The left panel shows answer-only success under native no-think; the right panel shows the accuracy change caused by strict answer-only suppression. The full area-level control table is given in Table I.2.

I Question-Structure Details

I.1 Performance across disciplines

We next address the domain component of *Q3* by holding the answer interface fixed and comparing no-thinking behavior across subject areas.

Higher accuracy does not imply easier no-thinking. Table I.1 fixes the answer interface to four-choice MCQ and varies the MMLU subject area. Across all four models, STEM combines the highest accuracy with the lowest ETR (0.3–36.4%), arguing against a simple positive relationship between task success and answer-only compliance. STEM also has the highest QRel. (0.509–0.784) and EIR (59.7–85.5%) in every model block, showing that the domain pattern is not confined to ETR. Thus, under this fixed interface, answer-only compliance is not simply a proxy for task success, and answer space does not account for all variation in no-thinking behavior.

Table I.1: No-thinking behavior varies by domain even under the same four-choice answer interface. Following the official MMLU subject-area split [14], we report Humanities, Social Sciences, and STEM and omit the heterogeneous Other category. The % column gives each area's share of the full MMLU test split; for each model, the remaining columns report Acc., ETR, QRel., and EIR under M2.

		Qwen3-4B				Qwen3-32B				Qwen3-235B				DeepSeek-V4-Flash			
MMLU area	**%**	**Acc.**	**ETR**	**QRel.**	**EIR**	**Acc.**	**ETR**	**QRel.**	**EIR**	**Acc.**	**ETR**	**QRel.**	**EIR**	**Acc.**	**ETR**	**QRel.**	**EIR**
Humanities	33.5%	62.5	36.8	0.488	56.6	71.6	95.5	0.036	4.1	77.1	40.6	0.435	49.8	83.5	1.7	0.772	82.7
Social Sciences	21.9%	78.7	78.2	0.167	16.2	88.3	87.7	0.098	10.7	88.3	41.4	0.452	49.6	92.3	2.2	0.777	69.2
STEM	22.5%	83.0	34.3	0.527	60.2	91.3	36.4	0.509	59.7	90.6	6.2	0.752	85.5	94.4	0.3	0.784	83.5

Note. For API models, Acc./ETR use the complete MMLU aggregate, whereas QRel./EIR use a 1,000-output domain subset; details are provided in Appendix Table I.2.

The domain and question-transformation comparisons use the pre-answer protocol, with $M6^{u}$ denoting the user-message-prefix condition. Within each model and mode, we compare domains, perturb option labels and order, and rewrite matched questions as Boolean verification or open numeric generation. These controls test how answer support affects response behavior.

Table I.2: MMLU area-level controls. Each mode cell reports Acc./ETR/EIR. The n column gives the sample size for Acc./ETR; the note specifies the EIR populations. $\mathrm{M6^u}$ denotes a user-message answer prefix.

Model	MMLU area	n Acc./ETR	M1	M2	M5	$\mathrm{M6^u}$
Qwen3-4B	Humanities	4705	63.7 / 0.1 / 99.9	62.5 / 36.8 / 56.6	58.0 / 100.0 / 0.0	62.8 / 34.6 / 55.7
	Social Sciences	3077	82.8 / 0.5 / 100.0	78.7 / 78.2 / 16.2	76.9 / 99.8 / 0.2	78.1 / 73.6 / 17.3
	STEM	3153	81.9 / 0.0 / 100.0	83.0 / 34.3 / 60.2	73.2 / 88.8 / 11.2	83.1 / 33.3 / 59.9
	Other	3107	78.4 / 0.3 / 100.0	74.9 / 68.1 / 17.8	72.9 / 98.5 / 1.5	75.0 / 63.9 / 17.2
Qwen3-32B	Humanities	4705	77.3 / 4.5 / 100.0	71.6 / 95.5 / 4.1	71.3 / 100.0 / 0.0	71.6 / 94.9 / 4.5
	Social Sciences	3077	90.2 / 1.5 / 100.0	88.3 / 87.7 / 10.7	87.9 / 100.0 / 0.0	88.4 / 88.2 / 10.0
	STEM	3153	89.9 / 1.5 / 100.0	91.3 / 36.4 / 59.7	84.0 / 97.7 / 2.3	91.6 / 38.1 / 57.4
	Other	3107	87.7 / 2.3 / 100.0	85.2 / 89.5 / 9.6	83.9 / 100.0 / 0.0	85.0 / 89.7 / 9.1
DeepSeek-V4-Flash	Humanities	4705	88.1 / 0.0 / 100.0	83.5 / 1.7 / 82.7	78.9 / 98.8 / 1.2	83.6 / 2.0 / 76.2
	Social Sciences	3077	93.1 / 0.0 / 97.8	92.3 / 2.2 / 69.2	91.6 / 99.9 / 0.0	92.6 / 1.8 / 56.7
	STEM	3153	94.8 / 0.0 / 99.2	94.4 / 0.3 / 83.5	88.9 / 94.7 / 5.2	94.4 / 0.4 / 79.0
	Other	3107	91.9 / 0.0 / 99.0	90.4 / 3.1 / 55.1	88.7 / 99.8 / 0.0	90.2 / 2.1 / 45.4
Qwen3-235B	Humanities	4705	82.2 / 0.0 / 100.0	77.1 / 40.6 / 49.8	76.5 / 97.6 / 1.2	76.8 / 11.4 / 64.7
	Social Sciences	3077	90.9 / 0.0 / 100.0	88.3 / 41.4 / 49.6	88.9 / 96.8 / 0.4	89.8 / 12.9 / 55.8
	STEM	3153	94.0 / 0.0 / 100.0	90.6 / 6.2 / 85.5	90.2 / 63.3 / 33.1	92.5 / 1.2 / 85.1
	Other	3107	89.1 / 0.0 / 100.0	87.6 / 29.3 / 49.8	86.0 / 93.2 / 2.9	88.1 / 8.9 / 50.7

Note. M1 is the thinking-on reference, M2 is native no-think, M5 is strict answer-only, and $\mathrm{M6^u}$ is the user-message prefix. Acc. measures final-answer accuracy, ETR measures strict answer-only compliance, and EIR measures visible explicit inference over the evaluated outputs, including empty-T cases. For Qwen3-4B/32B, EIR uses the full split. For Qwen3-235B and DeepSeek, it uses 1,000 items per model and mode: 323 Humanities, 224 Social Sciences, 248 STEM, and 205 Other. Blue, orange, and purple shading encode Acc., ETR, and EIR, respectively.

MMLU subject view. Table I.3 expands the area-level control in Table I.2 to the original MMLU subjects.

Table I.3: Subject-level MMLU results grouped by the original supercategories. M1 cells report Acc.; M2, M5, and $\mathrm{M6^u}$ cells report Acc./ETR. Darker orange indicates higher ETR. $\mathrm{M6^u}$ uses a user-message answer prefix.

MMLU area / subject	n	Qwen3-4B M1 Acc.	Qwen3-4B M2 Acc./ETR	Qwen3-4B M5 Acc./ETR	Qwen3-4B $\mathrm{M6^u}$ Acc./ETR	Qwen3-32B M1 Acc.	Qwen3-32B M2 Acc./ETR	Qwen3-32B M5 Acc./ETR	Qwen3-32B $\mathrm{M6^u}$ Acc./ETR
Humanities	4705	**63.7**	**62.5/36.8**	**58.0/100.0**	**62.8/34.6**	**77.3**	**71.6/95.5**	**71.3/100.0**	**71.6/94.9**
formal logic	126	81.7	73.0/0.8	58.7/100.0	78.6/0.8	94.4	85.7/7.9	65.9/100.0	85.7/7.9
high school european history	165	77.0	78.8/55.2	75.2/100.0	80.6/53.3	87.9	84.2/91.5	84.2/100.0	83.6/90.9
high school us history	204	86.8	82.8/72.1	82.4/100.0	83.8/64.2	93.1	94.6/96.6	94.1/100.0	94.1/93.1
high school world history	237	85.7	82.7/64.6	84.0/100.0	83.1/58.2	91.6	90.7/93.7	90.3/100.0	90.3/92.0
international law	121	76.9	75.2/90.9	75.2/100.0	76.9/89.3	88.4	85.1/99.2	84.3/100.0	85.1/99.2
jurisprudence	108	79.6	75.9/80.6	77.8/100.0	77.8/72.2	86.1	84.3/97.2	86.1/100.0	86.1/98.1
logical fallacies	163	84.0	80.4/29.4	82.2/100.0	81.6/20.9	87.7	86.5/87.7	86.5/100.0	88.3/82.8
moral disputes	346	69.7	70.2/87.3	67.3/100.0	70.8/80.1	80.9	80.3/98.8	80.9/100.0	80.9/98.6
moral scenarios	895	61.2	55.6/0.0	36.5/100.0	53.9/0.0	73.0	58.8/100.0	58.2/100.0	58.7/100.0
philosophy	311	67.8	71.1/92.0	70.7/100.0	70.7/81.4	79.7	78.5/99.7	78.5/100.0	78.8/99.4
prehistory	324	75.6	71.3/77.8	70.7/100.0	70.7/65.4	92.0	87.0/99.7	88.3/100.0	86.7/99.1
professional law	1534	44.5	46.9/11.5	46.2/100.0	47.9/16.8	64.7	58.6/98.6	59.5/100.0	58.5/98.2
world religions	171	83.6	80.7/45.0	79.5/100.0	79.5/30.4	89.5	86.5/95.9	86.5/100.0	86.5/94.7
Social Sciences	3077	**82.8**	**78.7/78.2**	**76.9/99.8**	**78.1/73.6**	**90.2**	**88.3/87.7**	**87.9/100.0**	**88.4/88.2**
econometrics	114	59.6	65.8/39.5	61.4/100.0	59.6/36.8	76.3	78.1/33.3	73.7/100.0	77.2/40.4
high school geography	198	86.9	84.8/82.3	83.3/100.0	82.8/73.7	93.4	89.9/97.5	89.9/100.0	90.4/98.0
high school government and politics	193	91.7	83.4/72.5	83.4/100.0	84.5/65.3	98.4	96.9/95.9	96.4/100.0	97.4/96.9

Continued on next page

MMLU area / subject	n	Qwen3-4B M1 Acc.	Qwen3-4B M2 Acc./ETR	Qwen3-4B M5 Acc./ETR	Qwen3-4B M6^{u} Acc./ETR	Qwen3-32B M1 Acc.	Qwen3-32B M2 Acc./ETR	Qwen3-32B M5 Acc./ETR	Qwen3-32B M6^{u} Acc./ETR
high school macroeconomics	390	85.9	79.5/73.1	74.4/99.2	78.7/72.8	93.6	91.3/60.0	89.5/100.0	91.5/61.3
high school microeconomics	238	94.1	85.7/74.8	80.7/100.0	83.2/75.2	96.2	94.5/63.0	94.5/100.0	95.0/62.6
high school psychology	545	91.9	87.5/71.0	86.4/99.8	88.3/64.6	95.6	94.7/96.3	94.7/100.0	94.7/96.3
human sexuality	131	81.7	76.3/78.6	74.0/100.0	76.3/68.7	89.3	89.3/95.4	88.5/100.0	89.3/96.2
professional psychology	612	75.3	71.6/82.2	71.2/99.8	71.6/78.1	87.1	83.3/97.4	83.0/100.0	83.3/97.4
public relations	110	71.8	67.3/81.8	64.5/100.0	66.4/70.0	71.8	75.5/100.0	75.5/100.0	74.5/100.0
security studies	245	71.8	66.5/95.5	69.4/100.0	66.9/94.3	79.6	77.6/99.6	78.8/100.0	78.0/99.6
sociology	201	83.1	84.1/94.5	81.6/100.0	81.6/89.6	90.5	88.6/100.0	87.6/100.0	88.6/99.5
us foreign policy	100	81.0	82.0/89.0	78.0/100.0	84.0/80.0	92.0	89.0/99.0	90.0/100.0	89.0/98.0
STEM	3153	**81.9**	**83.0/34.3**	**73.2/88.8**	**83.1/33.3**	**89.9**	**91.3/36.4**	**84.0/97.7**	**91.6/38.1**
abstract algebra	100	72.0	73.0/0.0	57.0/97.0	77.0/0.0	90.0	92.0/0.0	72.0/100.0	92.0/0.0
anatomy	135	72.6	63.0/60.7	54.8/100.0	59.3/71.1	87.4	80.7/83.0	80.7/100.0	80.7/88.9
astronomy	152	84.9	84.9/71.1	77.6/100.0	86.2/65.1	94.1	94.7/78.9	92.8/100.0	94.1/80.9
college biology	144	91.7	84.7/82.6	81.9/100.0	84.0/84.7	98.6	97.2/83.3	93.8/100.0	97.9/84.7
college chemistry	100	51.0	63.0/27.0	52.0/100.0	60.0/25.0	67.0	71.0/14.0	66.0/100.0	73.0/21.0
college computer science	100	69.0	77.0/25.0	68.0/98.0	77.0/26.0	85.0	91.0/25.0	81.0/100.0	94.0/29.0
college mathematics	100	64.0	71.0/0.0	61.0/90.0	78.0/0.0	76.0	86.0/0.0	66.0/97.0	85.0/0.0
college physics	102	81.4	87.3/15.7	62.7/98.0	91.2/16.7	91.2	93.1/10.8	73.5/100.0	94.1/8.8
computer security	100	81.0	81.0/70.0	82.0/100.0	80.0/62.0	84.0	89.0/84.0	85.0/100.0	89.0/83.0
conceptual physics	235	88.1	81.3/46.4	77.0/99.6	81.3/36.2	94.9	94.5/38.7	93.2/100.0	94.0/37.9
electrical engineering	145	75.2	73.1/49.0	68.3/98.6	69.7/53.1	84.8	84.8/43.4	82.1/100.0	84.8/44.8
elementary mathematics	378	96.8	96.0/0.0	81.0/68.3	95.5/0.0	97.1	97.6/3.7	89.4/99.7	97.4/5.3
high school biology	310	92.6	87.4/84.5	86.8/100.0	87.7/82.9	95.8	95.2/85.8	94.8/100.0	95.5/89.4
high school chemistry	203	75.4	84.7/31.0	70.4/98.5	87.2/28.1	89.7	91.1/20.2	81.3/99.5	91.1/21.2
high school computer science	100	92.0	91.0/33.0	83.0/97.0	91.0/27.0	96.0	96.0/42.0	92.0/100.0	96.0/43.0
high school mathematics	270	77.4	89.6/0.0	76.3/37.4	91.1/0.0	84.4	92.2/0.0	74.8/75.6	93.0/0.0
high school physics	151	79.5	82.1/18.5	66.2/96.7	81.5/19.9	87.4	89.4/7.3	79.5/98.7	91.4/11.3
high school statistics	216	81.5	85.2/15.7	75.5/85.2	83.8/16.2	88.0	93.1/31.5	86.6/100.0	91.7/34.3
machine learning	112	75.9	75.0/30.4	58.0/98.2	70.5/32.1	88.4	78.6/58.9	74.1/100.0	79.5/59.8
Other	3107	**78.4**	**74.9/68.1**	**72.9/98.5**	**75.0/63.9**	**87.7**	**85.2/89.5**	**83.9/100.0**	**85.0/89.7**
business ethics	100	79.0	69.0/43.0	72.0/100.0	71.0/45.0	81.0	82.0/99.0	81.0/100.0	81.0/99.0
clinical knowledge	265	81.9	74.3/79.2	73.2/99.2	76.2/79.2	89.4	87.2/88.7	86.0/100.0	86.8/89.8
college medicine	173	79.2	78.0/62.4	75.7/99.4	80.3/63.0	83.8	85.0/67.1	81.5/100.0	82.7/68.8
global facts	100	47.0	44.0/73.0	41.0/100.0	43.0/27.0	58.0	59.0/99.0	57.0/100.0	58.0/99.0
human aging	223	71.3	68.6/82.5	68.6/100.0	69.1/66.4	84.3	82.1/100.0	83.0/100.0	81.2/100.0
management	103	88.3	79.6/88.3	79.6/100.0	79.6/77.7	90.3	88.3/100.0	87.4/100.0	88.3/99.0
marketing	234	88.9	87.2/85.9	86.8/100.0	86.8/82.9	93.6	93.6/98.7	93.6/100.0	93.6/98.3
medical genetics	100	85.0	82.0/67.0	76.0/100.0	79.0/69.0	97.0	96.0/89.0	95.0/100.0	96.0/87.0
miscellaneous	783	86.5	83.1/50.2	81.1/99.9	83.9/43.2	94.4	91.3/91.7	90.5/100.0	91.1/92.7
nutrition	306	77.1	73.5/83.7	70.9/99.7	72.2/80.1	91.8	85.3/94.1	85.3/100.0	85.0/95.1
professional accounting	282	71.6	69.5/41.1	62.4/85.1	66.3/45.0	85.5	80.9/56.4	72.3/100.0	81.9/54.3
professional medicine	272	80.9	76.8/83.8	74.6/100.0	76.8/95.2	93.0	91.2/94.9	89.3/100.0	91.2/95.2
virology	166	47.0	48.8/88.0	48.8/100.0	49.4/80.1	55.4	51.8/98.2	56.0/100.0	53.6/97.0

Note. Area labels follow the MMLU paper's four supercategories: Humanities, Social Sciences, STEM, and Other. The rows below each area list the original MMLU subjects in that area. Color encodes ETR only; accuracy is printed numerically.

Interpretation of Table I.3. The subject-level rows show the same split: quantitative subjects are harder to compress, while many fact-like subjects reach high ETR without the same accuracy cost.

Surface perturbations. Table I.4 checks whether the MMLU pattern can be explained by label names or option order. Cons. is the percentage of paired base and perturbed outputs whose nonempty predictions agree after the perturbed labels are mapped back to the original choices. All matched pairs enter the denominator; an empty prediction on either side counts as inconsistent.

Table I.4: MMLU surface perturbations with the answer set fixed. Deltas are relative to the base version; darker colored cells indicate larger absolute changes. $M6^{u}$ denotes a user-message answer prefix.

Model	Mode	Variant	Acc.	ETR	QRel.	ΔAcc.	ΔETR	ΔQRel.	Cons.
Qwen3-4B	Mode 2	base	74.2	53.0	0.369	–	–	–	–
		choice shuffle	72.3	52.7	0.372	-1.9	-0.3	+0.003	82.5
		numeric labels	73.4	45.0	0.426	-0.8	-8.0	+0.057	89.9
	Mode 5	base	68.9	97.0	0.023	–	–	–	–
		choice shuffle	67.0	96.8	0.024	-1.9	-0.1	+0.001	80.7
		numeric labels	68.5	96.8	0.024	-0.4	-0.1	+0.001	93.0
	$M6^{u}$	base	74.5	49.9	0.385	–	–	–	–
		choice shuffle	72.5	50.4	0.383	-2.0	+0.5	-0.002	81.8
		numeric labels	73.2	45.6	0.414	-1.2	-4.3	+0.029	88.7
Qwen3-32B	Mode 2	base	82.4	79.6	0.164	–	–	–	–
		choice shuffle	82.1	79.0	0.168	-0.3	-0.6	+0.004	90.2
		numeric labels	82.0	84.2	0.126	-0.4	+4.6	-0.038	95.8
	Mode 5	base	79.8	99.4	0.005	–	–	–	–
		choice shuffle	79.0	99.4	0.005	-0.8	+0.0	+0.000	88.3
		numeric labels	78.5	99.5	0.004	-1.3	+0.1	-0.001	94.7
	$M6^{u}$	base	82.4	79.6	0.162	–	–	–	–
		choice shuffle	82.0	79.1	0.165	-0.4	-0.5	+0.003	90.1
		numeric labels	81.8	84.3	0.125	-0.6	+4.7	-0.037	95.1

Interpretation of Table I.4. The modest shifts in Acc./ETR/QRel. and the high consistency argue against a label-name or option-order explanation.

Candidate visibility. Table I.5 rewrites MMLU items as yes/no verification with one candidate answer shown.

Table I.5: MMLU boolean verification with a proposed answer shown. Darker cells indicate larger metric values or larger changes from MCQ. $M6^{u}$ denotes a user-message answer prefix.

		MCQ base			Boolean verification					Change	
Model	**Mode**	Acc.	ETR	QRel.	Acc.	ETR	QRel.	Pos.	Neg.	ΔAcc.	ΔQRel.
Qwen3-4B	Mode 1	77.6	0.1	0.852	80.0	65.6	0.833	82.4	77.6	+2.4	-0.019
	Mode 2	75.0	52.8	0.370	73.5	67.2	0.235	65.5	81.6	-1.5	-0.135
	Mode 5	69.4	97.0	0.023	69.1	100.0	0.000	52.9	85.2	-0.3	-0.023
	$M6^{u}$	74.6	50.4	0.382	72.9	76.8	0.179	63.9	81.8	-1.7	-0.203
Qwen3-32B	Mode 1	86.2	3.1	0.835	84.6	94.8	0.822	88.4	80.9	-1.6	-0.013
	Mode 2	82.6	79.9	0.162	81.2	96.9	0.026	81.1	81.2	-1.4	-0.136
	Mode 5	80.0	99.6	0.004	80.6	100.0	0.000	80.3	80.8	+0.5	-0.004
	$M6^{u}$	82.5	80.4	0.155	81.2	96.8	0.026	81.2	81.0	-1.3	-0.129

Interpretation of Table I.5. Under no-think conditions, Boolean verification raises ETR and lowers QRel. relative to the MCQ form. The Pos./Neg. columns show that models both accept true proposals and reject false ones.

Removing options from numeric-source items. Table I.6 tests the effect of removing the visible candidate set while keeping the answer source numeric.

Table I.6: Matched numeric-source items under MCQ, boolean verification, and open generation. Darker cells indicate larger open-generation values or larger Open–MCQ accuracy drops. $M6^{u}$ denotes a user-message answer prefix.

		MCQ			Bool verification			Open numeric					Drop
Model	**Mode**	Acc.	ETR	QRel.	Acc.	ETR	QRel.	Acc.	ETR	QRel.	Parse	Exact	Open–MCQ
Qwen3-4B	Mode 1	66.8	0.0	0.832	82.0	13.5	0.847	40.6	0.0	0.841	99.6	38.3	-26.2
	Mode 2	72.1	3.7	0.760	81.2	12.1	0.722	50.3	0.0	0.811	97.0	47.2	-21.8
	Mode 5	53.8	63.4	0.281	60.4	99.4	0.005	48.6	13.4	0.678	98.0	45.8	-5.2
	$M6^{u}$	72.0	1.4	0.771	81.6	12.9	0.717	50.6	0.0	0.811	97.9	47.5	-21.4
Qwen3-32B	Mode 1	72.7	0.2	0.821	81.2	44.8	0.841	43.9	0.0	0.832	99.8	41.6	-28.9
	Mode 2	77.8	10.5	0.709	78.9	62.5	0.314	52.6	0.0	0.810	98.0	49.0	-25.2
	Mode 5	52.9	94.1	0.046	69.6	100.0	0.000	40.4	82.7	0.119	98.0	38.0	-12.4
	$M6^{u}$	78.2	10.8	0.707	78.9	62.5	0.315	52.3	0.0	0.811	98.1	48.9	-25.9

Interpretation of Table I.6. Removing the options requires the model to generate a numeric value. Compared with MCQ, open numeric generation under M5 has lower accuracy and higher QRel. for both models. For Open numeric, Parse is the percentage of extracted answers that can be parsed as decimals, integer fractions, or percentages. Acc. compares their exact numeric values, while Exact compares answer strings after removing outer mathematical wrappers, normalizing whitespace, and removing commas. All three rates use all matched Open outputs as the denominator.

J User-Message-Prefix Results

User-message prefix. In $M6^{u}$, an answer-prefix template is appended to the *user message*, after which the model starts a new assistant response. M6 instead supplies the prefix as an unfinished assistant turn that the model continues.

Accuracy comparison. Across M2, M4, M5, and $M6^{u}$, Bool accuracy is relatively flat, MCQ accuracy varies more, and Open-ended tasks show the largest changes. In the natural-language comparison from M2 to M5, greater answer-only compliance accompanied reduced accuracy on Open-ended tasks. Under $M6^{u}$, Open-ended accuracy returned toward M2 levels. Accuracy here uses the final answer selected by the pre-answer parser.

Response-behavior comparison. The user-message prefix did not strengthen answer-only compliance relative to M5: from M5 to $M6^{u}$, mean ETR fell on MCQ and Open-ended tasks, while mean QRel. and EIR increased in all three families. On Open-ended tasks, ETR returned near zero and relevance, visible inference, and accuracy moved toward M2. For $M6^{u}$, QRel. and EIR evaluate the text preceding the selected final answer. M6 uses the residual-text definition in Appendix E.2, which also includes text after the first answer.

Experimental coverage. Table J.1 specifies the intervention and population used in each analysis.

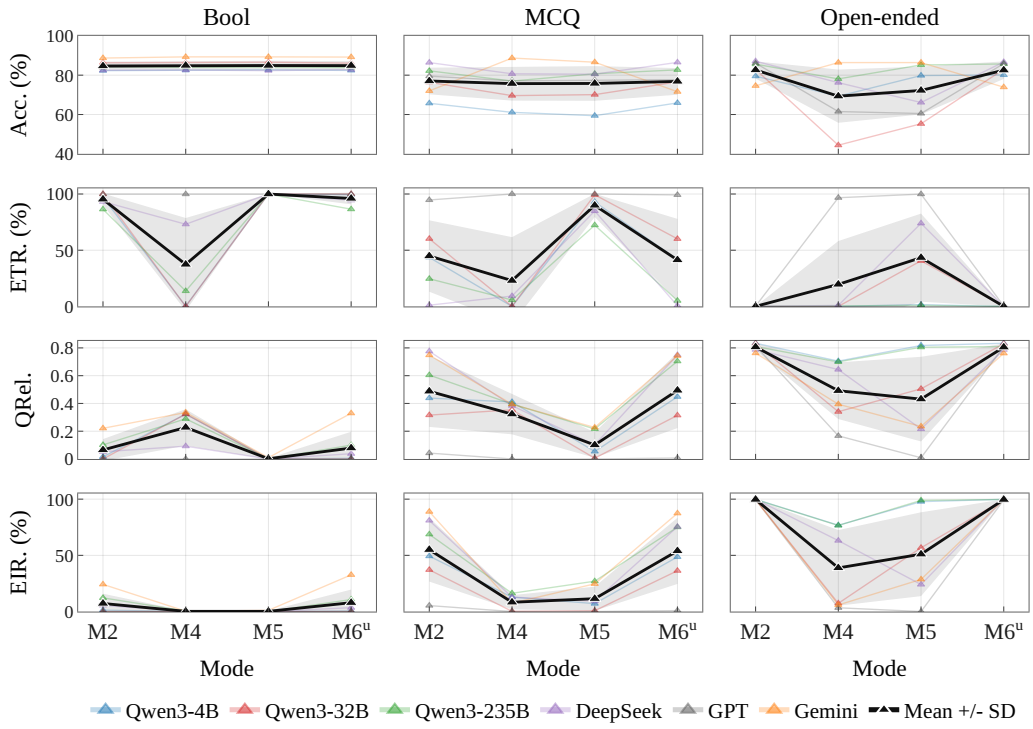

Figure J.1: User-message-prefix results across answer spaces. $M6^{u}$ supplies the answer prefix in the user message; M2/M4/M5 are the comparison modes. Colored lines show individual models; black lines and gray bands show their mean and one standard deviation.

Table J.1: Interventions and evaluation scope.

Result block	Intervention and evaluation scope
Figure 5; Tables E.2–E.4	Assistant-prefill M6: four models, 22,748 generations, first-box accuracy and residual-text metrics.
Table E.1	M1–M5 format/copy audit.
Tables D.3, D.4; native and M2/M5 comparisons	Native no-thinking and natural-language suppression comparisons under the pre-answer protocol.
Figure G.1; Table H.1	Global/scaling analyses with $M6^{u}$ as the user-message-prefix condition.
Tables I.2, I.3	MMLU domain/subject analyses under the pre-answer protocol, including $M6^{u}$; metric-specific sample sizes are given with the tables.
Tables I.4, I.5, I.6	Surface, candidate, and numeric-source transformations under the pre-answer protocol, including $M6^{u}$.
Table J.2; encoder/human validation	Full-corpus encoder and judge comparisons: M1–M5 and $M6^{u}$; human validation: the 600-output M2/M5 sample.

J.1 Whole-Corpus Judge Agreement

We compare the two judges on the 572,688-output corpus covering M1–M5 and $M6^{u}$. Opus returns usable labels for 572,645 outputs. The remaining 43 (0.0075%) receive no usable label from the API and are excluded only from the direct agreement calculation. Table J.2 reports agreement by model, benchmark, and mode. Five-way agreement requires identical rubric labels; binary agreement distinguishes explicit reasoning from all other categories.

Table J.2: GPT-5.5–Opus 4.8 agreement for M1–M5 plus $\mathrm{M6}^{\mathrm{u}}$. Each group pools the other experimental dimensions. Percentages are calculated over outputs for which both judges have a usable label.

Group	N	Five-way	Explicit vs. other
Overall			
All outputs	572,645	96.92%	97.30%
Non-empty T	391,010	95.48%	96.05%
Model			
Qwen3-4B	218,078	95.78%	96.23%
Qwen3-32B	218,079	98.47%	98.77%
Qwen3-235B-A22B	34,122	96.27%	96.43%
DeepSeek-V4-Flash	34,122	94.21%	94.78%
GPT-5.4	34,122	99.54%	99.88%
Gemini-3-Flash-Preview	34,122	94.96%	95.56%
Benchmark			
BoolQ	63,240	96.60%	98.02%
StrategyQA	24,732	96.30%	97.13%
MMLU	192,486	96.80%	96.88%
MMLU-Pro	168,359	96.05%	96.29%
GSM8K	39,828	99.62%	99.95%
MATH	84,000	98.04%	98.53%
Mode			
M1	95,433	98.91%	98.96%
M2	95,448	96.24%	96.40%
M3	95,434	92.93%	93.57%
M4	95,434	98.69%	99.62%
M5	95,448	99.30%	99.56%
$\mathrm{M6}^{\mathrm{u}}$	95,448	95.43%	95.69%

Agreement remains high after removing empty-T cases and within each model, benchmark, and mode; the matched EIR rates in Table D.3 reproduce the Level 3 ordering under the independent judge.

J.2 Whole-Corpus Format Audit

This audit covers the M1–M5 plus $\mathrm{M6}^{\mathrm{u}}$ corpus. Table E.1 reports the M1–M5-only audit, and Table E.2 reports M6 response structures.

Table J.3: Whole-corpus format audit for M1–M5 plus $\mathrm{M6}^{\mathrm{u}}$. Copy uses the same four-gram detector as Appendix E. Answer-first requires the first box to occur after at most three lexical tokens and to be followed by at least three lexical tokens. Percentages for Copy use all successful responses; Answer-first percentages use boxed responses.

Model	Responses	Copy	Answer-first
Qwen3-4B	218,100	235 (0.108%)	23/199,186 (0.012%)
Qwen3-32B	218,100	31 (0.014%)	27/205,146 (0.013%)
Qwen3-235B-A22B	34,122	4 (0.012%)	97/33,930 (0.286%)
DeepSeek-V4-Flash	34,122	19 (0.056%)	1/33,935 (0.003%)
GPT-5.4	34,122	0 (0.000%)	0/34,104 (0.000%)
Gemini-3-Flash-Preview	34,122	21 (0.062%)	1/28,907 (0.003%)
All	**572,688**	**310 (0.054%)**	**149/535,208 (0.028%)**

Paraphrase-only outputs. The primary judge classifies 655 outputs as paraphrase-only in the M1–M5 plus $\mathrm{M6}^{\mathrm{u}}$ corpus, including 477 from M1–M5. These outputs do not contribute to EIR, which counts only the explicit-reasoning category.

K Discussion, Limitations, and Future Work

We argue that "no-thinking" in LLMs should be evaluated as a response-level property rather than inferred from surface proxies such as missing reasoning tags, no-think settings, or short generations. By decomposing each output into pre-answer text T and final answer A (the first boxed answer A and residual text T for M6), we progressively measure strict answer-only compliance, pre-answer relevance, and visible explicit inference, while task accuracy evaluates A independently. Across local Qwen3 models, scaled variants, and frontier APIs, we find that explicit no-think controls do not reliably remove visible explicit inference; models exhibit "*thinking inertia*" organized by answer space; stronger answer-only constraints reveal a *no-thinking–performance trade-off* between compliance and accuracy; and compressibility is governed by the structural support of the question itself, not by question difficulty or model capability alone. These findings suggest that building LLMs

that can stop thinking on demand requires more than hiding reasoning traces or enforcing shorter outputs: it requires evaluating when direct answering is actually supported by the task.

K.1 Scope and Limitations

Our analysis deliberately takes the *observable response* as its unit of study. This choice provides a common behavioral basis across local models and proprietary APIs whose internal computation is neither equally exposed nor directly comparable. Our claims therefore concern visible output, not the absence of task-relevant computation from hidden states; provider signals are incorporated when available without treating them as equivalent descriptions of internal reasoning.

Our six-model, six-benchmark suite is designed to isolate answer-space structure under deterministic decoding whenever supported. Generalization to multilingual, code, tool-use, multi-turn, and stochastic settings, as well as robustness to evolving proprietary endpoints, remains open. QRel. is an operational relevance complement to the exact ETR boundary rather than a standalone reasoning detector. EIR directly separates explicit inference from paraphrases and relevant non-inferential text, but remains dependent on the visible-text rubric and judge calibration; Appendix D tests this dependence with an independent judge family, and Appendix E audits near-verbatim copying, semantic paraphrases, and response order. These boundaries delimit the claims while preserving a reproducible, provider-agnostic basis for studying no-thinking.

K.2 Implications and Future Work

No-thinking is a distinct controllable capability rather than the default complement of a reasoning mode. Direct answering depends on answer-space support and computational necessity rather than on a switch or model scale alone. This suggests task-aware routing: answer directly when the interface provides sufficient support, and preserve explicit computation when suppression would sacrifice accuracy. Visible compliance should not, however, be equated with lower inference cost when hidden reasoning may remain active.

One direction is to train models for calibrated stopping rather than uniform suppression, jointly optimizing accuracy, answer-only compliance, visible relevance and inference, latency, and actual reasoning-token cost. Extensions across languages and interactive tasks can test generality, while internal-state analyses can examine whether computation disappears, moves to a hidden channel, or is only omitted from view. In high-stakes settings, concise outputs may reduce cost and disclosure, but explanations remain important for verification and error detection. The aim is to make models' decisions about *when* and *how* to expose intermediate computation controllable and auditable, rather than to suppress such computation universally.

L Additional Related Work

Efficient and Adaptive Reasoning. Test-time computation can substantially improve language-model reasoning [31, 3], motivating more selective allocation. One line of work controls the length or budget of visible reasoning through length-conditioned reinforcement learning [1], token-budget prompting [11], budget forcing [23], and self-training for concise traces [24]. Another compresses intermediate reasoning by distilling shorter chains [18] or using terse drafts [45]; a recent survey provides a broader taxonomy [32]. Longer traces can impair efficiency and accuracy on simple problems; useful reasoning length depends on task difficulty and model capability [4, 43, 49]. Recent methods therefore train models to decide whether extended reasoning is needed [9, 47, 16, 34]. Selective reasoning has also been studied in vision-language models [36, 21].

Our scope differs from optimizing the amount of test-time computation. We ask whether a requested no-thinking state is realized in the *observable response*, following the common think/no-think focus on exposed CoT or reasoning channels rather than unverifiable latent cognition. We disable provider thinking interfaces or impose answer-only formats, then separately measure strict answer-only compliance, pre-answer relevance, and visible explicit inference. This distinction matters because intermediate computation need not be expressed in natural language, as demonstrated by latent-reasoning methods such as Coconut [12], and because a visible CoT need not faithfully reflect the computation that produced an answer [35, 20]. Recent work also studies whether concealed reasoning can be reconstructed or exposed. Trace Inversion synthesizes distillation-useful reasoning traces from

black-box inputs, answers, and optional summaries without access to hidden CoT [48]. Encrypted reasoning blocks exposed by proprietary APIs can also be transferred to weaker models within the same provider and decoded into plaintext [27]. Prompt repetition offers a complementary input-side technique for improving non-reasoning models and evaluates final-answer accuracy, length, and latency [22]. It does not ask whether visible question-conditioned text remains, nor does it distinguish paraphrases and relevant assertions from inference. Our contribution instead decomposes the response into pre-answer text and the final answer and evaluates input- and output-side interventions through ETR, QRel., EIR, task accuracy, and same-question answer-space controls.

Instruction Following. Our intervention spectrum relies on instruction following: the "no-think," "answer-only," and prefix-forcing modes depend on models obeying natural-language constraints. Instruction tuning [41, 30, 5] and human- and AI-feedback alignment [26, 2] train models to follow open-ended directives, while self-generated and synthetic instruction data [39, 33] broaden the instruction distribution. Instruction-following benchmarks such as IFEval [51] and FollowBench [17] evaluate compliance with output-format and content constraints, and related benchmarks study model behavior under complex or multi-constraint instructions [13, 28]. We treat "produce no *visible* reasoning" as a verifiable output constraint, and ask whether surface compliance removes visible question-conditioned payload rather than only matching the requested format.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Sections 3.2–3.4 test the claims about answer-space structure, no-thinking controls, and task accuracy. The measurement scope and limitations are stated in Section 2.2 and Appendix K.1.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Appendix K.1 discusses the restriction to observable responses, task and model coverage, evolving API endpoints, and dependence on the relevance encoder and inference judge.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper presents an empirical evaluation and does not state theoretical results requiring proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Sections 2.2 and 2.3 define the metrics and interventions; Appendices B, C, and E.2 specify the data, decoding settings, response parsing, and aggregation. The public code and response trajectories provide a route to reproduce or verify the reported results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The evaluation code and execution instructions are available at https://github.com/thinking-inertia/code, and the response trajectories are available at https://huggingface.co/datasets/thinking-inertia/llm-nothinking-trajectories. The release documents the separate pre-answer and assistant-prefill evaluation protocols.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Appendix B gives dataset splits, sample sizes, model interfaces, and decoding settings; Appendices C and E.2 give the evaluation and assistant-prefill details. The study evaluates existing models without training or fine-tuning.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: Figure 5 reports individual model results and their mean with one population standard deviation, with the aggregation defined in Appendix E.2. These bands describe variation across models rather than uncertainty across repeated runs.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Appendices B.2 and E.2 describe local and API deployment settings and generation budgets, but do not provide per-experiment wall-clock times, memory requirements, or a total compute estimate.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research follows the NeurIPS Code of Ethics. Appendix K.2 discusses implications for evaluation and the use of concise outputs in high-stakes settings.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix K.2 discusses task-aware routing and calibrated stopping, as well as the risk of losing explanations needed for verification and error detection when reasoning is suppressed.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The released assets are evaluation code and benchmark-based response records; the work does not release newly trained model weights or a scraped personal-data corpus requiring controlled access.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [No]

    Justification: The paper cites the benchmark and model sources, and the public release preserves upstream attribution and terms-of-use notices. It does not enumerate the exact license name for every asset.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The public code repository provides installation and evaluation instructions. The trajectory release includes a dataset card, field schema, release manifest, and upstream notices for the generated response records.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [No]

    Justification: Appendix D.3 describes the five volunteer annotators, their recruitment, the sample, blinding, and annotation rubrics. Compensation details and the complete participant-facing instructions are not reported.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [No]

    Justification: Appendix D.3 describes voluntary participation in the annotation study. Institutional approval or exemption, formal consent procedures, and participant risk disclosure are not reported.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Section 2.2 describes the LLM-based explicit-inference judge and its rubric, and Appendix D.2 gives the independent-judge validation. Sections 2.3 and 3.1 identify the evaluated LLMs and intervention interfaces.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.