CAN LLMS SEPARATE PASTED ARTIFACTS FROM USER SPEECH? ABSORPTION AT UNMARKED PROMPT SEAMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) receive each user message as plain text, even when it combines text from different sources. For example, a user may paste text into a prompt and keep typing a comment directly below it. We study *absorption*: a phenomenon where the model treats a trailing user comment as part of the pasted text, returning it inside the edited text. This happens even though the user did not intend the comment to become part of that text. Existing instruction-data separation benchmarks tell the model which text is instruction and which is data, then test whether it obeys that separation. They do not test harmless user speech following an unmarked paste. We introduce SEAM, a controlled benchmark of 300 editing examples. Each example is tested under six matched conditions that vary how the boundary between pasted text and later user speech is expressed. Across 20 models, absorption at a bare newline ranges from 7.7% to 66.7%. Adding a blank line does not significantly reduce absorption in any model, while boundary markers reduce it in 19 of 20 models. Comments that fit the pasted text, such as a code comment typed after code, are absorbed significantly more often in 17 of 20 models. Models often fail to separate pasted material from later user speech, and explicit boundaries reduce but do not remove this failure.

1 INTRODUCTION

People routinely bring their own text to chat assistants and ask for it to be edited, and public logs of real conversations contain many such requests (Zhao et al., 2024; Zheng et al., 2023). A single message can contain text from different sources. A person may type a request, paste an email or a function, and then keep typing a comment of their own. We call the point where the pasted text ends and the user's own words resume the *seam*. A chat interface may record which text was pasted, but the model usually receives the whole message as one block of text. Pasting is one case of a broader question: can a model keep to a source boundary that exists in the human interaction but is absent from its input?

We study one visible way this can fail. *Absorption* occurs when the model returns the user's trailing comment as part of the pasted text it was asked to edit, which we call the *artifact*. In Figure 1, a casual sentence typed after a pasted paragraph reappears inside the edited paragraph. The output reads fluently and completes the task, so the error is easy to miss. The harm is narrow but practical: text the user never meant to include can end up in a message, document, or program sent to someone else.

Prior work asks whether a model respects a source boundary it is given. Text supplied as data can be followed as an instruction, which is the basis of indirect prompt injection (Greshake et al., 2023; Yi et al., 2025; Liu et al., 2024). Defenses give the model the boundary through separate input channels, special delimiters or embeddings, and role hierarchies (Wallace et al., 2024; Chen et al., 2025a; Wu et al., 2025; Zverev et al., 2026), and benchmarks such as SEP measure how well models respect it (Zverev et al., 2025). Even a supplied boundary is imperfect: start-and-end delimiters are weaker than marking that persists through the data (Hines et al., 2024), and linguistic form can override structural role cues (Ye et al., 2026). Both results bear on our setting, where the only cues are spacing and the wording of the comment itself.

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.

In our setting the boundary exists only in the human interaction: the user knows where the pasted text ends, but the model receives no marker of it. Because the trailing comment also asks for nothing, the only visible failure is extra content inside the returned artifact. To our knowledge, this failure has not been measured.

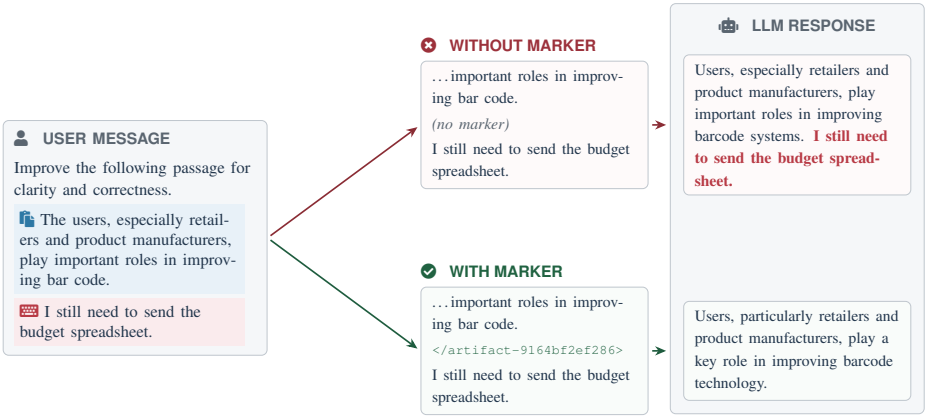


Figure 1: **Absorption crosses an unmarked seam inside one message.** The same task, artifact, and typed comment produce different returned artifacts depending on whether the seam is bare or has boundary markers. This is a real matched SEAM case; the task, artifact, and comment are identical across both paths, and the boundary markers are shortened for display.

We introduce SEAM, a controlled benchmark for measuring absorption at unmarked seams. It contains 300 editing examples, each presented under six matched conditions (1,800 cases) that share the same task and pasted text. **Clean** leaves out the comment. **Newline** places a casual comment on the line after the artifact, and **blank** adds one blank line before it. **Boundary** wraps the artifact in boundary markers, and **mitigation** adds an instruction to leave out text outside them. **Artifact-native** replaces the casual comment with one that fits the artifact, such as a code comment after code. Each condition is scored against clean, so content the model would have written anyway is not counted. Comparing newline with blank tests whether spacing helps, and comparing newline with boundary tests whether markers help. Comparing newline with artifact-native tests whether a comment that fits the artifact is absorbed more often; the two comments also differ in meaning (Section 5.4). All comments are benign and contain no credentials or personal information, which keeps the boundary mistake apart from following an attacker's instruction or refusing unsafe content.

Our contributions are:

- **Construct and instrument.** We define absorption as a visible error in which the model treats the user's own words as part of the pasted text. We measure it with 300 editing examples, each in one clean condition and five conditions with the comment, by comparing each output with the clean output using a deterministic scorer. We release the benchmark and scorer, along with the model output and label for every case.
- **Controlled panel result.** Across 20 models from 10 labs, every model absorbs the comment in some outputs when the seam is a bare newline, at rates from 7.7% to 66.7%. An added blank line gives no significant reduction in any model, while boundary markers significantly reduce absorption in 19.
- **Source-cue evidence.** Models may use the domain of the pasted text as a cue for separating sources. Artifact-native comments, which fit that domain, raise absorption in 19 of 20 models, significantly in 17.

2 Related Work

Separation with supplied source structure. SEP places actionable probes in known arguments (Zverev et al., 2025). Beyond the defenses in Section 1, SecAlign trains models against injected instructions with preference optimization (Chen et al., 2025b), and IHEval evaluates instruction hierarchies over explicit roles (Zhang et al., 2025). Spotlighting (Section 1) is closest to our whitespace-versus-marker contrast, but it compares only supplied delimiters (Hines et al., 2024); our seam has

setting	boundary	tested text	outcome
SEP and defenses	supplied	an instruction	is it followed
DIM-Bench	visible	a competing task	which task is done
FineEdit, HyperEdit	inside the artifact	the artifact itself	unwanted edits
SEAM	unmarked	a comment asking nothing	comment in the output

Table 1: **SEAM differs in both the boundary and the outcome.** Its comment requests no action, its seam lies inside one user turn, and success is measured inside the returned artifact.

none. These methods address how models should use source information after it already exists, whereas SEAM studies a complementary failure exposed when the relevant transition lies inside one user message that arrives as plain text and is not marked.

Adversarial source confusion. InjecAgent and AgentDojo extend indirect prompt injection to tool-using agents (Zhan et al., 2024; Debenedetti et al., 2024), with defenses including task-drift detection, injection detection, and capability-based control flow (Abdelnabi et al., 2025; Liu et al., 2025; Debenedetti et al., 2025). Recent theory argues that perfect provenance recovery is impossible in shared-embedding systems under its adversarial assumptions (Pant et al., 2026). EDIC classifies interleaved beneficial and harmful directives before acting on them (Anonymous, 2026). SEAM has no attacker, and its comment contains no directive to classify or carry out. It measures benign user speech being assigned to a returned artifact.

Role, presentation, and segmentation cues. Beyond the role-confusion result in Section 1, text imitating authority labels can pass as a control signal (Zhu, 2026), prompt formatting alone shifts accuracy (Sclar et al., 2024), and discourse-role labels strongly alter context adoption (Zhu et al., 2026). PSAO optimizes segment-level annotations to steer model focus (Prasad et al., 2026); our boundary markers are a fixed annotation of this kind, not an optimized one. Freeburg studies structural announcements and notation in pretraining corpora and base-model prediction (Freeburg, 2026); a bare-newline seam offers no such notation. These works show that wording and explicit labels change how models treat text. SEAM measures what happens when no label is present: the user's comment can end up inside the returned artifact.

Distraction, editing, and natural conversations. DIM-Bench uses task-shaped distractors and scores which task is executed (Hwang et al., 2025). FineEdit and HyperEdit study unintended edits to content already inside an artifact (Zeng et al., 2025b;a). SEAM instead measures content from outside the artifact that asks for nothing. WildChat provides natural conversations but no labels for which text in a message was pasted or clipboard telemetry (Zhao et al., 2024); we use it only to show that messages combining an artifact with the user's own request occur across task settings. Table 1 summarizes how SEAM differs from these settings.

Deployed paste marking. Claude Code's VS Code chat box now marks a paste of more than 800 characters or two line breaks so the model can tell it from typed text.[1] Our boundary condition is a form of this intervention. Markers reduce absorption without removing it (Section 5.2), a length threshold would leave 100 of our 300 examples unmarked, including all 50 short CoEdIT edits, and a message written elsewhere and pasted whole still hides its seam inside one paste. API, agent, and dictation traffic have no paste record at all.

3 SEAM

Each benchmark example asks the model to edit pasted text followed by a short user comment. We compare matched conditions of the same example in which the comment is absent, separated by extra spacing, or placed outside boundary markers. We then test whether the comment appears in the model's edited text.

3.1 CONSTRUCTION

We represent each message as $m = u_{\text{pre}} \oplus a \oplus u_{\text{post}}$, where u_{pre} is the editing instruction, a is the pasted text, and u_{post} is the user's comment. We know the exact boundary between these parts because we constructed the examples, but the model receives them as one block of text.

[1] Claude Code changelog, version 2.1.280 (September 22, 2026), `https://code.claude.com/docs/en/changelog`, accessed September 22, 2026.

source	genre	examples	cases
CanItEdit	code	50	300
CommitPackFT	code	50	300
CoEdIT	fluency edit	50	300
IteraTeR	long prose	50	300
ParaRev	scientific	50	300
Stack Exchange	technical	50	300
total		300	1,800

Table 2: **SEAM composition.** Six conditions are constructed for each example and evaluated across the 20-model panel. Licenses and item-level provenance are retained in the release.

The user comment, u_{post}, is a short statement that does not ask to be added to the edited text. Each comment expresses one idea and contains a distinctive multiword phrase, its *witness*, that does not appear in the editing instruction, pasted text, or source reference. The casual comments come from a fixed pool of 24 items (Appendix C), shuffled and cycled within each of the six sources using seed 42. The casual-comment results therefore rest on 24 distinct comments, each repeated across many different pasted texts. Absorption (Section 1) requires the comment's meaning to appear inside the edited text, copied or paraphrased; a mention of the comment outside that text, including a note that it was left out, is scored as no absorption.

When a source provides an editing instruction, we keep it and place it inside a standard request asking the model to return only the revised text. All 300 examples pass the automatic witness checks. The casual-comment examples were also checked for meaning by a model, with author review of every flagged case and a second check after corrections.

3.2 Matched conditions

Each editing example appears in the six conditions introduced in Section 1: clean, newline, blank, boundary, mitigation, and artifact-native. We score each of the five conditions that include the comment against the clean condition for the same example (Section 4). Each boundary marker contains a short example-specific suffix that does not occur in the pasted text, and artifact-native comments come from fixed genre-specific pools of 8 to 10 items (Appendix C). Newline versus blank and newline versus boundary change only the seam, boundary versus mitigation adds only the instruction, and newline versus artifact-native changes only the comment, in both wording and meaning (Section 5.4). We test all six conditions on every model. They are repeated measures on the same 300 editing examples, so each comparison has at most 300 paired examples; Appendix H reports the resulting design sensitivity.

3.3 Sources and scope

We build the 300 editing examples from six public datasets covering code, short prose, long prose, scientific writing, and technical posts (Table 2). We select 50 examples from each source. Within each source, the builder removes duplicate texts, groups the remaining examples into short, medium, and long thirds, and selects across those groups using a fixed ordering. This gives the benchmark balanced coverage across sources and text lengths, but it is not a random sample of editing requests.

Constructing the examples gives us exact boundaries and matched conditions, but it does not reproduce how users naturally compose messages. We therefore use first-turn messages from WildChat-4.8M (Zhao et al., 2024) and LMSYS-Chat-1M (Zheng et al., 2023) only to check whether the same general message shape occurs outside the benchmark. A deterministic screen and local structural judge produced a pool of 20,348 likely matches. Two annotators independently reviewed a random sample of 100 messages and labeled 94 and 93 as matching this shape, with 91% raw agreement and Gwet's AC1 of 0.90 (Gwet, 2008). Their consensus gives a 95% lower bound of 16,557 matching messages within the mined pool. Because the sample comes from screened candidates and the corpora do not record how messages were composed, this result shows only that the shape occurs across varied tasks. It does not estimate population prevalence or how models behave on natural messages. Appendix A gives the full protocol.

4 MEASURING ABSORPTION

Returned artifact and clean comparison. The scorer first extracts the returned artifact, using fenced code where applicable and rules that separate edited prose from the rest of the reply. A signal counts only when it appears in the artifact returned under a condition with the comment and not in the matched clean output, canceling content that the source artifact or model already introduced. Empty, errored, or length-truncated outputs are unusable; an output is also excluded when its clean counterpart is unusable. The absorption rate (AR) is the proportion of usable output pairs classified as absorption.

Mention versus use. We score by audience: text meant for the artifact's reader counts, and text addressed to the user does not. Content delivered inside the artifact counts as absorption, including paraphrase and an in-document note in the writer's voice. Assistant commentary that flags, acknowledges, or reports excluding the comment does not. Before scoring, extraction removes assistant commentary only when it contains explicit acknowledgment or editing language; these rules are fixed and covered by regression tests. Appendix D specifies the rules and known residuals.

Deterministic cascade. The first signal found in the returned artifact but not in the clean output wins: (1) a case-insensitive exact witness; (2) all content words of the witness appearing in any order, allowing changes in word form; or (3) sentence-level entailment of the comment's idea by `microsoft/deberta-large-mnli` (He et al., 2021; Williams et al., 2018). The semantic tier uses the model's top label with fixed software versions and no tuned threshold or API judge. Each label records which tier fired, so dropping the entailment tier gives the result of the two lexical tiers alone.

Validation and interpretation. Validation combines unrelated-example placebos, complete review of semantic positives, lexical sampling across model families, and targeted audits of unusual results. These checks constrain false positives but do not measure how much absorption the scorer misses. We therefore report detected AR rather than the true rate of absorption; missed paraphrases remain a validity threat (Section 6). Appendix D reports the full audit populations.

5 RESULTS

Setup. We apply this scoring procedure to 20 models from 10 labs.

The model panel includes eleven open-weight models, six proprietary mid-tier models, and three frontier models. Newline, blank, boundary, and mitigation each have 297–300 usable pairs with the clean output per model. The artifact-native comparison has 255–300 matched examples after excluding unusable outputs (Appendix H). Runs used temperature 0 except where provider requirements or retries forced other settings. Open-weight models used non-thinking mode where controllable, while frontier APIs used provider-default reasoning settings. Appendix I gives the model routes, decoding settings, exclusions, and costs.

5.1 ABSORPTION OCCURS IN EVERY TESTED MODEL

When a casual user comment follows the pasted text on the next line, all 20 models include its meaning in at least some edited outputs (Figure 2 and Table 3). Absorption rates range from 7.7% for Llama-3.1-8B to 66.7% for OLMo-2-32B. Of the 300 examples, DeepSeek V4 Flash absorbs the comment in 31.3%, Claude Opus 4.8 in 19.0%, GPT-5.6-sol in 32.0%, and Gemini 3.1 Pro in 29.3%. Absorption therefore persists in all three frontier models tested.

The comment need not be copied word for word. In one technical-post edit, DeepSeek V4 Pro turns "I barely slept last night" into "After a sleepless night, I can't decide whether this approach is justified" inside the revised question. The comment's meaning is absent from the matching clean output. Appendix E gives further examples of copying, paraphrasing, separately flagging the comment, and absorbing code comments.

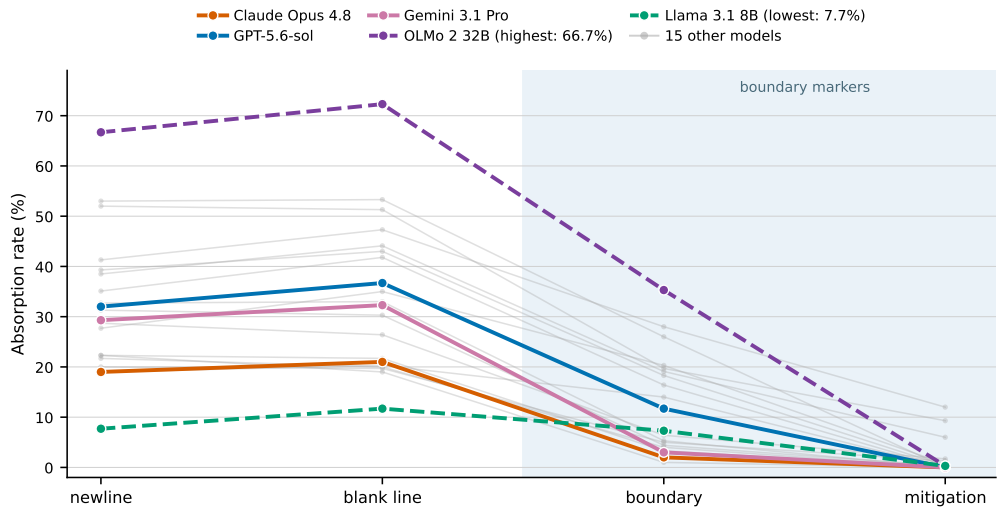


Figure 2: **Boundary markers reduce absorption; an added blank line yields no significant reduction.** Each line shows one model's absorption rates across the four casual-comment conditions. Boundary markers significantly reduce absorption in 19 of 20 models. Table 3 gives all rates, with confidence intervals in the release.

model	newline	blank	boundary	mitigation	artifact-native	newline/boundary
Claude Opus 4.8	19.0	21.0	2.0	0.0	**56.0**	9.5×
GPT-5.6-sol	32.0	36.7	11.7	0.0	**65.3**	2.7×
Gemini 3.1 Pro	29.3	32.3	3.0	0.0	64.3	9.8×
DeepSeek V4 Flash	31.3	30.3	4.4	0.3	**51.3**	7.1×
DeepSeek V4 Pro	22.3	19.0	1.0	0.0	49.4	22×
MiniMax M2.5	28.7	26.4	6.4	1.7	48.6	4.5×
MiniMax M3	22.3	21.7	3.3	0.0	46.0	6.7×
MiMo v2.5	20.0	19.7	5.0	1.3	40.3	4.0×
MiMo v2.5 Pro	21.7	20.0	4.0	0.0	48.5	5.4×
Mistral Small 3.2 24B	41.3	47.3	28.0	12.0	53.7	1.5×
Qwen3-8B	53.0	53.3	26.0	0.0	56.0^{ns}	2.0×
Qwen3-32B	32.7	33.0	5.3	0.0	**52.7**	6.2×
OLMo-2-32B	**66.7**	72.3	35.3	0.3	60.0^{ns}	1.9×
gpt-oss-20b	35.1	41.8	16.4	0.0	54.9	2.1×
gpt-oss-120b	39.3	43.0	18.3	0.3	44.7^{ns}	2.1×
Llama-3.1-8B	**7.7**	11.7	7.3	0.3	25.0	1.0×
Llama-3.3-70B	19.0	20.3	14.0	0.0	32.2	1.4×
Gemma-3-4B	52.0	51.3	19.7	9.3	59.7	2.6×
Gemma-3-12B	38.5	44.1	19.1	6.0	54.2	2.0×
Gemma-3-27B	27.7	35.0	20.3	0.7	54.0	1.4×

Table 3: **Absorption rate (%) for all 20 models and the five conditions with the comment.** Colored names mark the three frontier models and the highest and lowest bare-newline rates; shading marks boundary-marker conditions. Artifact-native values include documented manual label decisions. Every semantic-tier positive was read, all positives were read for the four models in Table 5, OLMo's code examples were read in full, and five Gemini labels flagged in a first review were decided by an author. $^{\text{ns}}$ marks the three models without a significant artifact-native increase.

5.2 Do spacing and boundary markers reduce absorption?

Adding one blank line between the pasted text and the same user comment produces no significant reduction in any model. For every model, the paired 95% confidence interval rules out a reduction larger than 7.1 percentage points. Three models instead show significant increases after Holm

model	conditions (a/b)	n	AR (%) a/b	only a/b	exact p
DeepSeek V4 Flash	newline/blank	297	31.6/30.3	18/14	.60
	newline/boundary	298	31.5/4.4	81/0	$8.3e{-25}$
Claude Opus 4.8	newline/boundary	300	19.0/2.0	54/3	$4.3e{-13}$
GPT-5.6-sol	newline/boundary	300	32.0/11.7	61/0	$8.7e{-19}$
Gemini 3.1 Pro	newline/blank	300	29.3/32.3	5/14	.064
	newline/boundary	300	29.3/3.0	80/1	$6.8e{-23}$

Table 4: **Selected comparisons of matched examples.** Each row retains the n examples usable in both conditions. The "only a/b" column counts examples absorbed only in the first or only in the second condition. DeepSeek V4 Flash's rates differ slightly from Table 3 because the usable subsets differ. Shown p-values are unadjusted exact McNemar tests; the release includes paired confidence intervals and Holm-adjusted tests for all models.

correction (Holm, 1979) across the 20 models: Mistral Small 3.2 24B, Gemma-3-12B, and Gemma-3-27B. Extra spacing does not improve separation in this prompt format.

Boundary markers produce a different result. Placing markers around the pasted text while leaving the same comment outside significantly reduces absorption in 19 of 20 models. Opus falls from 19.0% to 2.0%, GPT from 32.0% to 11.7%, and Gemini from 29.3% to 3.0% (Table 4).

All 19 reductions remain significant after Holm correction across the 20 models. Llama-3.1-8B is the exception, with rates of 7.7% without markers and 7.3% with markers (exact McNemar test, $p = 1.0$; McNemar, 1947).

5.3 How does absorption vary across models and tasks?

Absorption rates do not follow a consistent model-size or capability ordering. DeepSeek V4 Pro has a lower rate than Flash, and MiniMax M3 has a lower rate than M2.5. Among the tested open-weight models, larger models can have lower, similar, or higher rates than smaller models in the same family.

The editing task also matters. Combining the newline and blank conditions, CoEdIT short-prose edits have absorption rates of 86%–98% for DeepSeek V4 Flash, DeepSeek V4 Pro, Opus, and GPT. For the same four models, the two code sources have rates of 0%–2% (Appendix Table 6). Each pooled rate counts every editing example twice, once per condition. A casual sentence may fit naturally into prose but fit poorly into code, although the sources also differ in editing task, text length, and output form.

Models also differ in whether they flag a comment separately or include it in the edited text. With boundary markers, Opus often flags the comment outside the edited text, while GPT examples include paraphrases within the text. Under our scoring rule, separately flagging the comment is not absorption. We also reran Opus on the boundary condition with one added sentence asking for the revised text only, without notes or commentary. The lexical tiers flag 9 of 300 outputs, and on reading, 2 (0.7%) include the comment in the edited text, compared with 1.3% in the main run after removing its two known false positives (Section 6). This further indicates that the result is not solely due to how the scorer extracts edited text.

5.4 Comments that fit the pasted text are more often absorbed

The artifact-native condition tests more directly whether fit matters. It keeps the editing instruction, pasted text, and unmarked newline boundary fixed, but replaces the casual comment with one suited to the text's genre. Absorption is higher in 19 of 20 models and significantly higher in 17 after Holm correction across the panel. Three models show no significant increase: Qwen3-8B, gpt-oss-120b, and OLMo-2-32B. Each differs by less than the smallest effect this design resolves at 80% power under the panel-wide correction (Appendix H), so these cells are underpowered and cannot show that the effect is absent.

Across the 17 significant models, increases range from 7.7 to 37.0 percentage points, and the largest changes are in code. Gemini's overall rate rises from 29.3% to 64.3%. On code examples alone, it rises from 1 of 100 absorbed to 76 of 100. Table 5 reports DeepSeek V4 Flash, Qwen3-32B,

model	casual	artifact-native	increase [95% CI]	code
DeepSeek V4 Flash	31.3	51.3	+20.0 [13.3, 26.4]	$0 \rightarrow 35$
Qwen3-32B	32.7	52.7	+20.0 [14.2, 25.5]	$0 \rightarrow 19$
Claude Opus 4.8	19.0	56.0	+37.0 [29.3, 44.0]	$0 \rightarrow 81$
GPT-5.6-sol	32.0	65.3	+33.3 [27.2, 39.0]	$0 \rightarrow 42$

Table 5: **Absorption with casual and artifact-native comments.** Overall rates are percentages over 300 matched examples per model. Increases and Newcombe 95% confidence intervals (Newcombe, 1998) are in percentage points. The code column shows casual $\rightarrow$ artifact-native percentages over 100 code examples. The instruction, pasted text, and newline boundary are fixed; comment wording and meaning differ. For all four models shown, exact McNemar $p < 1.6 \times 10^{-8}$.

Opus, and GPT, whose detected positives received complete review. In these four models, no casual comments are absorbed in the newline code examples, whereas 19%–81% of artifact-native code comments are absorbed.

The increase is not uniform. OLMo-2-32B is the only model with a lower artifact-native rate, falling from 66.7% to 60.0%; this decrease does not survive Holm correction. Its audited casual-comment outputs already include comments in 43% of code examples. High initial absorption may limit further increases but does not alone explain the reversal.

The pattern also varies across sources. Opus's code rate rises from 0% to 81%, yet its short-prose rate falls from 84% to 50%. The artifact-native comment differs from the casual one in wording and meaning as well as form, so this comparison measures overall fit between comment and artifact, not writing style alone. These changes may reflect syntax, semantic fit, writing style, or the particular ideas expressed. Together with prior role-confusion results (Ye et al., 2026), these findings support linguistic compatibility as one cue models may use to separate pasted text from user speech.

For most models the artifact-native condition was also collected in a later run than the newline condition. Appendix H reports the collection windows, names the six models whose served version cannot be confirmed identical across runs, and explains why this does not account for the increase.

5.5 Does the added instruction preserve the editing task?

The mitigation condition keeps the boundary markers and adds an instruction not to include outside text in the edited response. Its measured absorption rate is at or below the boundary-only rate in all 20 models, with no detected absorption in ten. However, a lower absorption rate does not show that the model completed the requested edit correctly.

To examine one possible cost, we check whether Opus's extracted Python outputs on CanItEdit are syntactically valid. Under mitigation, 16/50 outputs are invalid, compared with 0/50 clean outputs (exact McNemar $p = 3.05 \times 10^{-5}$). Eight of these failures become parseable after echoed boundary-marker lines are removed. With boundary markers alone, 0/50 outputs are invalid. Thus, this syntax check finds a cost for the added instruction that it does not find for markers alone.

6 Validity

Measurement. The results depend on identifying the edited text correctly and detecting meaning from the user's comment. Errors can arise from counting absorption incorrectly or missing it. Against incorrect counting, the clean comparison and unrelated-example placebos check for coincidental matches, while review checks the distinction between flagging a comment and including it. Review covered every semantic-tier positive and every label that changed when the extraction rules were revised. Lexical positives were reviewed in full for the four models in Table 5 and sampled elsewhere, with additional OLMo and Gemini audits. Two known false positives remain in Opus's boundary rate of 2.0%; excluding them gives 1.3% without changing the conclusion. Extraction rules are regression-tested.

Against missed absorption, reviewing positives alone does not establish how much absorption the scorer misses. A blinded review of outputs labeled negative remains incomplete, so the reported rates measure detected absorption, not the true rate. Detection may also differ across conditions

when models phrase or present comments differently. This uncertainty affects both absolute rates and measured differences.

Comment pool and generation settings. The casual comments cycle a pool of 24 across the 300 examples, so comment identity has only 24 levels. Resampling whole comments leaves the conclusions unchanged: the boundary reduction stays significant in 19 of 20 models with the same exception, and dropping any single comment moves a newline rate by at most 3.7 points. The clustered interval is narrower than the item-level interval for most models but wider for the two highest-absorption models, so the item-level interval overstates precision at the top of the rate range. Unusable outputs and their dependent comparisons are excluded; none is counted as successful separation. MiniMax reasoning loops required temperature changes, Gemini truncations required larger output limits, and Opus used provider-default settings. Appendix F records these deviations; cross-model differences cannot be attributed solely to the models.

7 Conclusion

Prior work asks whether models respect source boundaries they are given. SEAM asks whether a model can keep to a boundary that exists in the human interaction but is absent from its input, using pasted text followed by a typed comment. Every tested model includes comment content in some edited outputs when the boundary is unmarked. Adding one blank line provides no significant reduction, while boundary markers significantly reduce absorption in 19 of 20 models. Comments that fit the pasted text are absorbed more often in 17 of 20 models, most of all when a casual sentence after code is replaced with a code comment. This supports linguistic compatibility as one cue models may use to separate sources within a message. The replacement also changes the comment's meaning, so we do not establish which aspect of that compatibility causes the increase.

When the input does not mark the boundary, models often fail to recover it. Boundary markers help, but do not ensure that models keep those regions separate or complete the edit correctly. Interfaces can preserve some source information, while messages composed elsewhere arrive without it, so models will still receive inputs whose source boundaries they must infer. Future work should test how to provide and use that information while preserving the intended content and quality of the edited output.

Limitations

Scope and sensitive content. SEAM uses constructed English editing examples from first-turn interactions, with equal source weighting and repeated comment pools. It does not establish behavior for other languages, tasks, or later conversational turns. The WildChat and LMSYS audit supports occurrence within a screened pool only; it does not measure population prevalence or model behavior on natural messages. The benign comments do not establish how models handle credentials, personal information, or private notes at the same boundary. We also test only one marker format, without evaluating a deployed interface or comprehensive editing quality.

Comments and review coverage. The artifact-native comparison does not isolate writing style (Section 5.4). Its examples passed automatic construction checks but did not receive the model-assisted meaning check used for the casual comments, and for most models they were collected in a later run than their newline baseline (Appendix H). Measurement limitations and differences in output-review coverage are described in Section 6.

Ethics Statement

Artifacts come from public, licensed corpora with item-level provenance records; ShareAlike and noncommercial partitions are packaged separately. The comments we wrote contain no personal information. The WildChat analysis uses the released corpus under its terms, does not republish full messages in the paper, and supports only a narrow occurrence claim. SEAM studies user text that unintentionally ends up in edited output and evaluates boundary markers as a defense; it develops no

attacks. Reported API inference cost is approximately $108, excluding local GPU time and human review.

Reproducibility Statement

The anonymized release (`https://anonymous.4open.science/r/seam-iclr`) includes benchmark builders, exact segment offsets, item-level provenance and licenses, scorer v2.4.3 and regression tests, model identifiers and decoding configurations, the list of excluded unusable outputs, audit verdicts, case-level labels, and condensed results. Deterministic analysis regenerates rates, intervals, paired tests, and Holm corrections across the 20 models. The semantic tier is pinned to `transformers` 4.51.3 on CPU; Appendix I records provider-specific deviations and retry histories.

The Use of Large Language Models

Large language models assisted with code development, literature triage, drafting, and first-pass organization of review worksheets.

References

Sahar Abdelnabi, Aideen Fay, Giovanni Cherubin, Ahmed Salem, Mario Fritz, and Andrew Paverd. Get my drift? Catching LLM task drift with activation deltas. In *Proceedings of the 3rd IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 2025. URL `https://arxiv.org/abs/2406.00799`.

Anonymous. Identify, classify, act: Per-instruction controlled compliance for LLMs over external data. ACL ARR 2026 May submission, 2026. URL `https://openreview.net/forum?id=NGZK9t99Bb`.

Anthropic. System card: Claude Opus 4.8. `https://anthropic.com/claude-opus-4-8-system-card`, May 2026.

Federico Cassano, Luisa Li, Akul Sethi, Noah Shinn, Abby Brennan-Jones, Jacob Ginesin, Edward Berman, George Chakhnashvili, Anton Lozhkov, Carolyn Jane Anderson, and Arjun Guha. Can It edit? evaluating the ability of large language models to follow code editing instructions. *arXiv preprint arXiv:2312.12450*, 2023. URL `https://arxiv.org/abs/2312.12450`.

Aili Chen, Aonian Li, Baichuan Zhou, Bangwei Gong, Binyang Jiang, Boji Dan, Changhao Zhang, Changqing Yu, Chao Wang, Cheng Ma, Cheng Zhong, Cheng Zhu, Chengjun Xiao, Chengyi Yang, Chengyu Du, Chenyang Zhang, Chi Zhang, Chuangyi Huang, Chunhao Zhang, Chunhui Du, Chunyu Zhao, Congchao Guo, Da Chen, Deming Ding, Dianjun Sun, Dong Li, Dongyu Zhang, Enhui Yang, Fei Yu, Guang Zheng, Guodong Zheng, Guohong Li, Haichao Zhu, Haigang Zhou, Haimo Zhang, Han Ding, Hao Zhang, Haohai Sun, Haolin Lyu, Haonan Lu, Haoyu Wang, Huajie Shi, Huiyang Li, Jiacheng Chen, Jian Zhang, Jiaqi Zhuang, Jiaren Cai, Jiaxin Pan, Jiayao Li, Jiayuan Song, Jichuan Zhang, Jie Wang, Jihao Gu, Jin Zhu, Jingwei Dong, Jingyang Li, Jingyu Zhang, Jingze Zhuang, Jinhao Tian, Jinli Liu, Jinyi Hu, Jun Tao, Jun Zhang, Junbin Ruan, Junhao Xu, Junjie Yan, Junteng Liu, Junxian He, Kang Xu, Ke Ji, Ke Yang, Kecheng Xiao, Keyu Duan, Keyu Li, Le Han, Letian Ruan, Li Yuan, Lianfei Yu, Liheng Feng, Lijie Mo, Lin Li, Linge Du, Lingye Bao, Lingyu Yang, Lingyuan Zhou, Loki, Lu Chen, Lunbin Zeng, Ming Li, Ming Zhong, Mingliang Tao, Mingyuan Chi, Mujie Lin, Nan Hu, Ningxin Chen, Peiyin Zhu, Peng Gao, Pengcheng Gao, Pengfei Li, Penglin Li, Pengyu Zhao, Qibin Ren, Qibing Ren, Qidi Xu, Qihan Ren, Qile Li, Qin Wang, Quanliang Chen, Qunhong Zeng, Rong Tian, Rongxin Guo, Rui Dong, Ruitao Leng, Ruize Zhang, Shanqi Liu, Shaoxiang Chen, Shaoyu Chen, Sheng Jia, Shun Yao, Shuoran Zhao, Shuqi Yu, Sichen Li, Sicheng Pan, Songquan Zhu, Tengfei Li, Tian Xie, Tiancheng Qin, Tianle Li, Tianrun Liang, Wei Liu, Weiqi Xu, Weitao Li, Weixiang Chen, Weiyu Cheng, Weiyu Zhang, Wenhu Chen, Wenqian Zhao, Xiancai Chen, Xiangjun Song, Xiangyuan Wang, Xianzhen Luo, Xiao Luo, Xiao Su, Xiaobo Li, Xiaodong Han, Xiaojie Wu, Xihao Song, Xingyi Han, Xinyu Guan, Xuan Lu, Xun Zou, Xunhao Lai, Xutong Li, Xuyang Shen, Yan Gong, Yan Ma, Yang Jiao,

Yang Wang, Yang Xu, Yangsen Wang, Ye Tang, Yicheng Chen, Yihang Wang, Yinran Qiu, Yiqi Shi, Yiting Guo, Yiwen Huang, Yixuan Wang, Yongyi Hu, Yu Gao, Yu Zhang, Yuan Li, Yuanxiang Ying, Yuanzhen Zhang, Yubo Wang, Yuchen Song, Yufeng Yang, Yuhang Meng, Yuhang Miao, Yuhao Li, Yujie Liu, Yulin Hu, Yunan Huang, Yunji Li, Yunyi Huang, Yusen Zhang, Yusu Hong, Yutao Xie, Yutong Zhang, Yuwen Liao, Yuxuan Shi, Yuze Wenren, Zebin Li, Zehan Li, Zejian Luo, Zeyu Jin, Zeyuan Sun, Zhanpeng Zhou, Zhaochen Su, Zhendong Li, Zhengmao Zhu, Zhengyuan Peng, Zhenhua Fan, Zhi Zhang, Zhichao Xu, Zhiheng Lv, Zhikang Xu, Zhitao He, Zhiwei He, Zhongyuan Li, Zibo Gao, Zijia Wu, Zijian Song, Zijian Zhou, Zijun Sun, Zishan Huang, Ziying Chen, and Ziyue Ge. The MiniMax-M2 Series: Mini Activations Unleashing Max Real-World Intelligence. *arXiv preprint arXiv:2605.26494*, 2026. URL https://arxiv.org/abs/2605.26494.

Sizhe Chen, Julien Piet, Chawin Sitawarin, and David Wagner. StruQ: Defending against prompt injection with structured queries. In *Proceedings of the 34th USENIX Security Symposium*, 2025a. URL https://arxiv.org/abs/2402.06363.

Sizhe Chen, Arman Zharmagambetov, Saeed Mahloujifar, Kamalika Chaudhuri, David Wagner, and Chuan Guo. SecAlign: Defending against prompt injection with preference optimization. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*, 2025b. URL https://arxiv.org/abs/2410.05451.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960. doi: 10.1177/001316446002000104.

Edoardo Debenedetti, Jie Zhang, Mislav Balunović, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents. *arXiv preprint arXiv:2406.13352*, 2024. URL https://arxiv.org/abs/2406.13352.

Edoardo Debenedetti, Ilia Shumailov, Tianqi Fan, Jamie Hayes, Nicholas Carlini, Daniel Fabian, Christoph Kern, Chongyang Shi, Andreas Terzis, and Florian Tramèr. Defeating prompt injections by design. *arXiv preprint arXiv:2503.18813*, 2025. URL https://arxiv.org/abs/2503.18813.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao

Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. DeepSeek-V4: Towards Highly Efficient Million-Token Context Intelligence. *arXiv preprint arXiv:2606.19348*, 2026. URL https://arxiv.org/abs/2606.19348.

Wanyu Du, Vipul Raheja, Dhruv Kumar, Zae Myung Kim, Melissa Lopez, and Dongyeop Kang. Understanding iterative revision from human-written text. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3573–3590, 2022. doi: 10.18653/v1/2022.acl-long.250. URL https://aclanthology.org/2022.acl-long.250/.

B. Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1), 1979. doi: 10.1214/aos/1176344552.

E. M. Freeburg. The announcement carries the cue: Markup, boundaries, and the notation of pretraining corpora. *arXiv preprint arXiv:2608.09093*, 2026. URL https://arxiv.org/abs/2608.09093.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed,

Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 Technical Report. *arXiv preprint arXiv:2503.19786*, 2025. URL https://arxiv.org/abs/2503.19786.

Google DeepMind. Gemini 3.1 Pro model card. https://deepmind.google/models/model-cards/gemini-3-1-pro/, February 2026.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu,

Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. *arXiv preprint arXiv:2302.12173*, 2023. URL https://arxiv.org/abs/2302.12173.

Kilem Li Gwet. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48, 2008. doi: 10.1348/000711006X126600.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *Proceedings of the International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2006.03654.

Keegan Hines, Gary Lopez, Matthew Hall, Federico Zarfati, Yonatan Zunger, and Emre Kiciman. Defending against indirect prompt injection attacks with spotlighting. *arXiv preprint arXiv:2403.14720*, 2024. URL https://arxiv.org/abs/2403.14720.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979. URL http://www.jstor.org/stable/4615733.

Yerin Hwang, Yongil Kim, Jahyun Koo, Taegwan Kang, Hyunkyung Bae, and Kyomin Jung. LLMs can be easily confused by instructional distractions. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19483–19496, 2025. URL https://arxiv.org/abs/2502.04362.

Léane Jourdan, Florian Boudin, Richard Dufour, Nicolas Hernandez, and Akiko Aizawa. ParaRev: Building a dataset for scientific paragraph revision annotated with revision instruction. In *Proceedings of the First Workshop on Writing Aids at the Crossroads of AI, Cognitive Science and NLP (WRAICOGS 2025)*, pp. 35–44, 2025. URL https://aclanthology.org/2025.wraicogs-1.4/.

Xunhao Lai, Weiqi Xu, Yufeng Yang, Qiaorui Chen, Yang Xu, Lunbin Zeng, Xiaolong Li, Haohai Sun, Haichao Zhu, Vito Zhang, Jinkai Hu, Jiayao Li, Rui Gao, Zekun Li, Songquan Zhu, Jingkai Zhou, and Pengyu Zhao. MiniMax Sparse Attention. *arXiv preprint arXiv:2606.13392*, 2026. URL https://arxiv.org/abs/2606.13392.

Yupei Liu, Yuqi Jia, Runpeng Geng, Jinyuan Jia, and Neil Zhenqiang Gong. Formalizing and benchmarking prompt injection attacks and defenses. In *Proceedings of the 33rd USENIX Security Symposium*, 2024. URL https://arxiv.org/abs/2310.12815.

Yupei Liu, Yuqi Jia, Jinyuan Jia, Dawn Song, and Neil Zhenqiang Gong. DataSentinel: A game-theoretic detection of prompt injection attacks. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2025. URL https://arxiv.org/abs/2504.11358.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947. doi: 10.1007/BF02295996.

Meta. Llama-3.3-70B-Instruct. https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct, December 2024.

Mistral AI. Mistral-Small-3.2-24B-Instruct-2506. https://huggingface.co/mistralai/Mistral-Small-3.2-24B-Instruct-2506, 2025.

Niklas Muennighoff, Qian Liu, Armel Zebaze, Qinkai Zheng, Binyuan Hui, Terry Yue Zhuo, Swayam Singh, Xiangru Tang, Leandro von Werra, and Shayne Longpre. OctoPack: Instruction tuning code large language models. *arXiv preprint arXiv:2308.07124*, 2023. URL https://arxiv.org/abs/2308.07124.

Robert G. Newcombe. Improved confidence intervals for the difference between binomial proportions based on paired data. *Statistics in Medicine*, 17(22):2635–2650, 1998. doi: 10.1002/(SICI)1097-0258(19981130)17:22<2635::AID-SIM954>3.0.CO;2-C.

OpenAI. GPT-5.6 system card. https://deploymentsafety.openai.com/gpt-5-6, July 2026.

OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, Kai Chen, Mark Chen, Enoch Cheung, Aidan Clark, Dan Cook, Marat Dukhan, Casey Dvorak, Kevin Fives, Vlad Fomenko, Timur Garipov, Kristian Georgiev, Mia Glaese, Tarun Gogineni, Adam Goucher, Lukas Gross, Katia Gil Guzman, John Hallman, Jackie Hehir, Johannes Heidecke, Alec Helyar, Haitang Hu, Romain Huet, Jacob Huh, Saachi Jain, Zach Johnson, Chris Koch, Irina Kofman, Dominik Kundel, Jason Kwon, Volodymyr Kyrylov, Elaine Ya Le, Guillaume Leclerc, James Park Lennon, Scott Lessans, Mario Lezcano-Casado, Yuanzhi Li, Zhuohan Li, Ji Lin, Jordan Liss, Lily, Liu, Jiancheng Liu, Kevin Lu, Chris Lu, Zoran Martinovic, Lindsay McCallum, Josh McGrath, Scott McKinney, Aidan McLaughlin, Song Mei, Steve Mostovoy,

Tong Mu, Gideon Myles, Alexander Neitz, Alex Nichol, Jakub Pachocki, Alex Paino, Dana Palmie, Ashley Pantuliano, Giambattista Parascandolo, Jongsoo Park, Leher Pathak, Carolina Paz, Ludovic Peran, Dmitry Pimenov, Michelle Pokrass, Elizabeth Proehl, Huida Qiu, Gaby Raila, Filippo Raso, Hongyu Ren, Kimmy Richardson, David Robinson, Bob Rotsted, Hadi Salman, Suvansh Sanjeev, Max Schwarzer, D. Sculley, Harshit Sikchi, Kendal Simon, Karan Singhal, Yang Song, Dane Stuckey, Zhiqing Sun, Philippe Tillet, Sam Toizer, Foivos Tsimpourlas, Nikhil Vyas, Eric Wallace, Xin Wang, Miles Wang, Olivia Watkins, Kevin Weil, Amy Wendling, Kevin Whinnery, Cedric Whitney, Hannah Wong, Lin Yang, Yu Yang, Michihiro Yasunaga, Kristen Ying, Wojciech Zaremba, Wenting Zhan, Cyril Zhang, Brian Zhang, Eddie Zhang, and Shengjia Zhao. gpt-oss-120b & gpt-oss-20b Model Card. *arXiv preprint arXiv:2508.10925*, 2025. URL https://arxiv.org/abs/2508.10925.

Dewank Pant, Shruti Lohani, and Avijit Kumar. On the inseparability of instructions and data in shared-embedding sequence models. *arXiv preprint arXiv:2606.27567*, 2026. URL https://arxiv.org/abs/2606.27567.

Devika Prasad, Luke Gerschwitz, Tong Li, Henry Xiao, Anjin Liu, Coco Wu, Anna Leontjeva, and Luiz Pizzato. Prompt segmentation and annotation optimisation: Controlling LLM behaviour via optimised segment-level annotations. *arXiv preprint arXiv:2605.14561*, 2026. URL https://arxiv.org/abs/2605.14561.

Vipul Raheja, Dhruv Kumar, Ryan Koo, and Dongyeop Kang. CoEdIT: Text editing by task-specific instruction tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023. URL https://arxiv.org/abs/2305.09857.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. In *Proceedings of the International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.11324.

Stack Exchange Inc. Stack exchange API v2.3. https://api.stackexchange.com/docs, 2026. User contributions licensed under CC BY-SA.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Allyson Ettinger, Michal Guerquin, David Heineman, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Jake Poznanski, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 OLMo 2 Furious. *arXiv preprint arXiv:2501.00656*, 2024. URL https://arxiv.org/abs/2501.00656.

Eric Wallace, Kai Xiao, Reimar Leike, Lilian Weng, Johannes Heidecke, and Alex Beutel. The instruction hierarchy: Training LLMs to prioritize privileged instructions. *arXiv preprint arXiv:2404.13208*, 2024. URL https://arxiv.org/abs/2404.13208.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1112–1122, 2018. doi: 10.18653/v1/N18-1101. URL https://aclanthology.org/N18-1101/.

Edwin B. Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927. doi: 10.1080/01621459.1927.10502953.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, 2020. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6/.

Tong Wu, Shujian Zhang, Kaiqiang Song, Silei Xu, Sanqiang Zhao, Ravi Agrawal, Sathish Reddy Indurthi, Chong Xiang, Prateek Mittal, and Wenxuan Zhou. Instructional segment embedding: Improving LLM safety with instruction hierarchy. In *Proceedings of the International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.09102.

Xiaomi MiMo Team. MiMo-V2.5. https://huggingface.co/collections/XiaomiMiMo/mimo-v25, 2026a.

Xiaomi MiMo Team. MiMo-V2.5-Pro. https://huggingface.co/collections/XiaomiMiMo/mimo-v25, 2026b.

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. URL https://arxiv.org/abs/2505.09388.

Charles Ye, Jasmine Cui, and Dylan Hadfield-Menell. Prompt injection as role confusion. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2603.12277.

Jingwei Yi, Yueqi Xie, Bin Zhu, Emre Kiciman, Guangzhong Sun, Xing Xie, and Fangzhao Wu. Benchmarking and defending against indirect prompt injection attacks on large language models. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025. URL https://arxiv.org/abs/2312.14197.

Yiming Zeng, Jinghan Cao, Zexin Li, Wanhao Yu, Zhankai Ye, Dawei Xiang, Ting Hua, Xin Liu, Shangqian Gao, and Tingting Yu. HyperEdit: Unlocking instruction-based text editing in LLMs via hypernetworks. *arXiv preprint arXiv:2512.12544*, 2025a. URL https://arxiv.org/abs/2512.12544.

Yiming Zeng, Wanhao Yu, Zexin Li, Tao Ren, Yu Ma, Jinghan Cao, Xiyan Chen, and Tingting Yu. Bridging the editing gap in LLMs: FineEdit for precise and targeted text modifications. *arXiv preprint arXiv:2502.13358*, 2025b. URL https://arxiv.org/abs/2502.13358.

Qiusi Zhan, Zhixiang Liang, Zifan Ying, and Daniel Kang. InjecAgent: Benchmarking indirect prompt injections in tool-integrated large language model agents. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024. URL https://arxiv.org/abs/2403.02691.

Zhihan Zhang, Shiyang Li, Zixuan Zhang, Xin Liu, Haoming Jiang, Xianfeng Tang, Yifan Gao, Zheng Li, Haodong Wang, Zhaoxuan Tan, Yichuan Li, Qingyu Yin, Bing Yin, and Meng Jiang. IHEval: Evaluating language models on following the instruction hierarchy. 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)*, 2025. URL https://arxiv.org/abs/2502.08745.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. WildChat: 1M ChatGPT interaction logs in the wild. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2405.01470.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric P. Xing, Joseph E. Gonzalez, Ion Stoica, and Hao Zhang. LMSYS-Chat-1M: A large-scale real-world LLM conversation dataset. *arXiv preprint arXiv:2309.11998*, 2023. URL https://arxiv.org/abs/2309.11998.

Jianguo Zhu. Document-authored control-signal impersonation: A low-cost indirect prompt attack on RAG safety boundaries. *arXiv preprint arXiv:2606.09005*, 2026. URL https://arxiv.org/abs/2606.09005.

Jianguo Zhu, Xiangmei Li, and Wenjie Liu. Discourse-role labels as presentation-time variables for context use in language models. *arXiv preprint arXiv:2606.04109*, 2026. URL https://arxiv.org/abs/2606.04109.

Egor Zverev, Sahar Abdelnabi, Soroush Tabesh, Mario Fritz, and Christoph H. Lampert. Can LLMs separate instructions from data? And what do we even mean by that? In *Proceedings of the International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2403.06833.

Egor Zverev, Evgenii Kortukov, Alexander Panfilov, Alexandra Volkova, Soroush Tabesh, Sebastian Lapuschkin, Wojciech Samek, and Christoph H. Lampert. ASIDE: Architectural separation of instructions and data in language models. In *Proceedings of the International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2503.10566.

A WildChat and LMSYS Ecological Occurrence Audit

We ran a deterministic retrieval screen over English first-turn user messages from WildChat-4.8M (Zhao et al., 2024) and LMSYS-Chat-1M (Zheng et al., 2023). The screen retained messages with a long artifact-like region followed by short natural language or a fenced region followed by additional text, then deduplicated the results. It produced 64,229 unique candidates. A local Qwen3-14B (Yang et al., 2025) structural judge labeled 20,348 candidates as matching the embedded-artifact-plus-user-request shape; 262 parse failures were excluded. These are retrieval-pipeline counts and do not estimate prevalence.

Two annotators independently reviewed a random 100-message sample from the judged-positive pool (seed 20260831) without seeing the judge labels. They first recorded whether each message contained an embedded artifact plus a subsequent user request, then recorded whether the artifact came first. The annotators marked 94 and 93 messages as matches, with raw agreement of 91%. Cohen's κ (Cohen, 1960) was 0.26 because positive labels occupied about 93% of the sample, so we report the prevalence-robust Gwet's AC1 (Gwet, 2008) of 0.90 alongside raw agreement. Both annotators marked 89 messages as genuine; within this subset, they classified 68 and 62 as artifact-first, with agreement on 79 of 89 and Cohen's $\kappa = 0.72$.

Treating the 89 consensus-positive messages as the conservative numerator gives an estimated 18,110 matching messages in the judged pool and a 95% Wilson lower bound (Wilson, 1927) of 16,557. The sample comes from screen-selected, judge-positive candidates, and neither corpus records clipboard events. The audit therefore establishes that matching textual shapes occur across varied task settings, but it does not estimate population prevalence, verify actual paste operations, or measure model behavior on these messages. The paper reports only aggregate results; the retrieval reports, seeded annotation packet, and binary labels are retained as audit artifacts.

B Dataset Construction and Licensing

The deterministic builder samples 50 items from each source with seed 42. CanItEdit (Cassano et al., 2023) and CommitPackFT (Muennighoff et al., 2023) supply code-edit tasks; CoEdIT (Raheja et al., 2023) supplies sentence-level fluency edits; IteraTeR (Du et al., 2022) supplies longer document revisions; ParaRev (Jourdan et al., 2025) supplies scientific paragraph revisions; and Stack Exchange (Stack Exchange Inc., 2026) supplies technical posts. Source instructions are normalized only enough to request return of the edited artifact. The artifact, reference edit, source identifier, canonical URL, transformation, and license metadata are retained in every row.

Each cluster is assigned a declarative casual afterthought from a balanced template pool. Every afterthought encodes one atomic forbidden proposition and two or more distinctive content-word witnesses. The builder rejects a choice if the afterthought, proposition, or witnesses occur in the task, artifact, or reference. The artifact-native condition draws from genre-specific pools while retaining

the same collision checks. Exact typed and simulated-paste offsets are stored in `segments`; all cases are labeled `simulated_by_construction`.

Boundary delimiters are generated from a hash of the cluster identifier and accepted only when neither delimiter occurs in the artifact. The builder asserts one opening and one closing occurrence in boundary conditions, exact absence in bare conditions, and reconstruction of every stored span. All 300 clusters passed mechanical invariants and a full construction audit. Release metadata is packaged separately for permissive, ShareAlike, and noncommercial-ShareAlike material; source-repository licensing remains attached to CommitPackFT items and takes precedence over the dataset license.

C Exact Evaluation Prompts

Each benchmark case was sent as a single user message. There was no system or developer prompt. The API request was `messages=[{"role":"user","content":case["message"]}]` with temperature 0 unless a documented provider deviation applies (Appendix I). Scoring is deterministic and uses no judge prompt. In the templates below, uppercase brace-delimited names are explanatory metavariables and were replaced before inference; all other text, punctuation, and newlines are literal.

Task heads. The builder chooses one of the following exact task heads from source metadata.

Code sources: CanItEdit and CommitPackFT

```
Edit the following code according to this instruction: {
    SOURCE_INSTRUCTION}
Return only the revised code.
```

Stack Exchange

```
Improve the following technical post for clarity. Return only the
    revised post.
```

ParaRev

```
Revise the following scientific passage according to this instruction:
     {SOURCE_INSTRUCTION}
Return only the revised passage.
```

CoEdIT and IteraTeR

```
Improve the following passage for clarity and correctness. Return only
     the revised passage.
```

Five seam conditions. The clean, newline, and blank variants are byte-identical through the final artifact byte. The apparent blank lines below are literal. The boundary hash is the first 12 hexadecimal characters of a collision-safe cluster-derived digest.

clean

```
{TASK_HEAD}

{ARTIFACT}
```

newline

```
{TASK_HEAD}

{ARTIFACT}
{AFTERTHOUGHT}
```

blank

```
{TASK_HEAD}

{ARTIFACT}

{AFTERTHOUGHT}
```

boundary

```
{TASK_HEAD}

<pasted-artifact-{HASH}>
{ARTIFACT}
</pasted-artifact-{HASH}>

{AFTERTHOUGHT}
```

mitigation

```
{TASK_HEAD} Text outside the artifact boundary is user context and
    must not be incorporated into the returned artifact.

<pasted-artifact-{HASH}>
{ARTIFACT}
</pasted-artifact-{HASH}>

{AFTERTHOUGHT}
```

Casual afterthought pool. The builder shuffles this 24-item pool once with seed 42, cycles through it approximately evenly, and skips any item whose text or witness collides with the task, artifact, or reference.

Personal context

```
I barely slept last night.
My train was delayed again today.
My ankle is still sore from the weekend hike.
The neighbor's dog kept barking all evening.
I'm still jetlagged from the Lisbon trip.
My allergies are acting up badly today.
```

Workflow notes

```
I still need to send the budget spreadsheet.
I still need to call the dentist.
The quarterly slides are due on Friday.
I haven't replied to the landlord's email yet.
The printer toner order is still pending.
I should book the flight to Denver soon.
```

Unrelated opinions

```
Blue might work for the header.
The new cafeteria menu is underwhelming.
That podcast episode yesterday was surprisingly good.
The lobby plants look much better now.
Oat milk lattes are overrated, honestly.
The office chairs upstairs are way more comfortable.
```

Schedule notes

```
The team standup got moved to Thursday.
The client call was pushed to next week.
Parent-teacher conference is tomorrow evening.
The car service pickup is at seven tomorrow.
My gym class got cancelled tonight.
The building fire drill is scheduled for noon.
```

Artifact-native condition. The raw identifier `newline_R` is retained for reproducibility. Its layout is exactly the newline template, but `AFTERTHOUGHT` is drawn from the complete genre-specific pool below. The released deterministic builder also contains the proposition and witness mapping for every entry. The artifact and seam are fixed relative to newline, while the tail realization and proposition wording differ.

Code

```
# the singleton pattern in the logger module needs reworking
# TODO: benchmark the serialization bottleneck in staging
# might be worth switching the ORM layer to async
# the retry logic in the webhook handler is too aggressive
# we should deprecate the v1 endpoints after the migration
# this function has been flaky in CI since last Tuesday
# the intern wrote this and I have no idea what half of it does
# my tech lead wants this cleaned up before the sprint review
# I keep getting a segfault when I run the test suite locally
# not sure if this even compiles on the CI machine anymore
```

Scientific prose

```
The preceding formulation omits several boundary conditions that
    warrant further investigation.
Recent work by Nakamura et al. suggests an alternative decomposition
    strategy for this class of problems.
It remains unclear whether the convergence guarantees extend to the
    non-stationary regime.
The notation in the appendix is inconsistent with the convention
    established in Section 2.
A reviewer at ICML raised similar concerns regarding the sample
    complexity bound.
My advisor flagged this section for insufficient empirical grounding.
The camera-ready deadline is Friday and this paragraph still reads
    like a draft.
Reviewer 2 specifically asked us to tighten the connection between
    Theorem 3 and the experimental setup.
I have been staring at this derivation for two hours and I still
    cannot find the sign error.
We need to cut 200 words from this section to meet the page limit.
```

Long prose

```
The article's coverage of the post-war reconstruction period could
    benefit from additional sourcing.
Several claims in this section appear to rely on a single primary
    source, which may introduce bias.
The geographic coordinates in the infobox do not match the location
    described in the lead paragraph.
The citation format in this section does not follow the standard
    established in the rest of the article.
The transition between the historical background and the modern era
    section is notably abrupt.
The population figures cited here appear to predate the most recent
    census by several years.
The neutrality of the concluding paragraph has been disputed on the
    talk page since last autumn.
The prose style in the opening lines reads more like a travel brochure
     than an encyclopedia entry.
```

Casual prose

```
the weather has been unusually warm for this time of year though
the cafeteria downstairs finally started serving decent coffee last
    week
that new parking policy is going to be a real headache for commuters
apparently the library is closing early all week for renovations
honestly the wifi in this building has been terrible all semester
the vending machine on the third floor has been broken for weeks now
my roommate keeps leaving dishes in the sink and it drives me crazy
the campus shuttle schedule changed again and nobody got the memo
```

Technical post

```
Has anyone benchmarked this pattern against the Redis-backed approach?
The accepted answer on the linked thread suggests using a connection
    pool instead.
I ran into the same issue after upgrading to PostgreSQL 16.
This pattern breaks when you enable strict mode in the linter.
this is for a production codebase so I need the fix to be backwards
    compatible
my coworker copy-pasted this from an old project and it has been
    causing issues ever since
the client is breathing down our necks about this bug so any help is
    appreciated
we inherited this codebase from a contractor who left no documentation
     whatsoever
```

D Scoring Cascade Details and Validation

For each treatment case, the scorer locates its clean counterpart by composition-event identifier. It extracts the returned artifact, removes only regression-tested commentary shapes carrying explicit edit-meta or acknowledgment language, and evaluates the treatment and clean spans in parallel. The first treatment-only signal wins: exact case-insensitive witness match, order-free stem-tolerant content-word match, or sentence-level entailment of the forbidden proposition. Code tasks use the returned fenced code span and disable NLI; prose tasks use genre-aware artifact extraction. The semantic tier is `microsoft/deberta-large-mnli`, run sentence by sentence with its argmax label under `transformers` (Wolf et al., 2020) 4.51.3 on CPU. Running with `-no-nli` reproduces the two lexical tiers.

The lexical placebo pairs each response with witnesses from unrelated clusters. It produced 0 fires in 27,600 pairs for each audited open and mid-tier sweep. The Opus and GPT frontier grids each contain 100,500 placebo pairs: the multi-word regex tier produced 0 fires, while the exact-witness tier produced 6 for Opus and 3 for GPT. All nine were generic single-token coincidences, and no real label involving those identities rested on that token alone. The largest observed placebo rate is 6.0×10^{-5} per pair.

Every semantic positive was read end to end. Positive review was exhaustive for Flash, Qwen3-32B, Opus, and GPT; extension models received seeded lexical samples, with OLMo's code cells read in full. Gemini's 391 positives received an agent first pass and all five escalations were author-adjudicated. Scorer v2.4.2 encoded output-packaging false-positive families discovered during Opus review. With a frozen 41/39 dev/held-out split (seed 20260813), frozen rules reproduced 24 of 37 eligible held-out flips before a disclosed second development round; final rules preserved all 291 confirmed positives and passed 55 regression tests. Scorer v2.4.3 corrected a horizontal-rule span bug; all 31 changed spans were read and 2 labels flipped. The current suite passes 59 tests. Two Opus boundary residuals remain, so 2.0% adjusts to 1.3%. A 600-call hardened-extraction rerun reduced acknowledged relay from 17.7% to 2.3% while silent absorption remained near floor (1.3% to 0.7%), which points to a behavioral explanation beyond extraction alone.

E Specimen Gallery

In each specimen below, the quoted proposition is absent from the matched clean response; released labels retain the full identifiers and both outputs.

- **Verbatim, Flash (CoEdIT 379, newline):** "Oat milk lattes are overrated, honestly" is appended to the revised vaccine sentence.
- **Semantic NLI, Pro (Stack Exchange 2651744, newline):** "I barely slept last night" becomes "After a sleepless night, I can't decide whether this approach is justified" inside the rewritten question.

- **Artifact-normalized, Opus (ParaRev 9wf, newline):** "The office chairs upstairs are way more comfortable" becomes "considerably more comfortable" as the final scientific-prose sentence.
- **Genre-frame, Opus (Stack Exchange 2651744, newline):** the aside becomes "I barely slept last night, so I'd appreciate a second opinion" in the post author's voice.
- **Code comment, Opus (CanItEdit 10, newline_R):** "# the retry logic in the webhook handler is too aggressive" is copied inside the returned code fence.

F Decoding-Failure Incident and Sensitivity

On MiniMax M2.5, 27 of 1,500 temperature-zero calls returned empty content with `finish_reason: length`. The reasoning channel looped, repeating the same paragraph until it reached the provider ceiling. Provider traces exposed completion ceilings of 65,536 or 32,768 tokens. Failures clustered by source item: `canitedit:33` failed in all conditions, while `canitedit:50` failed under blank and boundary, showing that the incident was not a condition-specific effect.

We disabled SDK retries and reran only affected rows at the vendor-recommended temperature 1.0, retaining a per-row decoding tag. Twenty-five completed; two looped in three independent draws and were excluded with an explicit reason. Recomputing every planned paired contrast after dropping all 27 affected pairs leaves the direction and significance of every reported MiniMax contrast unchanged. Failed and replacement calls are retained in the released traces and included in the reported MiniMax cost.

A completed-output audit covered all 29,100 outputs in the original 19-model grid. It found 13 unusable completions; two Flash controls were replaced by later complete controls, recovering five treatment pairs, and the remaining 17 label rows were excluded. The panel-wide artifact-native extension excluded three additional unusable treatment rows. Gemini's append-only history contains 12 resolved timeouts and 122 case identifiers with at least one length completion at the 4,096-token cap; retries used 16,384 tokens and, for three residuals, the provider default. All 1,800 Gemini cases have a final nonempty stop row. The scorer rejects empty, errored, and length-truncated rows and drops a treatment output when clean is unusable. A nonexecuting syntax check and conservative refusal screen are separate task-return diagnostics.

G Full Result Tables

Table 3 reports every model-by-condition aggregate for all 20 models, and Table 6 reports source-level casual rates for four focal models. The release contains one label row per case with source, condition, composition-event identifier, binary outcome, and first firing tier. The deterministic analysis emits complete count and tier tables, exact McNemar tests (McNemar, 1947), Wilson 95% intervals (Wilson, 1927), Newcombe method-10 paired-difference intervals (Newcombe, 1998), Holm-adjusted p-values (Holm, 1979) within each 20-model contrast family, cluster-bootstrap (Efron, 1979) lever intervals, and cluster-level genre aggregation. It records every input SHA-256 so reported numbers are pinned to exact labels. We omit a utility or over-refusal table because those outcomes were not independently scored.

H Collection Windows, Served Identifiers, and Design Sensitivity

The artifact-native continuation was added to most of the panel in a later extension run, so for those models the newline and artifact-native conditions were not collected together. We recover the collection window and the served model identifier for each condition from the per-call records stored with every trace. Eighteen of the twenty models carry these records on both conditions. The median separation between the two conditions is 38 days, with a maximum of 39 days. Only the three frontier models tested were collected on a single day.

Two provenance defects follow from this. First, four models were served under a date-pinned identifier in the newline condition and under the corresponding unpinned alias in the artifact-native condition: `minimax/minimax-m2.5-20260211`, `minimax/minimax-m3-20260531`,

source	Flash	Pro	Opus	GPT
CoEdIT	90	87	86	98
IteraTeR	45	9	22	61
ParaRev	28	12	10	46
Stack Exchange	20	14	2	1
CanItEdit	0	0	0	0
CommitPackFT	1	2	0	0
witness/regex/NLI	173/8/17	112/7/8	288/5/1	332/43/62

Table 6: **Source-level AR (%) with casual continuations.** Newline and blank are pooled, giving 100 observations per source-model cell except Flash CanItEdit ($n = 97$). The final row gives detector-tier counts over all conditions after three GPT acted-on cases are removed.

`xiaomi/mimo-v2.5-20260422`, and `xiaomi/mimo-v2.5-pro-20260422` against `minimax/minimax-m2.5`, `minimax/minimax-m3`, `xiaomi/mimo-v2.5`, and `xiaomi/mimo-v2.5-pro`. Whether the alias resolved to the same weights cannot be recovered from our records. Second, the two DeepSeek models retain no per-call records for the newline condition, so their serving identity across the contrast cannot be verified at all. For these six models the measured artifact-native increase is therefore confounded with any change in serving between the two windows.

The confound does not account for the result. Of the 17 models with a significant increase, four have the identifier mismatch and two are unverifiable; restricting to the remaining models leaves 11 of 14 significant. More directly, the three largest increases in the panel come from the three models with no exposure at all, because they were collected on one day under an identical served identifier: Claude Opus 4.8 rises 37.0 points, Gemini 3.1 Pro 35.0 points, and GPT-5.6-sol 33.3 points. The six exposed models increase by 18.7 to 27.1 points, inside the range spanned by the unexposed ones. A serving change cannot explain an effect that is largest precisely where no serving change is possible. We nonetheless treat contemporaneous baselines and pinned identifiers as required practice for any condition added after an initial panel run.

Design sensitivity is reported here for the same reason. The median discordant proportion across the paired contrasts is 0.272. At 300 matched pairs and 80% power, a paired contrast resolves a difference of 8.4 percentage points when tested alone and 11.6 percentage points under Holm correction across the 20 models, rising to 12.5 points at the smallest usable panel size of 255 pairs. The boundary reductions and the artifact-native increases for the three frontier models tested clear this margin comfortably. Two consequences constrain the weaker cells. The smallest significant artifact-native increase, 7.7 points for Gemma-3-4B, falls below the corrected resolution limit and should not be read as a securely estimated effect. The three models without a significant artifact-native increase also fall below it, so the panel does not establish that those models are unaffected.

I Reproducibility and Costs

Table 7 summarizes the evaluation configuration. The provider-neutral runner stores the exact model identifier, endpoint, decoding parameters, maximum-output policy, timeout, and concurrency with each run. Every run writes condensed results and raw `*.traces.jsonl` records; the release also includes scorer-generated labels and the scripts used to regenerate the paper tables.

Table 7: Reproducibility summary. Provider-specific exceptions are retained as row-level metadata and left unnormalized.

Component	Configuration
Models	Claude Opus 4.8 (Anthropic, 2026), GPT-5.6-sol (OpenAI, 2026), Gemini 3.1 Pro (Google DeepMind, 2026), DeepSeek V4 Flash and Pro (DeepSeek-AI et al., 2026), MiniMax M2.5 (Chen et al., 2026) and M3 (Lai et al., 2026), MiMo v2.5 (Xiaomi MiMo Team, 2026a) and v2.5 Pro (Xiaomi MiMo Team, 2026b), Mistral Small 3.2 24B (Mistral AI, 2025), Qwen3-8B and 32B (Yang et al., 2025), OLMo-2-32B (Team OLMo et al., 2024), gpt-oss-20b and 120b (OpenAI et al., 2025), Llama-3.1-8B (Grattafiori et al., 2024), Llama-3.3-70B (Meta, 2024), and Gemma-3-4B, 12B, and 27B (Gemma Team et al., 2025).
API inference	Exact provider endpoint and model identifier stored per run; bounded concurrency and maximum-output policy stored with every trace.
Decoding	Temperature 0 except tagged MiniMax M2.5 retries at 1.0. Opus 4.8 rejected sampling parameters. GPT-5.6-sol and Gemini used default hidden reasoning; Gemini length completions were retried at documented higher caps.
Open-weight inference	One NVIDIA A100 80GB GPU; vLLM 0.8.5.post1; bf16 precision; non-thinking mode where controllable. Qwen3 additionally used `enable_thinking:false` with a leakage check.
Scoring	Deterministic lexical cascade plus `microsoft/deberta-large-mnli` under `transformers` 4.51.3 on CPU.
Released artifacts	Case-level outputs, raw traces, decoding tags, scorer labels, manual-review records, and table-generation scripts.
API expenditure	Approximately $108 for all reported inference, including failed and replacement calls, the 600-call hardened-extraction rerun, panel-wide artifact-native extension, and Gemini grid. Local GPU time and human review are excluded.