Answer-Boundary Extraction Can Change Measured Prompt Effects: Sequential Factorial Evidence from ASDiv and DROP

Ming-Hao Wang[1], **Hung-Shin Lee**[2]

[1]National Taichung University of Science and Technology
[2]National Taiwan Normal University

Abstract

Prompt-transformation studies usually attribute benchmark changes to the prompt, but the reported score is produced jointly by the prompt, the model, and an answer extractor. We ask whether a measured prompt effect survives a change of extraction rule alone. Holding stored outputs fixed, we rescore a discovery-stage 2^4 factorial on ASDiv and a prospectively specified DROP study, with seven instruction-tuned checkpoints, using two non-nested extractors. Permitting post-answer text changes the extractor-specific prompt contrast by +41.68 pp on ASDiv and +19.72 pp on DROP; these are differences between prompt effects, not accuracy gains. The rule change can reverse an effect's sign: for Granite-2B on ASDiv, the terminality effect is +46.69 pp under a last-marker extractor and -12.06 pp under a final-line extractor. The interaction is driven mainly by final-boundary failures and is heterogeneous across checkpoints. An exploratory context-aware LLM judge did not remove this dependence and sometimes replaced a response's answer with the reference. We found no statistically significant boilerplate-by-extractor interaction on either task, and other secondary factors are less stable. Both frozen cap gates fail; mandatory common-ID analyses preserve the positive panel averages without repairing the primary runs. Extractor-sensitive effects should be attributed to the prompt–model–extractor pipeline, not to the prompt alone.

Keywords: Large Language Model Evaluation, Answer Extraction, Prompt Transformation, Response Boundaries

1 Introduction

Prompt studies typically attribute benchmark score differences to changes in model input. Yet when a task requires a canonical number, span, class, or structured object, the score is produced by a pipeline comprising the prompt, model, and answer extractor. Because the model generates text while the benchmark scores an object, extraction is part of measurement rather than neutral post-processing.

Consider `Answer: 7` followed by another line of explanation. A last-marker extractor returns 7, whereas a final-line extractor abstains because the marked answer is not on the last nonempty line. Conversely, if `Answer: 6` is followed by a bare final line 7, the final-line rule returns 7 and the last-marker rule returns 6. The rules encode distinct boundaries, and neither extractor's correctness set necessarily contains the other's.

Prompt studies estimate a *contrast* between transformed and reference conditions. It is unchanged when extraction affects both conditions equally, but changes when a manipulation shifts response form at the extraction boundary; the effect then belongs to the prompt–model–extractor pipeline, not the transformation alone.

Prior work shows sensitivity to prompt templates and output formats (Mizrahi et al., 2024; Zhuo et al., 2024; Long et al., 2025) and to metric or extraction choices (Chen et al., 2019; Molfese et al., 2025), but primarily compares absolute scores. We ask whether an estimated prompt effect changes under alternative extractors: whether the interaction persists for a non-nested pair, concentrates in answer-boundary factors rather than arbitrary edits, and transfers from short arithmetic problems to a different reasoning task under a specification fixed before target generation.

We conduct two sequential 2^4 factorial studies using the same seven 2B–8B instruction-tuned checkpoints. The ASDiv discovery study crosses boilerplate presence, marker alias, terminality, and unit policy on 200 items, with additional controls and a limited BF16 diagnostic. We then fixed the corresponding estimand, prompts, extractors, checkpoint revisions, and analysis for 200 unani-

Proceedings of the 38th Conference on Computational Linguistics and Speech Processing (ROCLING 2026), pages 25–35
November 6-7, 2026 ©2026 Association for Computational Linguistics

mously numeric DROP validation items (Dua et al., 2019). No DROP outputs preceded the external timestamps, and both extractors rescore identical stored responses.

With outputs fixed and only the extraction rule varied, allowing post-answer text yields extractor interactions of +41.68 pp on ASDiv and +19.72 pp on DROP, with positive panel averages under common-ID and checkpoint-resampling sensitivities. The rule can even reverse a checkpoint's estimated prompt effect (Granite-2B on ASDiv: +46.69 vs. -12.06 pp), and Qwen-3B reverses the sign of the interaction on both tasks. In post hoc checks, a re-implemented lm-evaluation-harness flexible filter shows the same direction, and the tested final-line relaxation changes the interaction substantially more than conservative Markdown normalization. A context-aware LLM judge reproduced most of the interaction but sometimes returned the reference answer instead of the response's own. We found no statistically significant boilerplate-by-extractor interaction on either task, and marker-alias and unit-policy results are less stable. The evidence supports a boundary-specific mechanism, not a universal coefficient or a general claim about prompt compression.

Our contributions are:

- We formalize extractor dependence as a paired factorial interaction and operationalize it with two non-nested answer-boundary rules, so the estimated effect can take either sign.
- We conduct sequential studies on ASDiv and DROP across seven fixed checkpoints, separating mechanism discovery from a prospectively specified cross-task test while rescoring identical stored outputs.
- We show that changing only the extraction rule can change the size and even the sign of a measured prompt effect; the interaction transfers across tasks, varies by task and checkpoint, is driven mainly by final-boundary failures, and reverses sign for Qwen-3B.
- We provide an auditable reporting protocol that discloses executable extraction, checkpoint heterogeneity, failed cap gates, and common-ID sensitivities without presenting them as repaired runs.

2 Related Work

Prompt and context transformation. Prompt-compression methods prune or rewrite surface content or learn compact representations. Selective Context removes low-information units (Li et al., 2023); LLMLingua performs budgeted token pruning (Jiang et al., 2023b); LLMLingua-2 learns token classification from distilled data (Pan et al., 2024); and LongLLMLingua adds question-aware compression for long contexts (Jiang et al., 2024). AutoCompressors and gist tokens instead encode context into learned representations (Chevalier et al., 2023; Mu et al., 2023). Rate-distortion and information-preservation analyses show that compression rate alone does not determine retained task information (Nagle et al., 2024; Łajewska et al., 2025). We examine a downstream measurement problem rather than compare compressors.

Prompt sensitivity and output format. Scores and model rankings vary across prompt templates (Mizrahi et al., 2024; Zhuo et al., 2024), and some apparent sensitivity may be induced by evaluation design (Hua et al., 2025). Format compliance is also distinct from semantic correctness (Long et al., 2025), while informationally similar output formats can yield substantial measured differences (Ravikumar et al., 2026). This work establishes that prompts and interfaces affect benchmark outcomes, but does not isolate whether a prompt *contrast* changes with the answer boundary used for scoring.

Scoring and answer extraction. Metric and extraction choices can alter question-answering and multiple-choice results (Chen et al., 2019; Molfese et al., 2025). The EleutherAI lm-evaluation-harness GSM8K configuration provides strict and flexible regex filters (EleutherAI, 2026). Our two primary rules are author-designed and not equivalent to that harness; Section 5.5 therefore adds an exploratory re-implementation of its flexible filter. Prior studies primarily compare absolute scores across evaluation procedures. We instead rescore identical outputs with two non-nested boundary rules and estimate a paired extractor-by-treatment interaction. This design separates baseline extractor disagreement from extractor dependence in the estimated prompt effect, without assuming that either rule is normatively correct.

Sequential evaluation. Repeated benchmark inspection can make later specifications adaptive (Dwork et al., 2015). Earlier ASDiv analyses informed our hypothesis, so V10 is discovery. We then froze the DROP subset, factorial, extractor

Threat to interpretation	Design response	Observable requirement
Nested rules predetermine the sign	Use two non-nested answer-boundary extractors	Checkpoint interactions can be positive or negative
Rule brittleness drives the effect	Rescore with harness, Markdown-normalized, and LLM-judge scorers (post hoc)	The interaction persists under alternative scorers
One edited prompt manufactures the result	Cross four prompt factors independently	Effects concentrate in boundary-relevant factors
One task or checkpoint drives the headline	Run sequential ASDiv and DROP studies on seven fixed checkpoints	Report cross-task magnitudes, reversals, and leave-one-out estimates
Length caps create a false boundary effect	Preserve failed audits and require a common-ID analysis	The primary direction persists after removing cap-affected IDs

Table 1: Identification logic. Each interpretive threat has a corresponding design response and observable criterion; no criterion is sufficient alone.

ordering, estimand, multiplicity family, and analysis before V11 generation. This chronology limits outcome-adaptive specification, but does not make the purposive checkpoint panel representative or establish prevalence beyond the tested tasks.

3 Estimand and Answer Extractors

For item x_i with canonical gold number y_i, checkpoint m, prompt condition c, and extractor e, define

$$\begin{aligned} R_{imc} &= G_m(c(x_i)), \\ \hat{y}_{imce} &= e(R_{imc}), \\ S_{imce} &= \mathbf{1}[\hat{y}_{imce} = y_i], \end{aligned} \tag{1}$$

where G_m includes the pinned tokenizer, checkpoint, quantization, and greedy generation protocol. Each extractor returns a canonical number or abstains; abstentions score as incorrect. All extractor comparisons reuse the same stored response R_{imc}.

For a binary prompt factor f, let $\mathcal{C}_{f=\ell}$ denote the eight factorial cells with level $\ell \in \{0, 1\}$. The item–checkpoint main effect under extractor e is

$$D_{imfe} = \frac{1}{8} \sum_{c \in \mathcal{C}_{f=1}} S_{imce} - \frac{1}{8} \sum_{c \in \mathcal{C}_{f=0}} S_{imce}. \tag{2}$$

We define the paired extractor interaction and fixed-panel estimator as

$$\begin{aligned} J_{imf} &= D_{imf,\textsc{LastMarker}} - D_{imf,\textsc{FinalLine}}, \\ \widehat{J}_f &= \frac{1}{|\mathcal{I}||\mathcal{M}|} \sum_{i \in \mathcal{I}} \sum_{m \in \mathcal{M}} J_{imf}, \end{aligned} \tag{3}$$

where $\mathcal{I}$ is the item set and $\mathcal{M}$ is the fixed seven-checkpoint panel. We report $100\widehat{J}_f$ in percentage points. The primary factor is terminality, with the ordered contrast LASTMARKER minus FINALLINE.

A positive interaction means that the level-1 versus level-0 prompt effect is larger under LASTMARKER than under FINALLINE; it does not imply that level 1 improves absolute accuracy under either extractor.

LASTMARKER. Return the canonical number following the final case-insensitive `Answer:` or `Final answer:` marker anywhere in the response; otherwise abstain.

FINALLINE. Inspect only the last nonempty line. Return a canonical number if the entire line is a bare number or a number optionally preceded by an accepted marker and optionally followed by units; otherwise abstain.

Both rules use the same numeric normalization. Neither contains the other: LASTMARKER targets recoverable marked content, whereas FINALLINE targets the numeric assertion at the response boundary. We do not designate either construct as universally correct.

4 Sequential Experimental Design

We use V10 and V11 as protocol labels for the ASDiv discovery and DROP prospective stages; the released frozen artifact versions are V10.2 and V11.1.

4.1 Shared Factorial and Checkpoint Panel

The factors were crossed independently. The exact prompt builder makes each manipulation concrete. For the selected ASDiv item, boilerplate level 0 contains the sentences `Solve the following grade-school math problem.` and `Read the following grade-school math problem carefully.`; level 1 deletes only the

Factor	Level 0	Level 1
Boilerplate	present	deleted
Marker	`Answer:`	`Final answer:`
Terminality	terminal answer	trailing text allowed
Units	numeric only	units allowed

Table 2: The shared 2^4 prompt factorial. Each main effect contrasts the averages of the eight cells at the two levels of the focal factor.

first sentence. The marker changes from `Answer:` to `Final answer:`. Terminality level 0 requires the response to end with `Answer: <number>`; level 1 requests `Answer: <number>` but explicitly permits additional explanation after the answer line. The unit instruction changes from `a number without units` to `a number; units are permitted`. These are prompt instructions, not extractor settings; the extractors are held fixed while the prompt factors vary. The released prompt builder and frozen inputs permit reconstruction of the full prompts.

Both studies used Mistral-7B-Instruct-v0.2 (Jiang et al., 2023a; Mistral AI, 2023); Qwen2.5-7B-Instruct and Qwen2.5-3B-Instruct (Qwen Team, 2024a,b); Phi-3.5-mini-instruct (Microsoft, 2024); Granite-3.3-8B-Instruct and Granite-3.3-2B-Instruct (IBM Granite Team, 2025); and OLMo-2-1124-7B-Instruct (OLMo Team, 2025). The panel is purposive and supports heterogeneity analysis, not model-population inference.

4.2 Stage 1: ASDiv Mechanism Discovery

V10 selected 200 ASDiv-A items deterministically (Miao et al., 2020). Its 16 factorial conditions and four mechanism controls yielded 28,000 NF4 rows. A limited 1,600-row BF16 diagnostic covered Mistral-7B and Qwen-7B but not the full factorial. Because earlier analyses had inspected ASDiv, V10 is mechanism discovery despite its pre-generation freeze.

4.3 Stage 2: Prospective DROP Test

DROP requires discrete reasoning over paragraphs (Dua et al., 2019). We selected 200 validation IDs whose annotations unanimously normalized to one numeric answer. Before generation, the analysis freeze fixed the data, IDs, prompts, token caches, extractor implementations, primary pair, multiplicity family, checkpoint revisions, and code.

V11 comprised 22,400 generations with greedy decoding, native end-of-sequence behavior, a 1,024-token cap, 4-bit NF4 double quantization (Dettmers et al., 2023), and FP16 computation. These settings were frozen before generation. After Mistral failed the cap gate, a timestamped add-only supersession retained its 3,200 rows, prohibited reruns, and authorized the six remaining checkpoints.

4.4 Statistical Analysis and Cap Policy

The inference unit was dataset ID. We resampled IDs 10,000 times while preserving within-ID dependencies. Two-sided sign-flip diagnostics used 100,000 draws, and Holm correction (Holm, 1979) covered the four predeclared interactions. Primary intervals condition on the fixed panel; an additional checkpoint bootstrap measures panel-composition sensitivity, not population uncertainty.

A checkpoint–condition cell failed the cap gate when more than 2 of 200 responses reached the limit. Primary estimates retained valid rows. The mandatory common-ID sensitivity excluded every ID with any cap hit, yielding one shared panel across checkpoints.

5 Results and Analysis

Unless noted otherwise, effects are LastMarker minus FinalLine interactions in percentage points, with uncertainty from the ID bootstrap. We first report the predeclared DROP test, then examine checkpoint heterogeneity, secondary factors, cap sensitivity, and the cross-task evidence.

5.1 Primary DROP Interaction

Across all V11 rows, the predeclared terminality interaction was +19.72 pp (95% CI $[+17.17, +22.35]$; Holm-adjusted $p = 0.00004$). The checkpoint-resampled interval remained positive but wider, $[+8.18, +29.88]$ pp, and all leave-one-checkpoint-out estimates were positive ($+17.04$ to $+24.29$ pp).

The extractor-specific components localize the difference. Terminality changed exact match by +0.13 pp under LastMarker (95% CI $[-1.55, +1.79]$) and by -19.60 pp under FinalLine ($[-21.38, -17.81]$). Thus, the interaction primarily reflects lost final-line credit when trailing text is allowed, not a comparable gain in recoverable marked answers.

5.2 Checkpoint Heterogeneity and Transfer

Figure 1 and Table 3 show a positive mean on both tasks, with the DROP mean approximately half

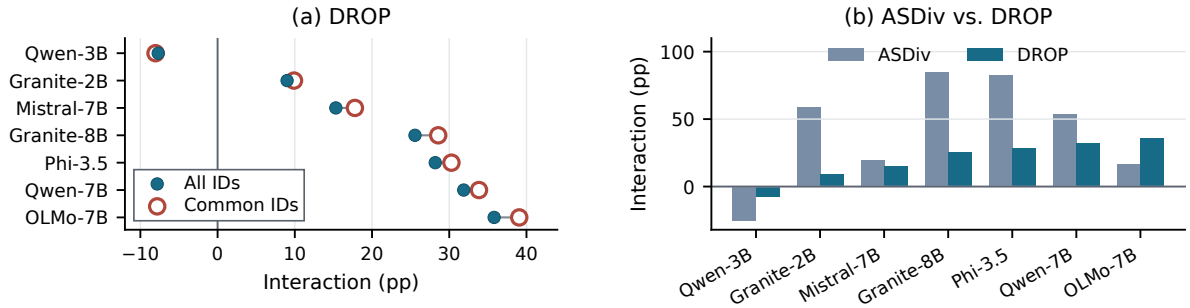

Figure 1: Checkpoint heterogeneity and cross-task transfer. Panel (a) shows DROP all-row effects and the required 168-ID common-panel sensitivity; lines join estimates for the same checkpoint. Panel (b) shows the same terminality-by-extractor estimand on ASDiv and DROP. Both task means are positive and Qwen-3B reverses sign, but magnitudes vary substantially; these are fixed-checkpoint results, not model-family estimates.

	All rows		Common-ID
Checkpoint	**ASDiv**	**DROP**	**DROP**
Qwen-3B	-25.44	-7.69	-8.04
Granite-2B	+58.75	+9.00	+9.90
Mistral-7B	+19.88	+15.31	+17.78
Granite-8B	+85.13	+25.56	+28.57
Phi-3.5	+82.88	+28.19	+30.28
Qwen-7B	+53.56	+31.88	+33.85
OLMo-7B	+17.00	+35.81	+39.06
Equal-checkpoint mean	**+41.68**	**+19.72**	**+21.63**

Table 3: Terminality-by-extractor interactions in percentage points. The Common-ID column excludes every ID with any V11 cap hit.

Factor	**Effect (pp)**	**ID-bootstrap 95% CI**	**Holm p**
Boilerplate	-0.24	[-1.04, +0.56]	0.548
Marker alias	+1.04	[+0.13, +1.92]	0.0476
Terminality	**+19.72**	**[+17.17, +22.35]**	**0.00004**
Unit policy	+7.44	[+6.39, +8.50]	0.00004

Table 4: Complete frozen V11 factorial family over all rows. Effects are LASTMARKER minus FINALLINE interactions; the primary factor is shown in bold.

the ASDiv mean. Six DROP checkpoints were positive, from +9.00 pp for Granite-2B to +35.81 pp for OLMo-7B. Qwen-3B remained negative at -7.69 pp, preserving the ASDiv sign reversal but not its magnitude.

Model family and size do not explain the pattern: Qwen-3B and Qwen-7B had opposing signs on both tasks, while the positive Granite effects differed by more than 16 points on DROP. The task shift also changed checkpoint rankings. Transfer therefore occurs at the mechanism level, not as a common coefficient or a simple family- or size-based rule.

5.3 Secondary Factor Interactions

The boilerplate-by-extractor interaction was not statistically significant on either task (Table 4), providing no evidence of a general interaction from prompt shortening. The marker-alias estimate was small: despite a fixed-panel Holm-adjusted value of 0.0476, its checkpoint-resampled interval $[-5.20, +5.41]$ pp crossed zero and its common-ID adjusted value was 0.137. We therefore do not treat it as robust.

Unit policy produced a smaller interaction of +7.44 pp (95% CI $[+6.39, +8.50]$; adjusted $p = 0.00004$). Its checkpoint-resampled interval $[+3.98, +11.29]$ pp remained positive, and all seven checkpoint estimates were positive. Because units alter whether FINALLINE accepts a numeric boundary, this result supports the same interface account. The ASDiv estimate, +0.75 pp, was not distinguishable from zero after correction, indicating task-dependent magnitude.

5.4 Cap-Gate Failure and Robustness

V11 contained 46 cap hits among 22,400 structurally valid rows. The predeclared cap-safe gate nevertheless failed: Mistral had one 3/200 cell, and Granite-8B had 9/200 and 5/200 cells. The source audits therefore remain failed, and we make no cap-safe replication claim.

The mandatory sensitivity excluded 32 cap-affected IDs and retained 168. The primary interaction was +21.63 pp (95% CI $[+18.87, +24.39]$), with a checkpoint-resampled interval of $[+9.20, +32.52]$ pp; unit policy remained positive at $+7.71$ pp. Cap-affected IDs therefore do not solely explain the panel mean, but the post-generation exclusion is a robustness

Analysis	Estimate	Interpretation
Point estimates		
All rows	+19.72 pp	Frozen primary; cap-safe gate failed
Common 168 IDs	+21.63 pp	Excludes any ID with any cap hit
Composition sensitivity		
ID + checkpoint bootstrap	[+8.18, +29.88] pp	Interval remains positive

Table 5: DROP primary and mandatory robustness analyses. The common-ID result does not retroactively convert failed audits into passes.

Pair	ASDiv	DROP
LM − FL (primary)	+41.68	+19.72
H − FL	+31.75	+14.62
	[+29.74, +33.73]	[+12.13, +17.14]
LM − H	+9.93	+5.11
	[+8.47, +11.44]	[+3.17, +7.19]
Common-ID sensitivity		
H − FL	+33.53	+15.78
LM − H	+10.83	+5.85

Table 6: Exploratory terminality-by-extractor interactions (pp). LM, FL, and H denote LASTMARKER, FINALLINE, and the re-implemented harness filter HARNESS. Brackets give unadjusted ID-bootstrap 95% CIs.

analysis, not a repaired primary run.

5.5 Exploratory Alternative Scorers

To connect the estimand to an extraction convention in use, we re-implemented the lm-evaluation-harness GSM8K `flexible-extract` filter and its exact-match normalization (EleutherAI, 2026): the last number-like regex match, compared after removing commas, dollar signs, and a trailing period. Its `strict-match` filter requires the #### marker, which our prompts never request. This HARNESS rule rescores the same stored responses under the same paired estimand. The analysis was specified after review and lies outside both frozen Holm families (Table 6).

Substituting HARNESS for LASTMARKER retains most of the interaction against FINALLINE on both tasks, and Qwen-3B again reverses sign (-25.56 and -10.44 pp). HARNESS also disagrees with LASTMARKER. On DROP, all seven checkpoint effects are positive (+2.75 to +7.88 pp). On ASDiv, the mean is driven mainly by Granite-2B (+60.94 pp), with the other checkpoints between -0.69 and +4.56 pp. Extractor dependence is therefore not confined to the two author-designed rules, although this check does not reproduce the harness's few-shot GSM8K prompting.

Two further exploratory checks ask where this sensitivity arises. Conservative Markdown normalization, applied before both extractors, changed the interaction by less than 0.1 pp on each dataset. Further relaxing FINALLINE to accept same-line trailing content increased the interaction by 4.88 pp on ASDiv and 3.42 pp on DROP. Rule-based decomposition identified repeated answer markers as the main positive contributor to Qwen-3B's interaction shift under this relaxation. The tested final-line relaxation thus changed the interaction substantially more than the conservative Markdown normalization; these results demonstrate extraction-rule sensitivity, not improved semantic validity.

Exploratory LLM judge. On a reference-level terminality slice (5,600 requests), Gemini 3.5 Flash (Vertex AI alias, temperature 0, JSON output) scored each response given the problem context but not the reference, under a protocol frozen and externally timestamped before any reply. Its interaction with FINALLINE was +37.93 on ASDiv and +23.14 pp on DROP, versus +42.71 and +30.00 pp for LASTMARKER. It matched rule-unambiguous extractions on 97.9% of ASDiv and 80.5% of DROP responses; post hoc, nearly every disagreement returned the reference answer (25/25; 180/199), replacing the response's answer in 1.8% and 16.4% of unambiguous cases. A context-aware judge thus partly grades the task rather than the response.

5.6 Discovery-Stage Controls

ASDiv included controls not repeated on DROP. Placebo deletion and boilerplate paraphrase yielded -2.14 pp and -2.07 pp. Exploratory contract removal and relocation yielded -14.21 pp and +5.14 pp outside the corrected family. These results motivated the narrower V11 prediction that answer-interface dimensions, rather than deletion itself, would interact with extraction.

The V10 BF16 diagnostic covered only two checkpoints and four conditions, so it cannot establish precision invariance. Its mixed checkpoint-specific changes caution against treating NF4 behavior as universal but do not alter the row-level NF4 estimands.

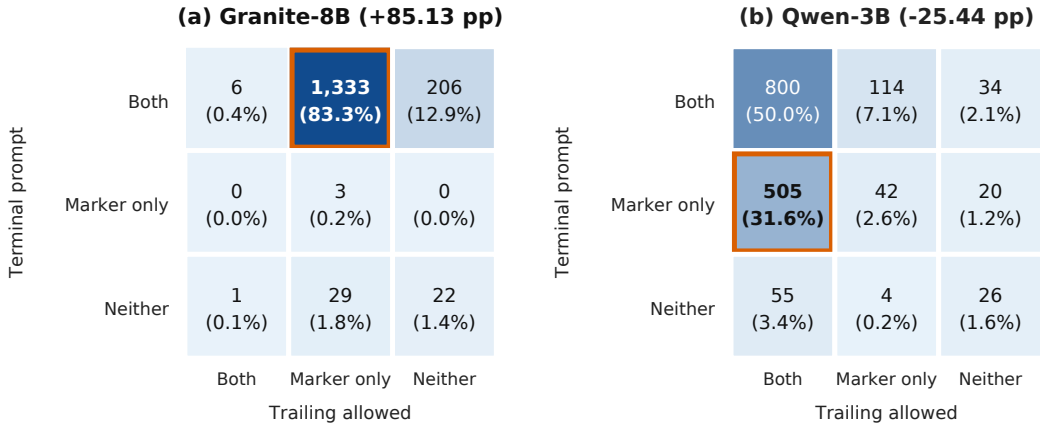

Figure 2: ASDiv paired terminal-to-trailing state transitions for the largest positive checkpoint (Granite-8B) and the sign-reversing checkpoint (Qwen-3B). Granite-8B moves mainly from both-correct to LastMarker-only; Qwen-3B moves in the opposite direction. The reversal demonstrates that the primary extractor pair is not nested.

5.7 Cross-Task Synthesis

Table 7 rules out two simple interpretations. DROP weakens the view that ASDiv was an isolated arithmetic artifact because the predeclared interaction remains positive on paragraph-based reasoning, with the same sign reversal and a stable common-ID analysis. Conversely, the approximately 22-point difference between task means, changed checkpoint ranking, and task-specific unit interaction argue against one universal coefficient.

The absence of a significant boilerplate interaction on both tasks is also informative: deleting short prompt text did not reproduce the interaction. What transfers is sensitivity to instructions that alter the admissible answer boundary. We therefore frame the result as an evaluation audit with a prompt-compression implication, not evidence that compression is generally unsafe.

Falsifiable boundary account. The mechanism predicts smaller interactions when generation is restricted to one typed field, both extractors target the same boundary, or a transformation cannot alter post-answer material. It predicts larger interactions when a transformation changes trailing content, accepted units, or the answer asserted last. Large effects for unrelated deletions but none for boundary manipulations would contradict the account.

Prospective contribution. Timestamping does not make the panel representative or establish all-task prevalence, but it separates hypothesis formation from DROP output inspection. Terminality, extractor ordering, the Holm family, and the cap response were specified before generation; the direction of the cross-task result was not selected after observing V11 responses.

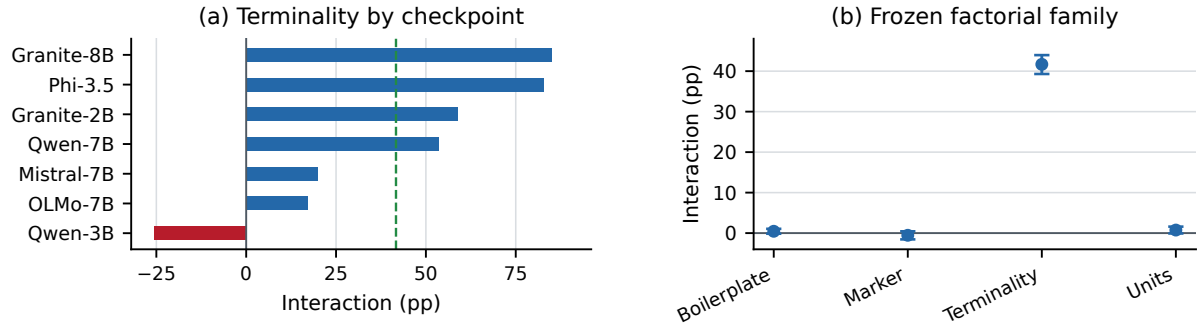

Figure 3: ASDiv V10.2 factorial evidence. Panel (a) shows checkpoint-specific terminality interactions, including the negative Qwen-3B effect. Panel (b) shows ID-bootstrap intervals for the four-factor Holm family; only terminality is distinguishable from zero on ASDiv. Together with the DROP results, this separates a transferring answer-boundary mechanism from task-specific secondary interactions.

Design or result	ASDiv V10.2	DROP V11.1	Cross-task reading
Study design			
Frozen target outputs	29,600	22,400	Both use the same fixed seven-checkpoint panel
Prompt conditions	16 factorial + 4 controls	16 factorial	The confirmatory 2^4 grid is shared
Generation cap	512	1,024	Both original cap-safe gates fail
Empirical comparison			
Terminality interaction	+41.68 pp	+19.72 pp	Positive task means, different magnitudes
Common-ID sensitivity	+44.35 pp	+21.63 pp	Cap-affected IDs do not explain either mean
Boilerplate interaction	+0.45 pp	-0.24 pp	No significant interaction on either task
Unit-policy interaction	+0.75 pp	+7.44 pp	Boundary sensitivity is stronger on DROP
Sign-reversing checkpoint	Qwen-3B	Qwen-3B	The ordered pair can move in either direction

Table 7: Evidence ledger for the sequential studies. The shared result is qualitative and mechanism-level; coefficients and secondary factors are task-dependent.

6 Evaluation and Reporting Implications

Define the construct before the extractor. Evaluation should first specify whether the target is recoverable answer content, compliance with an output interface, or the final asserted answer. The extraction rule should then operationalize that construct, including normalization and abstention. Disagreement between implementations of the same construct is a robustness issue; disagreement between different constructs is a construct-sensitivity result. In our setting, LASTMARKER and FINALLINE intentionally measure different answer boundaries, so neither is a universal reference scorer.

Use paired boundary-sensitivity analyses. When a prompt transformation may change answer placement, studies should rescore identical stored outputs with at least one transparent alternative boundary rule. They should report the prompt effect under each extractor and their paired interaction, rather than only two absolute accuracies. Checkpoint-level estimates and common-item analyses are also necessary when aggregate effects may hide reversals or execution failures.

Attribute claims to the evaluated pipeline. A nonzero interaction means that the treatment effect depends on the evaluator's answer construct. Such a result should be attributed to the prompt–model–extractor pipeline, not reported as a prompt-only gain or loss. Failed gates and post-generation exclusions should remain visible in the reporting record rather than being absorbed into a corrected headline result. Table 8 summarizes the minimum disclosure.

Report item	Minimum disclosure
Target construct	Recoverable content, interface compliance, or final assertion
Prompt contrast	Exact rendered prompts and achieved token change
Extraction	Executable rules, normalization, and abstention behavior
Estimand	Extractor-specific effects and their paired interaction
Sensitivity	Alternative boundaries, checkpoints, common items, and failed gates

Table 8: Minimum reporting record for prompt transformations that may change answer placement.

Implication for prompt compression. Compressor comparisons should include paired rescoring under an alternative boundary when compression changes answer placement or the admissible boundary.

7 Artifact Availability and Reproducibility

The artifact is publicly available.[1] It contains rendered prompt caches, pinned checkpoint revisions, generation settings, executable extractors, analysis code, stored outputs, failed audit records, supersession manifests, timestamp receipts, and checksums. The release covers 52,000 generations: 29,600 from ASDiv V10.2 and 22,400 from DROP V11.1. It also holds the frozen LLM-judge protocol, raw

[1] https://github.com/howwang0507/prompt-extraction-audit/releases/tag/v12.3

replies, and incident records.

The reconstruction pipeline validates unique dataset–checkpoint–condition keys, reapplies the released extraction rules, and reproduces every reported estimate; primary, exploratory, and superseded records remain distinguished, and fixed-seed scripts reproduce the post hoc analyses in Section 5.5.

8 Conclusion

This work asks whether a measured prompt effect survives a change of extraction rule alone. Rescoring identical outputs with two non-nested rules, permitting post-answer text produced interactions of +41.68 pp on ASDiv and +19.72 pp on DROP, driven mainly by final-boundary failures, positive in common-ID and checkpoint-composition sensitivities, and heterogeneous across checkpoints; for some checkpoints the rule change reversed the effect's sign. In exploratory checks, the tested final-line relaxation changed the interaction substantially more than conservative Markdown normalization. An exploratory context-aware LLM judge did not remove this dependence and sometimes substituted the reference answer for the response's own answer. We found no statistically significant boilerplate-by-extractor interaction on either task, and neither extractor was established as normatively superior.

The practical consequence is a reporting requirement. Studies that compare prompts on extracted answers should report the executable extraction rule and normalization, the share of responses the rule cannot extract in each condition, and whether the prompt effect changes under at least one alternative rule applied to the same outputs (Section 6). Effects that change under such rescoring belong to the prompt–model–extractor pipeline. This is a measurement result about the tested numeric tasks, not a ranking of extractors or evidence that prompt compression is generally unstable.

9 Limitations

Scope and sampling. We study two numeric tasks and seven purposive checkpoints; numeric answers allow exact-match scoring that isolates the answer boundary. DROP broadens reasoning demands but shares the ASDiv template, extractors, normalization, and estimand, leaving a possible interface artifact. ID-bootstrap intervals condition on this panel; checkpoint resampling measures composition sensitivity, not population uncertainty. Results may not generalize to nonnumeric outputs, native interfaces, other decoding regimes, or model populations.

Construct validity and intervention. Terminality directly targets post-answer text, strengthening mechanism identification but not representing natural prompt edits. Beyond the harness-style filter and one context-aware LLM judge (alias-addressed without a dated snapshot, reference-level slice only), we did not compare human judgments or structured-output constraints. Without human judgments of intent or compliance, neither extractor is normatively superior. The factorial also manipulates instructions rather than learned compressors at matched budgets, so its compression implication is methodological, not comparative.

Sequential evidence. Earlier ASDiv analyses informed the hypothesis, so V10 is discovery. V11 preceded DROP generation but tests one deterministic numeric subset. Timestamping limits outcome-adaptive specification without validating the construct or sampling. Additional ASDiv controls were exploratory and outside the Holm family.

Execution and robustness. Frozen cap gates fail in both studies. Common-ID analyses preserve the direction after post-generation exclusion but change the item set and do not repair primary runs. Most outputs use NF4; the limited BF16 diagnostic covers neither all checkpoints nor the full factorial, so precision invariance is unestablished.

10 Ethical Considerations

The experiments use public benchmarks and checkpoints and involve no participants or personal data. The main risk is overstatement, such as reading an extractor interaction as universal prompt instability or proof that one rule is correct; we therefore report non-significant factors, reversals, failed gates, and missing human adjudication.

References

Anthony Chen, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Evaluating question answering evaluation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 119–124. Association for Computational Linguistics.

Alexis Chevalier, Alexander Wettig, Anirudh Ajith, and Danqi Chen. 2023. Adapting language models to compress contexts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3829–3846. Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota. Association for Computational Linguistics.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. 2015. Generalization in adaptive data analysis and holdout reuse. In *Advances in Neural Information Processing Systems*, volume 28, pages 2350–2358. Curran Associates, Inc.

EleutherAI. 2026. GSM8K task configuration in lm-evaluation-harness. https://github.com/EleutherAI/lm-evaluation-harness/blob/main/lm_eval/tasks/gsm8k/gsm8k.yaml. Accessed 2026-09-25.

Sture Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70.

Andong Hua, Kenan Tang, Chenhe Gu, Jindong Gu, Eric Wong, and Yao Qin. 2025. Flaw or artifact? rethinking prompt sensitivity in evaluating LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 19889–19899. Association for Computational Linguistics.

IBM Granite Team. 2025. Granite-3.3-8b-instruct model card. https://huggingface.co/ibm-granite/granite-3.3-8b-instruct. Accessed 2026-07-26.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023a. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023b. LLMLingua: Compressing prompts for accelerated inference of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13358–13376. Association for Computational Linguistics.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2024. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1658–1677. Association for Computational Linguistics.

Weronika Łajewska, Momchil Hardalov, Laura Aina, Neha Anna John, Hang Su, and Lluis Marquez. 2025. Understanding and improving information preservation in prompt compression for LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 17520–17541. Association for Computational Linguistics.

Yucheng Li, Bo Dong, Frank Guerin, and Chenghua Lin. 2023. Compressing context to enhance inference efficiency of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6342–6353. Association for Computational Linguistics.

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

Shen-yun Miao, Chao-Chun Liang, and Keh-Yih Su. 2020. A diverse corpus for evaluating and developing english math word problem solvers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 975–984. Association for Computational Linguistics.

Microsoft. 2024. Phi-3.5-mini-instruct model card. https://huggingface.co/microsoft/Phi-3.5-mini-instruct. Accessed 2026-07-16.

Mistral AI. 2023. Mistral-7b-instruct-v0.2 model card. https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2. Accessed 2026-07-14.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. 2024. State

of what art? a call for multi-prompt LLM evaluation. *Transactions of the Association for Computational Linguistics*, 12:933–949.

Francesco Maria Molfese, Luca Moroni, Luca Gioffrè, Alessandro Scirè, Simone Conia, and Roberto Navigli. 2025. Right answer, wrong score: Uncovering the inconsistencies of LLM evaluation in multiple-choice question answering. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 18477–18494. Association for Computational Linguistics.

Jesse Mu, Xiang Lisa Li, and Noah Goodman. 2023. Learning to compress prompts with gist tokens. In *Advances in Neural Information Processing Systems*, volume 36, pages 19327–19352. Curran Associates, Inc.

Alliot Nagle, Adway Girish, Marco Bondaschi, Michael Gastpar, Ashok Vardhan Makkuva, and Hyeji Kim. 2024. Fundamental limits of prompt compression: A rate-distortion framework for black-box language models. In *Advances in Neural Information Processing Systems*, volume 37.

OLMo Team. 2025. 2 OLMo 2 Furious. *arXiv preprint arXiv:2501.00656*.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Dongmei Zhang. 2024. LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 963–981. Association for Computational Linguistics.

Qwen Team. 2024a. Qwen2.5-7B-Instruct model card. `https://huggingface.co/Qwen/Qwen2.5-7B-Instruct`. Accessed 2026-07-25.

Qwen Team. 2024b. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Rishi Ravikumar, Nuhu Ibrahim, and Riza Batista-Navarro. 2026. Lost in formatting: How output formats skew LLM performance on information extraction. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5498–5513. Association for Computational Linguistics.

Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. 2024. ProSA: Assessing and understanding the prompt sensitivity of LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1950–1976. Association for Computational Linguistics.