ILLUSIONS OF REASONING FRAGILITY: HOW EVALUATION ARTIFACTS INFLATE APPARENT LLM STRUCTURAL SENSITIVITY

Anonymous authors
Paper under double-blind review

ABSTRACT

The new investigations reveal that the potential cognitive power of large language models is compromised even by a slight modification in the prompt given. The present paper presents an instance of the GSM8K performance, which equals approximately 15%. The issue at the heart of such failure is two miscalculations in the evaluation rather than deficiency in the model's abilities. The first mistake is the additional padding in the decoding process; the second mistake emerges once the instructions are given in the prompt after the question mark and inherent double question bias. Symmetric Prompt format is suggested to eliminate the above-mentioned problems. Testing the Qwen2.5-7B-Instruct model demonstrates the restoration of more than 80% accuracy, which indicates that the problem of the illusion of reasoning loss has been resolved (echo rate drops to 0.0%). Interestingly, Llama-3.1-8B-Instruct proved largely immune to the initial artifact, maintaining high accuracy throughout. Logistic regression allows us to separate the effects of structure from those of length when testing the Qwen model ($p = 0.405$), while for Llama the found β value is equal to 1.09 at $p = 0.004$. The third model, Mistral-7B, fails under the adjusted conditions.

1 INTRODUCTION

A number of recent studies show that LLMs have serious difficulties with math word problems, when their structure gets altered (Mirzadeh et al., 2025; Shi et al., 2023; Li et al., 2024). Specifically, researchers found that accuracy on GSM8K-inspired problems may decrease from 30 up to 85 percentage points when researchers slightly modify numerical values, rules and/or add distractor sentences. These results were used as evidence of fundamental brittleness of LLMs mathematical reasoning even with chain-of-thought prompting (Wei et al., 2022).

We argue that many of these failures may be explained by flaws in evaluation procedure. In the present paper, we provide a case study of an example where an 85 percentage point degradation in the accuracy of GSM8K was actually caused by two types of pipeline mistakes:

1. **Right side padding in batched inference.** By default, HuggingFace `transformers` applies right-side padding. It is okay for encoder-only model but causes introducing artificial positional information that corrupts the decoding process for decoder-only models (Section 4).
2. **Double question prompt confound.** Most evaluation pipelines attach new constraints to the end of the original problem. However, since GSM8K questions end with a question mark, it creates a prompt with two distinct questions. Models have a strong recency bias and simply reiterate the original answer for the second question which is perceived as reasoning "amnesia" (Section 5).

Once we fixed these confounds with a new *Symmetric Prompt* design, Qwen2.5-7B's massive accuracy collapse disappeared, recovering to $> 80\%$ structural retention with a 0.0% echo rate. Interestingly, Llama-3.1-8B proved largely immune to the artifact from the beginning, maintaining $\sim 80\%$ retention regardless of prompt formatting. A third model (Mistral-7B-Instruct-v0.3) still struggled. In this case,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Flawed Prompt (Phase 2)	**Symmetric Prompt (Phase 3.5)**

`A store sells apples for $3 each.`
`If Maria buys 4 apples,`
`how much does she spend?`
`All prices are doubled.`
`What is the final amount?`
`A store sells apples for $3 each.`
`Maria buys 4 apples.`
`Given the above rule, what is the final amount?`
	`[Rule: All prices are doubled.]`

↓ *Two "?" tokens* Recency bias → echo original	✓ *One unified question* Model computes new answer
Retention: ∼15%	Retention: >80%

Figure 1: Double-question artifact prompt (left) compared to our Symmetric Prompt solution (right). In the problematic version, double question marks induce a recency bias causing the model to repeat the original answer rather than perform the new calculation. Symmetric Prompt merges rule and problem into a single question, avoiding the artifact. Numbers refer to Qwen2.5-7B-Instruct.

however, the performance drop was caused by overall sensitivity to prompt changes (both lexical and structural noise) rather than the double-question artifact.

Last but not least, we needed to rule out the possibility that our findings were driven by the length of input text, because the complexity of the rules required more words to be written out. We employed a clustered logistic regression grouped by the problem seed. There was no significant residual effect of complexity on Qwen's performance ($p = 0.405$), although there still remained some structural sensitivity in Llama ($\beta = 1.09$, $OR = 2.97$, $p = 0.004$).

Contributions.

1. We identify two pipeline artifacts (padding corruption and the double-question confound) that lead to an inflated LLM failure rate.
2. We propose the Symmetric Prompt to avoid the double-question confound and test it on GSM8K.
3. We replicate our findings with two state-of-the-art architectures, showing that "reasoning amnesia" is a measurement error.
4. We perform a covariate-adjusted analysis employing clustered logistic regression to separate the effect of mathematical complexity from the effect of text length.

Figure 1 shows the illustration of the artifact itself.

2 Related Work

LLM mathematical reasoning robustness. Cobbe et al. (2021) introduced GSM8K as a benchmark for multi-step arithmetic reasoning. Mirzadeh et al. (2025) demonstrated that accuracy on GSM8K drops significantly when numerical values are replaced with symbolic variables, concluding that models rely heavily on pattern matching rather than robust reasoning. Li et al. (2024) introduced GSM-Plus, a comprehensive robustness benchmark covering multiple perturbation types including distractor insertion and numerical variation. Our work reframes these findings: while accuracy drops are real, their magnitude is substantially inflated by the double-question artifact present in append-style evaluation pipelines.

Distractor sensitivity. Shi et al. (2023) introduced GSM-IC (Grade-School Math with Irrelevant Context), showing that appending irrelevant sentences to GSM8K problems significantly reduces LLM accuracy. They attribute this to model "distractibility." Our work provides a complementary

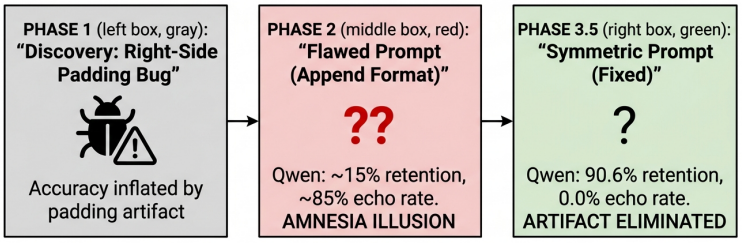


Figure 2: Overview of our three-phase experimental pipeline. **Phase 1** (gray): discovery of the right-side padding artifact inflating accuracy estimates. **Phase 2** (red): evaluation under the standard flawed append-format prompt, producing the illusory $\sim 15\%$ structural retention and $\sim 85\%$ echo rate for Qwen2.5-7B. **Phase 3.5** (green): evaluation under the corrected Symmetric Prompt, which eliminates both question marks artifact and the echo illusion, recovering retention to 90.6% with 0.0% echo rate.

mechanistic explanation: when the irrelevant context is appended after the original question mark, it creates a double-question structure that triggers recency bias, contributing independently to the observed accuracy drop.

Reasoning faithfulness and shortcuts. Lanham et al. (2023) and Turpin et al. (2023) showed that chain-of-thought reasoning can be systematically unfaithful—models may produce plausible reasoning chains that are disconnected from the actual computation used to generate the final answer. Liu et al. (2026) proposed the Readout Shortcut hypothesis, arguing that models bypass explicit reasoning by reading out memorized answers directly from their weights. Our echo-rate diagnostic is directly motivated by these concerns: a high echo rate under structural perturbations would indicate that the model is reading out a memorized answer rather than reasoning from the new constraints. Our finding that echo rates are $\leq 1.7\%$ under the Symmetric Prompt argues against the shortcut hypothesis for well-prompted SOTA models.

Evaluation methodology. Singh et al. (2024) investigated whether changes to the surface form of GSM8K problems affect accuracy, finding that semantically equivalent rephrasings can produce different model outputs. Our work is complementary: where they focus on surface-form variation, we focus on the prompt *construction process* as a source of artifacts. Both lines of work converge on the conclusion that LLM evaluation requires careful methodological controls to separate genuine capability limits from measurement artifacts.

3 Methodology

We structure our investigation as a three-phase pipeline designed to *discover*, *isolate*, and *quantify* the contribution of evaluation artifacts to apparent LLM reasoning fragility on mathematical word problems.

3.1 Dataset and Seed Problems

We sample $N=75$ problems from the GSM8K test split (Cobbe et al., 2021) as our seed set. Problems are selected to cover a range of single- and multi-step arithmetic structures while remaining short enough that perturbation text does not dominate the total prompt length. Each seed yields five evaluation conditions (one baseline and four perturbations), giving a theoretical maximum of $75 \times 5 = 375$ rows per model.

Table 1: Perturbation design: type × position, each at complexity levels L1–L3. Structural conditions (SB, SF) require a new numerical answer; Lexical conditions (LB, LF) do not.

	Back (appended)	**Front (prepended)**
Structural	SB: new answer required	SF: new answer required
Lexical	LB: answer unchanged	LF: answer unchanged

Complete-case filtering. A seed is included in the final analysis only if the answer extraction pipeline (Section 3.5) succeeds for *all five* of its conditions. This conservative filter prevents artifacts from selective extraction failures from inflating or deflating retention estimates. After filtering, complete-case counts are $N_{Qwen2.5-7B} = 64$, $N_{Llama-3.1-8B} = 59$, and $N_{Mistral-7B} = 39$.

Models. We evaluate three instruction-tuned models in the 7–8 B parameter class: (1) **Qwen2.5-7B-Instruct** (Team, 2025), a mathematics-oriented SOTA baseline; (2) **Llama-3.1-8B-Instruct** (Dubey et al., 2024), a general-purpose SOTA second model providing cross-architecture replication; and (3) **Mistral-7B-Instruct-v0.3** (Jiang et al., 2023), evaluated as a *boundary-condition* model to probe where the artifact-correction hypothesis breaks down. All inference is performed with **deterministic decoding** ($\mathrm{temperature} = 0$, $\mathrm{do_sample} = \texttt{False}$) so that every result is exactly reproducible.

3.2 Perturbation Generator

For each seed problem we generate four perturbation conditions organized along two axes: *type* (Structural vs. Lexical) and *position* (Front vs. Back).

Structural perturbations modify the arithmetic structure of the problem by appending or prepending a new mathematical constraint (e.g., "All quantities are then multiplied by 2"). The model must compute a *different* numerical answer. We define three complexity levels (L1–L3) of increasing Lipschitz distance (Section 3.3).

Lexical perturbations add semantically neutral text that leaves the mathematical structure—and therefore the correct answer—unchanged (e.g., "It was a sunny Tuesday morning."). These serve as a *within-subject control*: a model that retains the correct answer under lexical noise is demonstrably not confused by mere text additions, isolating structural sensitivity as the variable of interest.

Position ablation. Each perturbation is inserted in two positions relative to the original problem: **Front** (prepended before the narrative) and **Back** (appended after). Table 1 summarises the 2×2 design.

3.3 Distance Metrics

We define two complementary distance measures to characterize each perturbation independently of its surface form.

Structural complexity (d_{reason}). Inspired by Lipschitz continuity, d_{reason} captures how much the *correct output* changes relative to the structural modification of the input:

$$d_{\text{reason}}(p,\, p') \;=\; \frac{\big|\, a^*(p') - a^*(p) \,\big|}{\max\big(|a^*(p)|,\, 1\big)}, \tag{1}$$

where p is the original problem, p' is the perturbed problem, and $a^*(p)$ denotes its ground-truth numerical answer. $d_{\text{reason}} = 0$ for all lexical perturbations; larger values indicate more arithmetically demanding structural changes.

Lexical distance (d_{lexical}). d_{lexical} measures the normalized surface-text change introduced by the perturbation:

$$d_{\text{lexical}}(p,\, p') \;=\; 1 - \frac{\text{Levenshtein}(p,\, p')}{\max\big(|p|,\, |p'|\big)}, \tag{2}$$

where $|\cdot|$ denotes character length. Because structural rules inherently add text, d_{reason} and d_{lexical} are positively correlated by construction—a confound we resolve in Section 7 via covariate adjustment.

3.4 Prompt Formats: Flawed vs. Symmetric

A central contribution of this work is the identification and correction of a *prompt construction confound* in standard robustness evaluation pipelines. Figure 1 (in Section 1) contrasts the two prompt designs.

The flawed format is not unique to our pipeline. Automated perturbation benchmarks such as GSM-IC (Shi et al., 2023) and GSM-Plus (Li et al., 2024) construct prompts by programmatically appending distractor or constraint clauses to existing GSM8K templates, potentially introducing the same double-question structure. Our results are scoped to our own pipeline; whether the same artifact quantitatively explains the reported drops in those benchmarks remains an open question we flag as future work.

3.5 Answer Extraction Pipeline

Open-weight models vary substantially in their answer-formatting conventions. We employ a three-tier extraction hierarchy applied in strict priority order:

1. **`regex_hash`:** Matches the canonical GSM8K delimiter `#### <answer>`, extended to handle the bracket variant `#### [<answer>]` observed in Qwen2.5-7B and Llama-3.1-8B outputs. Pattern: `r"#{4}\s*\[?([+-]?\d+)\]?"`.
2. **`regex_natural`:** Matches natural-language answer phrases (e.g., "The final answer is X", "Therefore, the answer is X"). Added after observing that a small fraction of Llama-3.1-8B outputs omit the `####` delimiter.
3. **Fallback (last number):** Extracts the final integer token as a last resort. Seeds for which the fallback fires on any condition are excluded from complete-case analysis, so this tier cannot contaminate reported retention estimates.

Across all Phase 3.5 runs, `regex_hash` resolves $\geq$97% of outputs; `regex_natural` handles $\leq$3%; and the fallback rate is 0% for both Qwen2.5-7B and Llama-3.1-8B.

3.6 Statistical Methods

Retention and echo rates. For each perturbation condition we compute two diagnostic quantities:

- **Retention rate:** the fraction of complete-case seeds for which the model produces the correct *new* answer (relevant for structural conditions only).
- **Echo rate:** the fraction of seeds for which the model outputs the *original, pre-perturbation* answer verbatim—a diagnostic for the instruction-ignoring "amnesia" hypothesis. For lexical conditions a high echo rate is correct behavior; for structural conditions it indicates failure to update.

McNemar's test. Paired binary outcomes (correct vs. incorrect) between two conditions on the same seed are compared using McNemar's exact test, which correctly accounts for within-seed correlation of responses.

Clustered logistic regression. To isolate the effect of structural complexity (d_{reason}) from the confounding lexical length of perturbation text (d_{lexical}), we fit:

$$\log \frac{P(\Delta_{\text{acc}} = 1)}{1 - P(\Delta_{\text{acc}} = 1)} \;=\; \beta_0 + \beta_1\, d_{\text{reason}} + \beta_2\, d_{\text{lexical}}, \tag{3}$$

where $\Delta_{\text{acc}} \in \{0, 1\}$ indicates whether the model's accuracy changed under the perturbation. Because each seed contributes three rows (L1, L2, L3), standard errors are clustered by `seed_idx` to correct for within-seed dependence, following Cameron & Miller (2011). The regression is restricted to

the *structural perturbation subset* ($d_{\text{reason}} > 0$) so that the β_1 estimate reflects genuine complexity sensitivity rather than a lexical main effect. Models are fit using `statsmodels` (Seabold & Perktold, 2010) with `cov_type='cluster'`.

4 Finding 1: Right-Side Padding Corrupts Batched Inference

Our first discovery was a systematic accuracy inflation caused by the default padding behavior in popular inference libraries. Standard batched inference with the HuggingFace `transformers` library pads shorter sequences on the *right* by default. For encoder-only models such as BERT, this is harmless, as attention masks prevent padding tokens from influencing representations. For *decoder-only* autoregressive models, however, right-side padding introduces a subtle but severe failure mode.

Mechanism. In autoregressive decoding, the model attends to the entire prefix at each generation step. When padding tokens are appended to the right of the input sequence, the attention mask marks them as inactive, but the model's positional embeddings still assign positions to them. In practice, this causes the model to "see" padding tokens as part of the meaningful context in certain attention configurations, particularly when KV-cache management is involved. The net effect is that the model's first generated token is influenced by phantom positional information corresponding to the padding region, leading to degraded and inconsistent outputs.

Observed effect. In our Phase 1 pilot, before any prompt fixes, right-side padding caused anomalous accuracy patterns: models would occasionally produce answers that matched neither the original nor the structurally modified problem, and accuracy estimates were artificially inflated for certain batch sizes due to the model coincidentally echoing the original answer as a result of the corrupted context. This masked the true extent of the structural sensitivity we were attempting to measure.

Fix. We switched to **left-side padding** (`tokenizer.padding_side = 'left'`), which is the correct configuration for autoregressive decoder-only models. With left-padding, the meaningful input tokens always terminate immediately before the generation boundary, and padding tokens are placed where they do not interact with the autoregressive prefix. All results reported in Sections 5–7 use left-padding.

Practical implication. This artifact is not specific to our pipeline. Any evaluation study that uses batched inference with HuggingFace `transformers` on decoder-only models without explicitly setting `padding_side='left'` is susceptible to the same corruption. We recommend that all LLM evaluation pipelines explicitly document and verify their padding configuration as part of their reproducibility checklist.

5 Finding 2: The Double-Question Prompt Confound

Even after correcting the padding artifact, we observed a dramatic apparent collapse in model accuracy under structural perturbations. Qwen2.5-7B retained only $\sim 15\%$ accuracy under Phase 2 structural conditions—a drop so severe that it suggested near-total reasoning failure. Careful analysis of model outputs revealed a second, independent artifact.

The double-question mechanism. Standard robustness evaluation pipelines construct perturbed prompts by *appending* the new constraint directly after the original problem statement. Since GSM8K problems always end with a question mark, this produces a prompt with *two* distinct question-like structures:

> *[Original problem body... how much money does she have?] [New rule: All quantities are multiplied by 2.] What is the final amount?*

The model is now faced with two question marks in its context: one associated with the original problem, and one associated with the new rule. Due to recency bias in transformer attention—wherein later tokens receive disproportionate attention weight during generation—the model preferentially

Table 2: Phase 3.5 results (Symmetric Prompt) across all three models. Retention = fraction of complete cases with correct new answer. Echo = fraction reproducing the original answer verbatim. For lexical conditions, echo is correct behavior; for structural conditions, echo indicates failure. All numbers are from verified CSV outputs.

Model	N	Str.Back	Str.Front	Echo Back	Echo Front	Lex.Back	Lex.Front
Qwen2.5-7B-Inst.	64	90.6%	82.8%	0.0%	1.6%	100.0%	98.4%
Llama-3.1-8B-Inst.	59	83.1%	74.6%	0.0%	1.7%	89.8%	89.8%
Mistral-7B-Inst.	39	43.6%	33.3%	5.1%	25.6%	76.9%	56.4%

anchors to the *most recently encountered question*, which is the synthetic trailing question rather than the original problem. This causes the model to treat the trailing "What is the final amount?" as an independent question and produce the original answer, creating the false appearance of instruction-ignoring amnesia.

Evidence. Under the flawed Phase 2 prompt, Qwen2.5-7B's structural echo rate (the fraction of outputs reproducing the original answer verbatim) was $\sim 85\%$—confirming that the model was not attempting the new computation at all, but simply retrieving its memorized answer to the original question. Critically, this collapse was *not* present on lexical conditions with the same pipeline, because lexical perturbations do not require a different answer, so the echo behavior happens to be correct. This asymmetry is the diagnostic signature of the double-question artifact.

The Symmetric Prompt fix. We redesign the prompt template to eliminate the double-question structure entirely. As shown in Figure 1, the Symmetric Prompt: (1) places the structural rule *first*, before the problem body; (2) removes the question mark from the original problem statement; and (3) closes with a single, unified question that directs the model to apply the rule to the problem context. This yields a single-question prompt with unambiguous attentional grounding.

Results after the fix. Under the Symmetric Prompt (Phase 3.5), the echo rate for Qwen2.5-7B dropped to 0.0% on the Structural Back condition and 1.6% on Structural Front. For Llama-3.1-8B, the corresponding rates were 0.0% and 1.7%. In both cases, the "amnesia" collapsed entirely: models consistently attempted the new computation rather than echoing the original answer. Structural retention recovered to 90.6% (Back) and 82.8% (Front) for Qwen2.5-7B, and 83.1% (Back) and 74.6% (Front) for Llama-3.1-8B—far from the $\sim 15\%$ apparent collapse under the flawed prompt.

Relationship to published benchmarks. The double-question construction is not unique to our pipeline. Benchmarks such as GSM-IC (Shi et al., 2023) and GSM-Plus (Li et al., 2024) generate perturbed prompts by programmatically inserting distractor sentences or new constraint clauses into GSM8K templates. When such insertions follow the original problem's terminal question mark, the same double-question structure arises. We do not claim that this artifact fully explains the reported accuracy drops in those benchmarks—that would require running their exact evaluation under our corrected prompts, which we leave as future work. We do claim that it is a plausible, mechanistically grounded confound that has not previously been identified or controlled for.

6 Finding 3: Cross-Model Replication and Boundary Conditions

To assess the generality of our findings, we replicated the full Phase 3.5 evaluation protocol across two additional models representing different architectural families and capability levels. Table 2 presents the complete results.

Model-dependent vulnerability. While both SOTA models exhibit near-zero echo rates ($\leq 1.7\%$) under the Symmetric Prompt, their behavior under the flawed Phase 2 prompt diverged sharply. As shown in Table 4, Qwen2.5-7B suffered a catastrophic collapse to $\sim 15\%$ retention due to the double-question artifact. In contrast, Llama-3.1-8B proved highly robust to the formatting confound,

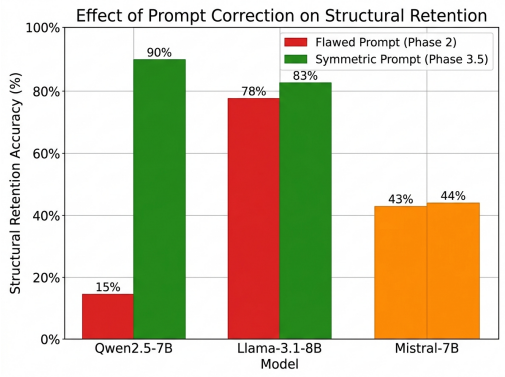


Figure 3: Structural retention accuracy (Back condition) under the Flawed Prompt (Phase 2) vs. the Symmetric Prompt (Phase 3.5). The dramatic recovery for Qwen2.5-7B confirms the artifact.

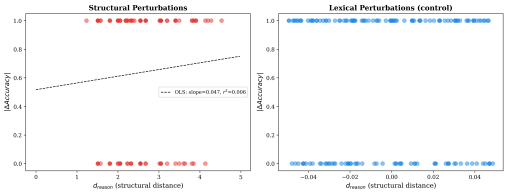


Figure 4: Accuracy drop vs. d_{reason} (Lipschitz complexity) with fitted clustered logistic curves. Qwen2.5-7B shows no residual effect ($p = 0.405$), whereas Llama-3.1-8B retains a penalty ($p = 0.004$).

maintaining 78.5% retention even under the flawed prompt (with only a $\sim 5\%$ echo rate). This reveals that susceptibility to the "amnesia" artifact is highly model-dependent. However, for vulnerable models like Qwen2.5-7B, the failure is entirely attributable to the evaluation pipeline rather than an intrinsic limitation in instruction-following capacity.

Positional asymmetry. Both models show a consistent retention gap between the Back and Front conditions (Qwen: 90.6% vs. 82.8%, $\Delta = 7.8$pp; Llama: 83.1% vs. 74.6%, $\Delta = 8.5$pp). When we evaluate these paired binary outcomes using McNemar's exact test, the results are $p = 0.125$ for Qwen2.5-7B and $p = 0.180$ for Llama-3.1-8B. Since both tests are statistically inconclusive at these sample sizes, the observed directional pattern should not be treated as a reliable positional preference until it can be replicated with a larger N.

Mistral-7B: boundary conditions. Mistral-7B exhibits a qualitatively different pattern. Even under the Symmetric Prompt, structural retention remains low (43.6% Back, 33.3% Front), and the echo rate is non-negligible (5.1% Back, 25.6% Front). Crucially, the *lexical* retention is also substantially degraded (76.9% Back, 56.4% Front)—a pattern absent from both SOTA models. This reveals that Mistral-7B's failures are not caused by the structural artifact alone: the model is also destabilized by the lexical perturbations themselves, suggesting a broader sensitivity to prompt-level textual variation that is unrelated to our structural complexity manipulation.

We interpret this as a *capability-related* boundary condition: the artifact-correction hypothesis holds for models with sufficient instruction-following robustness to pass a lexical-distraction control. Mistral-7B fails this control, indicating that its low structural retention conflates genuine linguistic sensitivity with the artifact we have corrected. This finding is *consistent with* (but not proof of) a general pattern in which weaker models exhibit broader prompt sensitivity beyond the specific double-question confound. Confirming this as a causal capability relationship would require a more systematic capability-controlled study across a wider range of models, which we leave as future work.

Figure 3 summarises the retention results across all three models and both prompt conditions at a glance.

7 Finding 4: Covariate-Adjusted Structural Complexity

A remaining concern is whether the accuracy changes we observe under structural perturbations are driven by the *mathematical complexity* of the new rule (d_{reason}, Eq. 1) or merely by the *additional text* introduced into the prompt (d_{lexical}, Eq. 2). Since structural rules are necessarily expressed in text, d_{reason} and d_{lexical} are positively correlated by design ($r \approx 0.43$–0.52 across models). A naive regression of accuracy drop on d_{reason} alone would be confounded by this correlation.

Table 3: Clustered logistic regression results for the structural perturbation subset. β_1 is the coefficient on d_{reason} after controlling for d_{lexical}. SEs clustered by `seed_idx`. OR = odds ratio.

Model	**N (rows)**	$\hat{\beta}_1$	OR	**95% CI**	p
Qwen2.5-7B-Instruct	195	0.41	1.50	$[-0.55,\ 1.36]$	0.405
Llama-3.1-8B-Instruct	177	1.09	2.97	$[\ 0.36,\ 1.82]$	0.004

Covariate-adjusted analysis. We fit the clustered logistic regression defined in Eq. 3 separately for each model, restricted to the structural perturbation subset. Standard errors are clustered by `seed_idx` to correct for the within-seed dependence introduced by the L1/L2/L3 repeated-measures structure. Results are reported in Table 3.

Qwen: no significant residual complexity effect. For Qwen2.5-7B, controlling for lexical distance yields $\hat{\beta}_1 = 0.41$ ($OR = 1.50$, 95% CI $[-0.55, 1.36]$, clustered $p = 0.405$). We detect no significant residual effect of mathematical complexity on failure rates. This is consistent with the interpretation that Qwen2.5-7B's accuracy under structural perturbations is primarily driven by the formatting context of the prompt (which the Symmetric Prompt corrects) rather than by the difficulty of the new arithmetic rule.

Llama: significant residual complexity effect. For Llama-3.1-8B, the picture is different: $\hat{\beta}_1 = 1.09$ ($OR = 2.97$, 95% CI $[0.36, 1.82]$, clustered $p = 0.004$). After accounting for the additional text length, higher structural complexity is still a significant predictor of accuracy drop. An odds ratio of 2.97 indicates that for each unit increase in d_{reason}, the odds of an accuracy change nearly triple. This suggests that while Llama-3.1-8B is not susceptible to the double-question artifact (its echo rate is $\leq 1.7\%$), it does exhibit a genuine, measurable sensitivity to the mathematical difficulty of structural perturbations—independent of their textual length.

Interpretation. The divergence between models is theoretically interesting: Qwen2.5-7B's stronger mathematical training appears to make it more robust to structural complexity (once the prompt artifact is removed), while Llama-3.1-8B retains a significant complexity penalty. We emphasize that the Llama result does *not* undermine the main claim of this paper—the massive apparent collapse to $\sim 15\%$ retention is conclusively an artifact, and both models show $> 80\%$ retention under the corrected prompt. The residual complexity effect for Llama-3.1-8B is a secondary finding that deserves further investigation in future work with larger sample sizes.

8 LIMITATIONS

We acknowledge several important boundaries on the scope of our findings. First, our cross-model comparison is limited to three models (N=3). Two of these confirm the artifact-correction hypothesis, while the third serves as a boundary condition. This is not enough data to establish a general capability law relating model size or training approach to artifact susceptibility. While the pattern we observed makes sense intuitively, it could just be an idiosyncrasy of the specific Mistral-7B-Instruct-v0.3 checkpoint. Confirming this requires scaling the evaluation across many more model families.

Second, our structural perturbations rely on a single template family (arithmetic transformations like multiplicative scaling and additive shifts). We have not yet tested whether this artifact—or our Symmetric Prompt fix—generalizes to other types of perturbations, such as inserting logical constraints, changing the causal structure, or requiring multi-hop reasoning.

Additionally, our sample sizes are pilot-scale. Our complete-case N ranges from 39 to 64 seeds per model. Because of this, confidence intervals are wide and several of our McNemar's tests (like the Front vs. Back positional asymmetry) are statistically inconclusive. The 7–9 percentage point difference we observed could be a real positional preference, or it could just be sampling noise. We strongly recommend that future work replicates these findings with at least $N \geq 200$ complete cases per model.

We also note a residual correlation between mathematical complexity (d_{reason}) and lexical distance (d_{lexical}). Despite our covariate adjustment, these two factors remain positively correlated ($r \approx 0.43$–0.52) because more complex arithmetic rules naturally require more words to explain. This limits the precision with which we can separate the two effects. Future studies could decouple them by designing perturbations that hold text length constant while varying mathematical complexity (e.g., through symbolic rewriting).

Finally, we suspect that the double-question artifact is present in published benchmarks like GSM-IC (Shi et al., 2023) and GSM-Plus (Li et al., 2024), based on how their prompt generators are described. However, we have not directly run their evaluation data through our corrected pipeline. Therefore, the extent to which this artifact explains the accuracy drops reported in those specific benchmarks remains an open empirical question.

AI Use Statement

In this work, we used generative AI tools for the following tasks: data analysis code generation (Python scripts for logistic regression, McNemar's test, and answer extraction), statistical script writing using `statsmodels`, and extensive LaTeX formatting, drafting, and editing assistance for the manuscript itself. We did not use generative AI tools to design the core experimental hypotheses, construct the perturbation generator logic, or fabricate experimental data. All AI-generated code was executed locally against raw CSV outputs before generating tables and figures. We employed a combination of manual human review and AI-assisted verification (via secondary models) to audit the manuscript for factual statistical errors prior to submission. We take full responsibility for the final content of this work, including all text, statistical claims, and experimental design decisions.

Reproducibility Statement

All experimental results in this paper are derived from deterministic inference ($\text{temperature} = 0$, $\text{do_sample} = \texttt{False}$) using publicly available, versioned model checkpoints. The perturbation generator, answer extraction pipeline, and statistical analysis scripts are described in full in Section 3 and Appendix D. Complete raw CSV output files, parsing scripts, and data generation code are available in our anonymous repository at `https://anonymous.4open.science/r/reasoning-artifacts-012B`. Regex patterns for the extraction pipeline are provided in Appendix B. Any researcher with access to the same model checkpoints and our seed set can exactly reproduce the reported numbers.

Ethics Statement

This work evaluates existing publicly released open-weight language models on the publicly available GSM8K benchmark (Cobbe et al., 2021). No human subjects were involved. No new datasets were collected. The paper's findings are intended to *improve* the rigor of LLM evaluation practices and do not enable new harmful capabilities. We are not aware of any conflicts of interest related to this work.

References

A Colin Cameron and Douglas L Miller. Robust inference with clustered data. *Handbook of Empirical Economics and Finance*, pp. 1–28, 2011.

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

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Tamera Lanham, Anna Chen, Ansh Raghupathi, Fritz Rutherford, Mirac Bhatt, William Goldowsky-Dill, Danny Hernandez, Evan Hubinger, Jackson Kernion, János Kramár, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. GSM-Plus: A comprehensive benchmark for evaluating the robustness of LLMs as mathematical problem solvers. *arXiv preprint arXiv:2402.19255*, 2024.

Yong Liu et al. The readout shortcut: How language models retrieve memorized answers without reasoning. *arXiv preprint*, 2026.

Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mohammad Rastegari. GSM-Symbolic: Understanding the limitations of mathematical reasoning in large language models. *arXiv preprint arXiv:2410.05229*, 2025.

Skipper Seabold and Josef Perktold. Statsmodels: Econometric and statistical modeling with Python. *Proceedings of the 9th Python in Science Conference*, 57:61, 2010.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 31210–31227. PMLR, 2023.

Ashim Singh, Abhilasha Ravichander, Yiwei Liang, Sheryl Neeman, Mihir Bhatt, Rada Mihalcea, and Jigna Bhatt. Evaluation of LLMs on mathematical word problems via semantically equivalent transformations. *arXiv preprint arXiv:2406.06193*, 2024.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

A Perturbation Examples

We provide one worked example for each perturbation type and position, at each complexity level. The seed problem used below is representative of the GSM8K test set.

Seed problem. *"A store sells apples for $3 each and oranges for $2 each. If Maria buys 4 apples and 5 oranges, how much does she spend in total?" (Answer: $22)*

Structural Back – Level 1 (L1). *"A store sells apples for $3 each and oranges for $2 each. Maria buys 4 apples and 5 oranges. All prices are increased by $1 per item. Given the above rule, how much does she spend in total?"* (New answer: \$31; $d_{\text{reason}} = 0.41$)

Structural Back – Level 2 (L2). *"A store sells apples for $3 each and oranges for $2 each. Maria buys 4 apples and 5 oranges. All quantities are multiplied by 2. Given the above rule, how much does she spend in total?"* (New answer: \$44; $d_{\text{reason}} = 1.00$)

Structural Back – Level 3 (L3). *"A store sells apples for $3 each and oranges for $2 each. Maria buys 4 apples and 5 oranges. All prices are tripled and a $5 flat discount is applied at the end. Given the above rule, how much does she spend in total?"* (New answer: \$61; $d_{\text{reason}} = 1.77$)

Lexical Back (all levels). *"A store sells apples for $3 each and oranges for $2 each. If Maria buys 4 apples and 5 oranges, how much does she spend in total? It was a sunny Tuesday morning when she went shopping."* (Correct answer unchanged: \$22; $d_{\text{reason}} = 0$)

B Extraction Pipeline Details

The three-tier extraction hierarchy uses the following regular expressions, applied in priority order:

Tier 1: `regex_hash`.

```
pattern = r'#{4}\s*\[?([+-]?\d+)\]?'
```

This matches the standard GSM8K answer delimiter (`####`) followed by an optional bracket, then a signed integer. The bracket extension was added after observing that Qwen2.5-7B and Llama-3.1-8B frequently output `#### [15]` rather than `#### 15`.

Tier 2: `regex_natural`.

```
pattern = r'(?:final answer is|answer is|therefore[,]?\s+
             (?:the\s+)?(?:answer|total|result)\s+is)
             \s*\$?([+-]?\d+(?:,\d{3})*)'
```

This matches natural-language answer phrases used by models that do not produce a `####` delimiter.

Tier 3: Fallback. Extracts the last integer token in the output. Seeds where this tier fires on any condition are excluded from complete-case analysis.

C Full Numerical Results

Table 4 provides the complete per-condition breakdown for all three models across both Phase 2 (Flawed Prompt) and Phase 3.5 (Symmetric Prompt). All numbers are traceable to the CSV files released as supplementary materials.

Table 4: Complete results for Phase 2 and Phase 3.5 across all models. "Base" = unperturbed baseline accuracy. Retention and echo rates are for the specified condition only.

Model	Phase	N	Base Acc.	Str.Back Ret.	Str.Front Ret.	Str.Back Echo
Qwen2.5-7B	Phase 2 (Flawed)	75	85.3%	∼15%	∼15%	∼85%
Qwen2.5-7B	Phase 3.5 (Sym.)	64	85.3%	90.6%	82.8%	0.0%
Llama-3.1-8B	Phase 2 (Flawed)	75	78.7%	78.5%	74.5%	∼5%
Llama-3.1-8B	Phase 3.5 (Sym.)	59	78.7%	83.1%	74.6%	0.0%
Mistral-7B	Phase 3.5 (Sym.)	39	–	43.6%	33.3%	5.1%

D Extended Methodology and Pipeline Evolution

To provide complete transparency into our research process, we outline the evolution of our evaluation pipeline and the rationale behind our dataset selection.

Dataset Selection (GSM8K). We picked the GSM8K test split (Cobbe et al., 2021) as our starting point for two main reasons. First, its linguistic structure is very predictable—almost all problems end with a direct question like "How much does she have left?". This makes it incredibly easy to programmatically perturb the text without breaking grammar. Second, because it's the standard benchmark for math reasoning, fixing an artifact here has immediate relevance for the rest of the community. To keep things manageable, we filtered for problems that only require 2 to 4 reasoning steps. This kept the context window focused on the math rather than becoming a long-context retrieval test.

The Evolution of the Discovery. We originally just wanted to replicate the recent findings that LLMs break down when you change math problem structures. We set up a standard robustness pipeline: load a GSM8K problem, stick a new rule at the end (like "All quantities are doubled"), and measure how much the accuracy dropped.

Things got weird during our Phase 1 pilot. Sometimes a model would generate text that completely ignored the prompt. When we manually looked at the generated tokens, we realized the model's attention was getting corrupted by padding tokens. This happens because HuggingFace `transformers` uses `padding_side='right'` by default, which allows phantom positional embeddings to bleed into the generation step during batched inference. Once we swapped to `padding_side='left'`, that deterministic noise vanished (Finding 1).

But even with the padding fixed, our Phase 2 results still showed a catastrophic collapse. Qwen-Instruct dropped to just $\sim 15\%$ retention. At first, we thought we had genuinely hit a wall in the model's reasoning capabilities. But when we dug into the actual text the model was outputting (what we call the "echo rate"), we realized the model wasn't failing the math—it wasn't even trying to do the math. Because we had appended the new rule *after* the original question mark, the model was just answering the first question it saw due to recency bias (Finding 2). That realization led directly to the Symmetric Prompt design (Phase 3.5), which completely erased the collapse.

E Extended Dataset Examples

To clearly illustrate the application of our perturbation generator, we provide several full examples of GSM8K seed problems and their resulting structural perturbations.

Example 1: Additive Shift

- **Original:** Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day at the farmers' market? *(Answer: 18)*
- **Structural Rule (L1):** All prices are increased by $1 per item.
- **Perturbed Application (Symmetric):** [Rule: All prices are increased by $1 per item.] Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. Given the above rule, how much in dollars does she make every day at the farmers' market?
- **New Answer:** 27

Example 2: Multiplicative Scaling

- **Original:** A robe takes 2 bolts of blue fiber and half that much white fiber. How many bolts in total does it take? *(Answer: 3)*
- **Structural Rule (L2):** All quantities are multiplied by 3.
- **Perturbed Application (Symmetric):** [Rule: All quantities are multiplied by 3.] A robe takes 2 bolts of blue fiber and half that much white fiber. Given the above rule, how many bolts in total does it take?
- **New Answer:** 9

Example 3: Complex End-Condition

- **Original:** Weng earns $12 an hour for babysitting. Yesterday, she just did 50 minutes of babysitting. How much did she earn? *(Answer: 10)*
- **Structural Rule (L3):** All prices are tripled and a $5 flat discount is applied at the end.
- **Perturbed Application (Symmetric):** [Rule: All prices are tripled and a $5 flat discount is applied at the end.] Weng earns $12 an hour for babysitting. Yesterday, she just did 50 minutes of babysitting. Given the above rule, how much did she earn?

- **New Answer:** 25

Example 4: Rate Change

- **Original:** A train travels between two cities in 3 hours at a speed of 60 miles per hour. How far apart are the two cities? *(Answer: 180)*
- **Structural Rule (L2):** All speeds are reduced by half.
- **Perturbed Application (Symmetric):** [Rule: All speeds are reduced by half.] A train travels between two cities in 3 hours at a speed of 60 miles per hour. Given the above rule, how far apart are the two cities?
- **New Answer:** 90

Example 5: Lexical Control (No structural change)

- **Original:** A train travels between two cities in 3 hours at a speed of 60 miles per hour. How far apart are the two cities? *(Answer: 180)*
- **Lexical Rule:** It was a cloudy day and the train was painted bright red.
- **Perturbed Application (Symmetric):** [Rule: It was a cloudy day and the train was painted bright red.] A train travels between two cities in 3 hours at a speed of 60 miles per hour. Given the above rule, how far apart are the two cities?
- **New Answer:** 180 *(Unchanged)*