EXACT FEEDBACK IS NOT CONTROL: EVALUATING TEXT-BASED CLOSED-LOOP REVISION IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Closed-loop revision is increasingly adopted in large language model applications to iteratively reduce mismatches between generated outputs (e.g., a paragraph) and user targets (e.g., required paragraph length). However, revision failures may stem from incomplete feedback or models failing to act effectively on correct feedback, making it difficult to systematically assess the revision capabilities of LLMs. In this work, we propose a fixed-budget revision protocol in which deterministic verifiers report all remaining violations across three constraint families: exact-length, lexical, and compositional. This holds feedback correctness and completeness fixed, allowing us to isolate and evaluate model-side closed-loop revision. We conduct experiments on 19 mainstream open-source and closed-source models, including Llama 3.1 8B and GPT-5.6 Sol, and observe large cross-model and cross-constraint differences in final joint success, with controller-level mean final joint success ranging from 17.4% to 99.8%. Substantial cross-model gaps persist when the same initial draft is used. In controlled experiments, we find reproducible model-specific differences in how exact feedback is translated into revisions, while post-training and model scale reshape these responses without consistently bringing them closer to exact correction. Interestingly, across all constraint families, failure trajectories often repeat earlier outputs, and prior recurrence is associated with substantially lower subsequent recoverability. Finally, matched-state interventions across constraint families show that removing earlier dialogue, with the current draft and feedback fixed, changes recurrence escape without reliably improving final success; effects depend on the model, task, and trigger-state composition. Overall, exact feedback makes revision errors observable, but does not make the closed loop reliable. Code and data are available anonymously[1].

1 INTRODUCTION

Feedback-driven revision is increasingly used in large language model (LLM) systems to bring generated outputs closer to user requirements. Such systems revise behavior across trials (Shinn et al., 2023), modify code using execution results (Yang et al., 2024), replan from environmental observations (Huang et al., 2023), and refine candidate programs using evaluator scores (Novikov et al., 2025). In each setting, feedback identifies a gap between the current output and the target, and LLMs must decide how to act on it repeatedly and ultimately complete the task successfully.

However, revision can be unreliable due to a lack of reliable external feedback in reasoning (Huang et al., 2024) and inaccurate model-generated feedback by itself (Olausson et al., 2024). Studies show that models can persist with incorrect answers despite critiques (Jiang et al., 2025) and surveys further emphasize the role of feedback quality (Kamoi et al., 2024). When feedback itself may be unreliable, poor revision outcomes do not reveal whether the bottleneck lies in the feedback or in the model's response to it. This leaves open how reliably LLMs can act on feedback over repeated revisions when that feedback is complete and correct.

In this work, we propose a trajectory-level framework to evaluate fixed-budget closed-loop revision in LLMs, as illustrated in Figure 1. We study revision under explicit, deterministically checkable constraints and use verifiers that report all remaining violations after each revision. The same frozen

[1] https://anonymous.4open.science/r/exact-feedback-9700/

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.

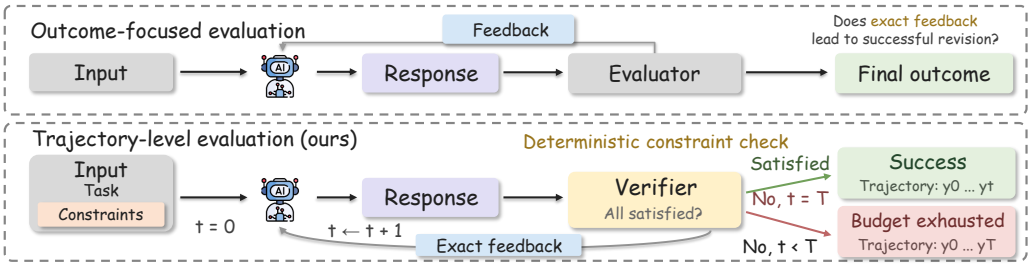

Figure 1: Two views of LLM revision: outcome-focused evaluation (top) and trajectory-level evaluation under exact feedback (bottom, this work).

rules generate feedback and determine final success, making feedback correctness and completeness controlled experimental conditions and allowing us to isolate model-side revision.

In particular, we build a reusable evaluation suite spanning three text-based constraint families—exact-length, lexical, and compositional—and use it to evaluate 19 mainstream open- and closed-source models. We first examine cross-model and cross-task variation, including fixed-draft comparisons that hold the starting response constant. We then characterize model-specific feedback-to-action responses under controlled correction requests, and test whether these patterns replicate on unseen texts and remain informative during independent multi-round revision. Finally, we analyze recurrent failure trajectories and use matched-state history interventions to examine how retained dialogue shapes subsequent editing behavior and final success.

Overall, our contributions and findings can be summarized as follows:

- **An auditable framework and evaluation suite for isolating model-side revision.** We introduce a fixed-budget request–model–verifier loop for closed-loop revision. Deterministic verifiers report all remaining violations and use the same frozen rules to determine final success, making feedback correctness and completeness controlled experimental conditions. The accompanying evaluation suite provides frozen cases, trajectory-level records, and generation and analysis pipelines for reproducible evaluation.
- **A systematic characterization of closed-loop revision behavior.** We evaluate 19 model configurations across three text-based constraint families and find substantial cross-model and cross-constraint variation. Large revision gaps persist from the same starting responses, showing that they are not simply inherited from differences in initial generation.
- **A controlled analysis of feedback-to-action responses.** By holding drafts fixed and varying exact correction requests, we identify reproducible, model-specific response patterns and show that post-training and model scale reshape these behaviors in different ways. These response patterns remain informative during independent multi-round revision.
- **A trajectory-level analysis of output recurrence and history effects.** Failed revision trajectories frequently revisit earlier outputs across all three constraint families. Matched history interventions substantially alter escape from recurrence and subsequent editing behavior, while producing much smaller and heterogeneous changes in final success.

2 Related Work

Iterative refinement and self-correction. Self-Refine alternates critique and revision using the same model, without additional training (Madaan et al., 2023). CRITIC grounds critiques in external tool feedback (Gou et al., 2024), while SCoRe trains multi-turn self-correction through reinforcement learning (Kumar et al., 2025). These approaches improve revision through different feedback sources or changes to the revision policy. Their effectiveness depends on the setting: reasoning studies find that self-correction without external feedback can fail to improve answers (Huang et al., 2024), and code-repair studies identify feedback quality as a limitation (Olausson et al., 2024). A broader survey likewise emphasizes the conditions under which feedback supports correction (Kamoi et al., 2024).

Closer to our question, Feedback Friction finds that models can persist with incorrect answers despite targeted critiques generated with access to ground-truth answers (Jiang et al., 2025). RefineBench reports substantial cross-model variation in refinement performance (Lee et al., 2026). These studies leave open whether cross-model refinement gaps persist when the starting draft and complete, exact feedback are held fixed, and whether such gaps reflect reproducible differences in how models translate the same feedback into revisions.

Exact verification and constrained revision. Constrained decoding enforces lexical and logical requirements during generation (Lu et al., 2021), while length-control methods target outputs of a prescribed length (Li et al., 2024). We study how a revision policy acts on exact feedback under standard decoding. Reliable verification has also been used for repeated correction. On arithmetic, graph coloring, and planning, Stechly et al. (2025) compare sound external feedback at different levels of detail with repeated sampling. DeCRIM decomposes multi-constraint instructions and uses critiques to guide revision, including rule-based feedback on IFEval (Ferraz et al., 2024). Our controlled experiments hold drafts fixed and vary accurate correction requests, measuring the resulting edits from the input–output perspective of system identification (Ljung, 1999). We test whether these responses replicate on held-out texts and remain informative during independent revision, then examine failure trajectories and history effects under complete violation reports.

Closed-loop behavior and history. Trajectory studies track answers repeatedly in question answering (Jiang et al., 2025), correctness transitions during self-correction (Liu & Meng, 2026), and textual fixed points during repeated abstract polishing (Wu et al., 2026). Our analysis follows repeated full outputs while explicit constraints remain unsatisfied, connecting trajectory behavior to constraint satisfaction as in closed-loop verification (Wang et al., 2020). History also matters when instructions are gradually revealed: models can rely too heavily on earlier answers (Laban et al., 2026). We test history's role after task requirements are fully specified. By holding the current draft and verifier report fixed, we distinguish the effect of earlier dialogue from differences in the text being revised or the feedback it receives. We measure intervention effects on the next revision and final success separately, testing whether gains in immediate behavior are accompanied by gains in task completion.

3 A Unified Closed-Loop Revision Protocol

Problem definition. Given a task x and an initial draft y_0, we study whether an LLM can satisfy a set of verifiable constraints $\mathcal{C}_x$ within a budget of T revisions. Joint success requires every constraint to hold in the same draft:

$$J(x, y) = \prod_{c \in \mathcal{C}_x} c(y), \tag{1}$$

where each constraint is a binary check: $c(y) = 1$ if draft y satisfies c, and $c(y) = 0$ otherwise. Thus $J(x, y) = 1$ indicates joint success.

Let y_t denote the draft after t revisions. The verifier V_k for constraint family k checks y_t and reports all remaining violations. If the draft is unsuccessful and the budget has not been exhausted, model m produces y_{t+1} according to its revision policy π_m:

$$V_k(x, y_t) = (J_t, v_t), \qquad y_{t+1} \sim \pi_m(\cdot \mid x, y_t, v_t, h_t), \tag{2}$$

where $J_t = J(x, y_t)$ is the success indicator, v_t is the violation report, and h_t contains retained earlier drafts and feedback. The policy π_m specifies the distribution over revised texts given the task, current draft, feedback, and history, under the model's fixed tokenizer, chat template, and decoding configuration. This configured model is called a "controller" and also a "reviser" when we describe its role in revising a supplied draft. Each model generates its own initial draft y_0 in the main evaluation; fixed-draft comparisons also use drafts generated by other models.

Constraint families. We apply the protocol to three families of verifiable constraints, as Figure 2 shows. The *exact-length constraints* require both the requested word count and a fixed set of content anchors: specified words or phrases that must be retained. The feedback reports the current and target word counts, the required addition or deletion, and any missing anchors. The *lexical constraints* require specified words to appear, occupy specified positions, or end specified sentences, with word- or sentence-count requirements where applicable. The *compositional constraints* combine sentence

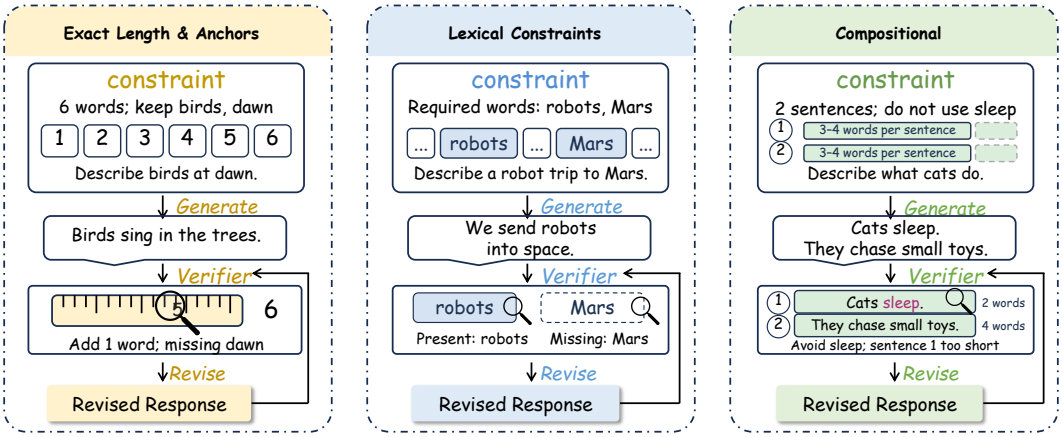

Figure 2: Illustration of exact-length, lexical, compositional constraints and the shared revision loop.

count with forbidden words or per-sentence length bounds. The latter two constraint families together use five structures from COLLIE (Yao et al., 2023); each case includes a human-written response that satisfies all its constraints. We fix the rules for word counting, sentence splitting, and matching, as well as the feedback fields, so that every violation report is reproducible. Detailed constraint definitions are given in Appendix A.1.

Loop outcomes. We measure whether the loop succeeds and how many revisions it takes. The capture time is the index of the first draft that satisfies all constraints:

$$T_{\text{cap}} = \min\{0 \le t \le T : J_t = 1\}, \tag{3}$$

where we allow up to T revisions in total, stopping at the first successful draft or when the budget is exhausted ($T_{\text{cap}} = \infty$). Final joint success (shortened to final success) is $\mathbf{1}[T_{\text{cap}} \le T]$, where $\mathbf{1}[\cdot]$ equals one when its condition holds and zero otherwise.

We fixed $T = 8$ in the main experiments to allow repeated correction while limiting inference cost and dialogue growth. Each model receives the same maximum number of revisions while token costs can differ. Eight rounds do not imply convergence and Appendix A.4 reports an audit that continues selected failures to round 32. To describe how revision progresses, we also track error reduction and output repetition. *Contraction* means a decrease in the violation measure for the task's constraint family. An unsuccessful output *recurs* when it is byte-identical to an earlier draft.

Evaluation design. We evaluate 480 cases per constraint family. Exact-length targets and anchors are derived from human-written references and prompts. Lexical and compositional cases come from COLLIE (Yao et al., 2023). Cases are shared across models within each family, but differ across families. We test 19 model configurations—12 local checkpoints from seven model families and seven API systems—under the eight-revision protocol, yielding 27,360 attempted trajectories. Success rates use protocol-complete records; paired comparisons use cases available for all models being compared. We use source-stratified bootstrap intervals with 20,000 resamples, exact McNemar tests (McNemar, 1947) with Holm correction (Holm, 1979), and case-clustered interaction models for the main evaluations. Data and statistical details are given in Appendix A.

4 Revision Performance and Feedback-to-Action Responses

4.1 Revision Performance across Rounds and Models

Table 1 shows substantial variation in final joint success even under exact feedback. API systems perform better on average than local checkpoints, but large differences remain within both groups, with controller-level mean success ranging from 17.4% to 99.8%. Performance also varies strongly across constraint families: exact-length and lexical tasks remain challenging for many models, whereas compositional tasks generally achieve higher success.

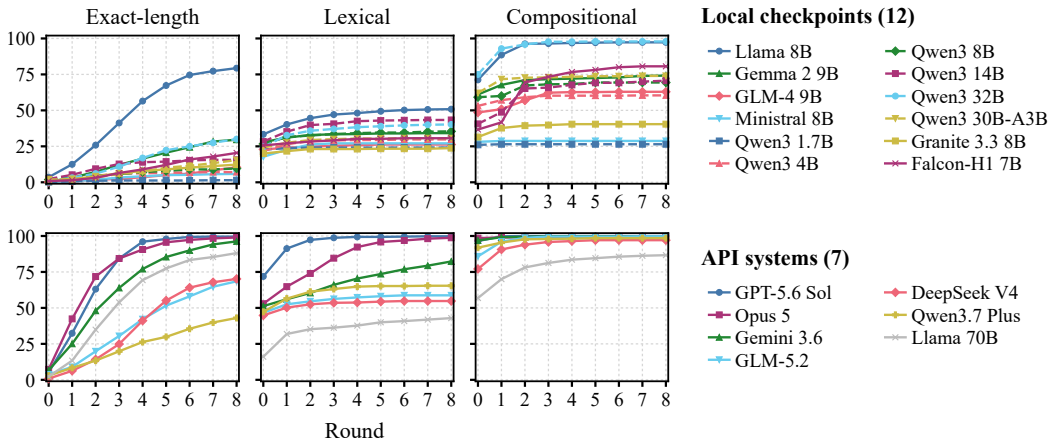

Figure 3: Cumulative joint success across three constraint families. Round 0 is the initial draft. Full model names appear in Table 1.

At the same time, this overall local–API separation is not uniform. Llama 3.1 8B, for example, reaches a mean success rate of 75.8%, matching or outperforming four of the seven API systems despite being a local 8B checkpoint. Overall, **exact feedback does not eliminate substantial model- and task-dependent variation, and closed-loop revision performance does not follow a simple ordering by deployment category or nominal model scale**.

Figure 3 shows cumulative joint success over revision rounds 0–8 and reveals distinct recovery profiles across both tasks and models. **Exact-length tasks benefit most visibly from repeated revision**: several controllers continue to improve well into the later rounds, although the rate of recovery differs substantially across models. Some systems approach saturation after only a few revisions, while others, including Llama 3.1 8B and several API models, continue making meaningful gains near the end of the budget. Lexical tasks show a different pattern: most improvements occur in the early rounds and then diminish, leaving many models at a relatively stable plateau. Compositional tasks are more front-loaded, with many controllers already performing strongly after the first few revisions and showing little additional gain thereafter. These curves also reveal that the same revision budget has very different value across controllers: some models continue recovering from earlier failures, whereas others stop improving much earlier. Overall, **models differ not only in final success, but also in the speed, persistence, and timing of recovery over repeated revisions**. Detailed round-by-round results are reported in Appendix A.4.

Model	Exact	Lex.	Comp.	Mean ± SD
Local checkpoints				
[1] Llama 3.1 8B	79.4	50.8	97.3	75.8 ± 23.5
[2] Qwen3 32B	30.0	40.2	97.9	56.0 ± 36.6
[3] Gemma 2 9B	29.6	34.2	74.2	46.0 ± 24.5
[4] Falcon-H1 7B	20.4	30.6	80.6	43.9 ± 32.2
[5] Qwen3 14B	15.6	43.3	70.6	43.2 ± 27.5
[6] Qwen3 30B-A3B	15.4	30.6	74.0	40.0 ± 30.4
[7] Qwen3 8B	9.6	35.4	69.4	38.1 ± 30.0
[8] GLM-4 9B	10.4	26.3	62.9	33.2 ± 26.9
[9] Qwen3 4B	7.1	29.8	60.4	32.4 ± 26.7
[10] Granite 3.3 8B	12.3	23.8	40.4	25.5 ± 14.1
[11] Ministral 8B	5.8	27.1	28.8	20.6 ± 12.8
[12] Qwen3 1.7B	1.5	24.2	26.5	17.4 ± 13.8
Local mean ($n = 12$)	*19.8*	*33.0*	*65.2*	*39.3 ± 23.4*
API systems				
[13] GPT-5.6 Sol	99.6	99.8	100.0	99.8 ± 0.2
[14] Claude Opus 5	99.0	98.8	99.2	99.0 ± 0.2
[15] Gemini 3.6 Flash	96.2	82.3	99.8	92.8 ± 9.2
[16] GLM-5.2	68.5	58.8	99.8	75.7 ± 21.4
[17] DeepSeek V4 Flash	70.1	54.8	97.1	74.0 ± 21.4
[18] Llama 3.1 70B	88.1	42.9	86.7	72.6 ± 25.7
[19] Qwen3.7 Plus	43.0	65.4	98.5	69.0 ± 27.9
API mean ($n = 7$)	*80.6*	*71.8*	*97.3*	*83.3 ± 12.9*
Overall mean ($n = 19$)	**42.2**	**47.3**	**77.1**	**55.5 ± 18.8**

Table 1: Final joint success (%); the last column reports the mean and sample SD across the three constraint families.

Draft Source	Reviser	Success
Llama 3.1 8B	Llama 3.1 8B	76.2
	Qwen3 14B	17.1
	Gemma 2 9B	33.8
	GLM-4 9B	12.9
Qwen3 14B	Llama 3.1 8B	84.6
	Qwen3 14B	15.8
Gemma 2 9B	Llama 3.1 8B	81.2
	Gemma 2 9B	25.4
GLM-4 9B	Llama 3.1 8B	85.4
	GLM-4 9B	8.3

Table 2: Final joint success (%) from fixed initial drafts on exact-length tasks.

Are later revision gaps simply inherited from differences in initial generation ability? We test this on 240 exact-length cases, where local-model revision performance is most clearly separated. In these fixed-draft comparisons, different revisers

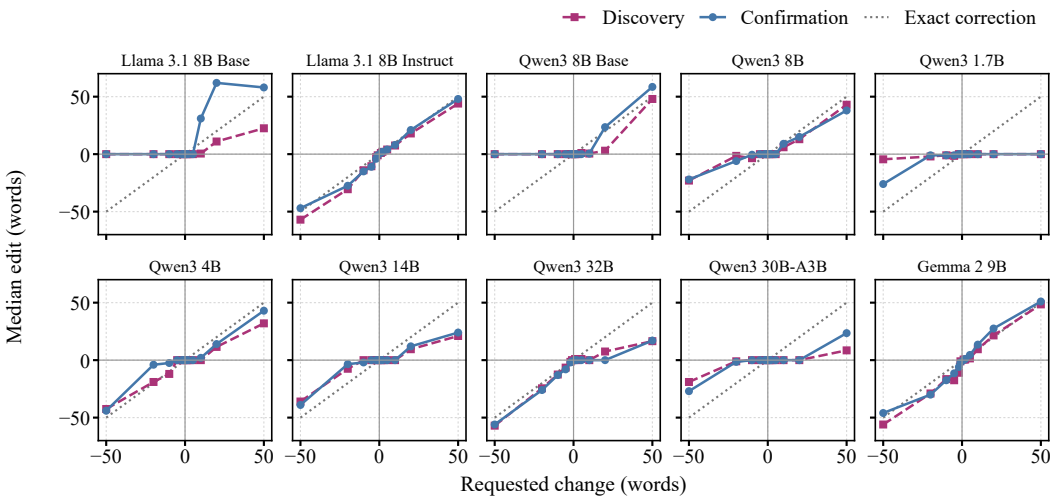

Figure 4: Revision curves from discovery and confirmation texts for all ten checkpoints, spanning post-training conditions and model sizes. Dotted lines mark exact correction, where the median word-count change equals the requested change.

receive the same task, initial draft, target, anchors, and revision budget, while retaining their native interfaces as in the main evaluation. Table 2 shows that, even from identical starting drafts, Llama 3.1 8B retains a 42.5–77.1 percentage-point advantage in final success across the six comparisons, all significant after Holm correction. Notably, Llama succeeds more often when revising each paired model's drafts than when revising its own. The advantage remains 45.3–77.1 points after excluding initially successful drafts (Appendix F.1). Overall, **large cross-model revision gaps persist even when the initial draft is held fixed, showing that they are not simply inherited from differences in initial generation**.

4.2 Revision Responses Across Post-training and Model Scale

Exact-length feedback lets us directly compare the requested word-count change with the actual edit. Because natural revision trajectories entangle correction requests with model-dependent drafts and histories, we hold each source draft fixed and vary only its target word count and corresponding verifier feedback. These controlled one-step revisions characterize feedback-to-action responses, while the original multi-round trajectories are reserved for testing whether the resulting response curves generalize to naturally occurring revision states.

For target N and word count $W(y_t)$, we define the signed error d_t and actual word-count change a_t:

$$d_t = W(y_t) - N, \qquad a_t = W(y_{t+1}) - W(y_t), \qquad d_{t+1} = d_t + a_t. \tag{4}$$

The requested change is $c = -d_t$: positive values request additions and negatives request deletions. An exact correction has $a_t = c$, correct direction means $a_t d_t < 0$, and error expansion means $|d_{t+1}| > |d_t|$. Zero action means no change in word count ($a_t = 0$), though the text may change.

We estimate each model's response curve $g_m(c)$ as the median word-count change at each requested correction. Curves are estimated on 24 *discovery* texts and evaluated without refitting on 48 disjoint *confirmation* texts. Figure 4 shows all ten checkpoints on the same texts and request grid. Discovery and confirmation curves correlate strongly for every checkpoint (Spearman $\rho > 0.85$), indicating that the response patterns replicate across unseen texts.

Figure 4 also shows that post-training and model scale reshape revision responses in different ways. Both Llama 3.1 8B Base and Qwen3 8B Base exhibit near-zero responses over a range of deletion requests, but instruction tuning reshapes these behaviors differently. For Llama, instruction tuning largely removes this deletion-side dead zone and brings both directions closer to exact correction; for Qwen, it reduces error-expanding edits but also increases zero-action responses. Across the dense Qwen3 series from 1.7B to 32B, increasing model size does not bring responses uniformly closer to exact correction; for example, Qwen3 32B can over-correct deletion requests while still

under-correcting additions. Gemma 2 9B provides a contrasting response profile, with median edits tracking the exact-correction line relatively closely in both directions. Overall, the curves reveal distinct response failures, including directional asymmetry, under-correction, and near-zero action. The Qwen3 30B-A3B MoE checkpoint provides a contrasting case, with zero median word-count change over tested requests from -10 to $+20$ words in both discovery and confirmation. Quantitative comparisons are in Appendix C.1.

Taken together, these results reveal systematic differences in how models translate exact feedback into revision actions. **Post-training can substantially reshape a shared Base-model failure pattern, but does so in checkpoint-specific ways, while increasing model scale does not consistently move responses toward exact correction**; models instead exhibit distinct patterns of under-correction, asymmetric editing, and near-zero action. Importantly, these differences extend beyond the controlled setting: across nine checkpoints, model-specific response curves reduce edit-prediction MAE on independent multi-round trajectories by 1.91 words relative to a shared curve, an 11.0% relative reduction (95% CI 8.5–13.9). Thus, **feedback-to-action responses are reproducible and remain informative as revision states evolve, but better local responses alone do not determine successful closed-loop revision**. Detailed comparisons and baselines are in Appendix C.1.

5 Failure Dynamics and Targeted Interventions

5.1 Output Recurrence across Constraints

Failed trajectories frequently repeat complete outputs. Among failed trajectories, recurrence occurs in 84.8% of 5,272 exact-length, 78.7% of 4,805 lexical, and 90.2% of 2,093 compositional trajectories, respectively. In Figure 5, we compare complete outputs by exact byte equality and distinguish four patterns: a **terminal fixed point** ends with three identical drafts; a **terminal cycle** ends with two identical consecutive blocks of $p \in \{2, 3, 4\}$ drafts, each containing at least two distinct outputs; **other repeat** contains repetition without either terminal pattern; and **no repeat** contains only distinct drafts, including the initial draft. Fixed points take priority. These categories describe behavior within the observed revision budget, not guaranteed long-run dynamics.

Repetition often persists, but its form varies across tasks and models. Terminal fixed points or cycles account for 69.0%, 57.6%, and 79.5% of failures in exact-length, lexical, and compositional tasks, respectively. Fixed points alone account for 55.4%, 48.9%, and 71.2%: compositional failures are especially dominated by fixed points, whereas lexical failures contain larger shares of other repetition and no repetition. Controller differences also matter. For example, on exact-length tasks, Qwen3 1.7B's failures are predominantly fixed points, whereas Granite and Falcon-H1 have large nonrepeating shares despite their low final success. **These rates are conditional on failure.** In particular, a large repeated-output share among a strong model's few failures does not imply frequent repetition overall. Recurrence is an important failure pattern, not a complete explanation of model quality. Appendix G reports the counts and formal classification rules.

Prior recurrence is associated with lower subsequent recoverability. For exact-length tasks, we restrict the analysis to trajectories that remain unresolved after revision 4 and have at least one

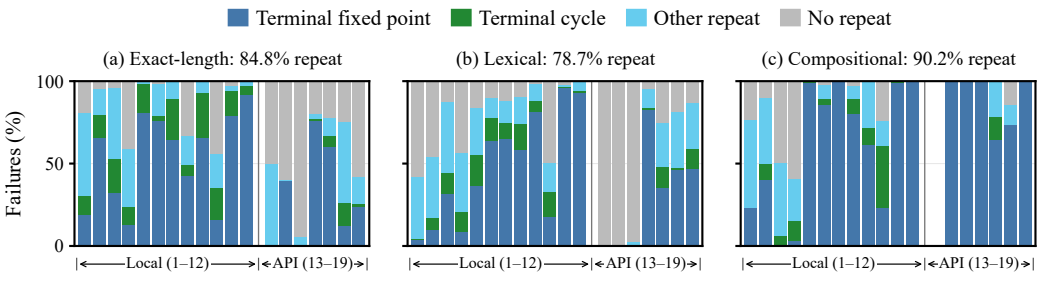

Figure 5: Output recurrence conditional on failure. Each bar is normalized by that model's failure count within the constraint family. Bars follow the model order in Table 1, grouped as local (1–12) and API (13–19); the empty GPT-5.6 Sol bar in (c) indicates no failures.

subsequent revision. We then test whether recurrence observed through revision 4 is associated with successful recovery during revisions 5–8. To distinguish this association from differences in the current revision state, we adjust for current error and missing anchors, prior error reduction and editing behavior, checkpoint, source, and length band (full specification in Appendix C.1).

Table 3 shows a consistent negative association between prior recurrence and later recovery. An odds ratio below 1 indicates that trajectories with more prior repetition have lower odds of succeeding during revisions 5–8 after accounting for the measured revision state and history. In the local panel, increasing the prior repeat fraction from 0 to 1 corresponds to an adjusted odds ratio of 0.059, meaning that the estimated odds of later success fall to about 5.9% of those at the no-repeat end of the scale. The API panel shows the same pattern (OR $= 0.017$), with a combined estimate of 0.047. All three 95% confidence intervals lie well below the no-association value of 1, with small p values, supporting a robust negative association. The same direction is observed across most individual controllers and data sources, with additional landmark and length-subset checks reported in Appendix C.1.

Panel	**Size** n	**Odds Ratio**	95% **CI**	p **Value**
Local	5,039	0.059	0.019–0.186	1.19×10^{-6}
API	1,235	0.017	0.002–0.120	4.70×10^{-5}
All	6,274	0.047	0.017–0.130	3.60×10^{-9}

Table 3: Adjusted odds of success in revisions 5–8 among cases unresolved at revision 4. Odds ratio compares repeat fractions 1 versus 0; n is the risk-set size. Intervals and p values cluster by case.

Recurrence therefore carries information about later recoverability even after adjustment for the measured state and history. This is a predictive association, not evidence that repetition itself causes failure. To test whether retained dialogue contributes to continued recurrence, we intervene on earlier history while holding the current draft and verifier feedback fixed, and measure subsequent revision behavior and final success separately in the next subsection.

5.2 History Interventions at Matched States

To move from association to intervention, we compare continuations from the same recurrent state while varying only the retained dialogue history. The full-history condition preserves earlier draft–feedback pairs, whereas the reset condition removes them. The task, current draft, verifier report, decoding settings, and remaining revision budget are otherwise identical, allowing us to isolate how retained history changes subsequent revision. Our primary exact-length analysis includes 83 Llama 3.1 8B and 120 GLM-4 9B recurrent states selected at revision three or later, with over-length Llama drafts and under-length GLM drafts (Appendix C.2).

Removing earlier history substantially changes continuation behavior, but much less reliably improves final success. As Figure 6 shows, next-step escape increases by 42.2 percentage points for Llama and 49.2 for GLM. Correct-direction editing increases by 32.5 and 25.0 points, error contraction by 18.1 and 24.2 points, and later recurrence falls by 33.7 and 27.5 points. These changes go beyond merely producing a different output: after reset, continuations are more likely to move toward the target, reduce word-count error, and avoid returning to previously visited outputs. Yet the gains in final success are much smaller, increasing by only 10.8 points for Llama and 5.0 for GLM, to 21.7% and 8.3%, respectively. Most reset continuations therefore still fail to satisfy all constraints within the remaining budget. **History removal can strongly reshape local revision behavior without producing proportionate improvements in closed-loop completion.** Figure 7 broadens this result beyond the primary exact-length intervention.

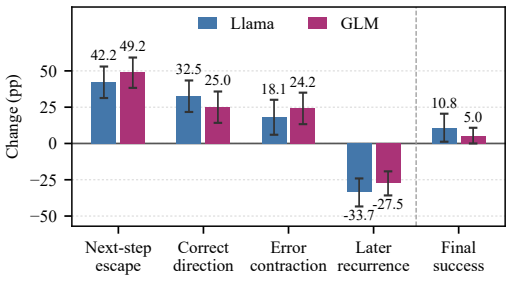

Figure 6: Reset-minus-full effects on selected exact-length states; stratified paired-bootstrap 95% intervals. Lower recurrence is better.

History effects vary across models and tasks, but local escape remains weakly coupled to completion. Across 868 matched recurrent states from Gemma, Qwen, and Falcon on lexical and

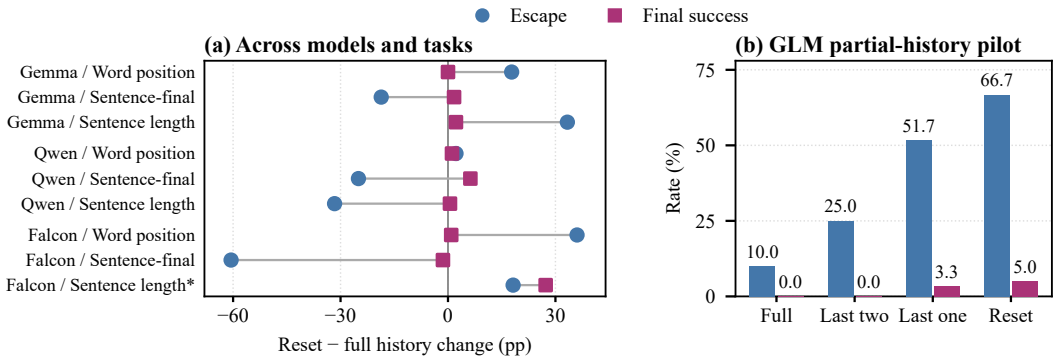

Figure 7: History effects beyond the primary exact-length intervention. (a) Reset-minus-full changes for nine model–task cells (868 states); lines pair outcomes. *Descriptive cell ($n = 11$). (b) Four categorical history conditions on the same 60 GLM states, task-balanced; last two/one retain that many draft–feedback pairs.

compositional tasks, removing earlier dialogue changes escape from -60.6 to $+36.0$ percentage points across the nine model–task cells (Figure 7a). Final-success changes are much smaller in eight of the nine cells, ranging from -1.4 to $+6.2$ points; the remaining Falcon sentence-length cell contains only 11 states and is reported descriptively. A 60-state GLM partial-history pilot shows the same separation: retaining progressively less dialogue raises escape from 10.0% under full history to 66.7% after reset, while final success remains between 0.0% and 5.0% (Figure 7b). These results show that history removal can help or hinder recurrence escape depending on the model and task, while large local changes need not translate into comparable gains in completion. The direction also depends on the trigger-state composition: excluding first-revision repeats reverses some escape effects (Appendix F.3).

Taken together, the matched-state experiments show that retained dialogue is part of the effective revision state: changing history can substantially alter continuation behavior even when the current draft and feedback are fixed. **History therefore shapes closed-loop dynamics, but manipulating history alone is not sufficient to make revision reliable.** Full cross-task results, paired partial-history contrasts, and trigger-state sensitivity are reported in Appendices C.3, C.4, and F.3. Appendix D reports supplementary internal-model analyses.

6 Discussion and Conclusion

Taken together, our results suggest that closed-loop revision should be understood as a property of the interaction between the model and its evolving revision state, rather than of feedback quality alone. Once feedback correctness and completeness are held fixed, substantial differences remain across models and constraints, persist from matched starting drafts, and appear as reproducible feedback-to-action response patterns. These differences do not follow a simple ordering by model scale or deployment category. At the trajectory level, recurrent outputs mark states with lower subsequent recoverability, while matched-history interventions show that changing retained dialogue can strongly alter local continuation behavior without reliably improving final completion. The same intervention can even help or hinder recurrence escape across models and tasks. Together, these findings separate three notions that are often conflated in iterative revision: observing the error, translating feedback into an effective action, and reliably completing the loop.

Evaluation should therefore consider not only endpoint success or single-step improvement, but also consistent feedback incorporation, recurrence, and retained history. More broadly, improvements at one stage need not propagate to end-to-end completion: changing history can substantially alter local revision behavior with much smaller gains in final success. Our fixed-budget, deterministically checkable text tasks do not cover subjective objectives, incomplete feedback, or changing environments. Within this scope, **exact feedback provides observability, not control: reliable closed-loop revision requires models whose responses remain effective as the revision state evolves.**

Reproducibility statement

Section 3 and Appendix A specify verifiers, data, controllers, decoding, interventions, and provenance limits. Anonymous supplementary materials contain frozen inputs, configurations, integrity manifests, the claim-to-artifact ledger, and generation/analysis entry points. Reruns require reviewer-supplied checkpoints or API credentials; neither is included.

AI use statement

Generative AI tools were used to assist with methodological and experimental-design feedback, implementation, data processing and analysis. They were also used for literature search and summarization, figure and artifact preparation, and manuscript drafting and editing. Generative AI was not used to generate the evaluation datasets or to formulate or prove mathematical claims. All AI-assisted research outputs were reviewed and verified by the authors, who conducted and audited the reported experiments and take full responsibility for the final content of this work.

References

Alibaba Cloud. Qwen3.7-Plus model information. Alibaba Cloud Model Studio documentation, 2026. URL https://help.aliyun.com/en/model-studio/qwen3-7-plus. Accessed 2026-08-26.

Anthropic. Introducing Claude Opus 5. Anthropic product announcement, July 2026. URL https://www.anthropic.com/news/claude-opus-5. Published July 24, 2026; accessed 2026-09-14.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. URL https://journals.ametsoc.org/view/journals/mwre/78/1/1520-0493_1950_078_0001_vofeit_2_0_co_2.xml.

A. Colin Cameron and Douglas L. Miller. A practitioner's guide to cluster-robust inference. *Journal of Human Resources*, 50(2):317–372, 2015. doi: 10.3368/jhr.50.2.317. URL https://doi.org/10.3368/jhr.50.2.317.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free Dolly: Introducing the world's first truly open instruction-tuned LLM, 2023. URL https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm.

DeepSeek. DeepSeek-V4-Flash. DeepSeek API documentation, 2026. URL https://api-docs.deepseek.com/updates/. Accessed 2026-08-26.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall/CRC, 1993. doi: 10.1201/9780429246593. URL https://doi.org/10.1201/9780429246593.

Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 889–898, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1082. URL https://aclanthology.org/P18-1082/.

Thomas Palmeira Ferraz, Kartik Mehta, Yu-Hsiang Lin, Haw-Shiuan Chang, Shereen Oraby, Sijia Liu, Vivek Subramanian, Tagyoung Chung, Mohit Bansal, and Nanyun Peng. LLM self-correction with DeCRIM: Decompose, critique, and refine for enhanced following of instructions with multiple constraints. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7773–7812. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.458. URL https://aclanthology.org/2024.findings-emnlp.458/.

Gemma Team, Morgane Riviere, et al. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

GLM Team, Aohan Zeng, et al. ChatGLM: A family of large language models from GLM-130B to GLM-4 all tools, 2024. URL https://arxiv.org/abs/2406.12793.

Google. Gemini 3.6 Flash. Gemini API model documentation, 2026. URL https://ai.google.dev/gemini-api/docs/models/gemini-3.6-flash. Accessed 2026-08-26.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. CRITIC: Large language models can self-correct with tool-interactive critiquing. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 57734–57811, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/fef126561bbf9d4467dbb8d27334b8fe-Paper-Conference.pdf.

Aaron Grattafiori et al. The Llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

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

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 32808–32824, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/8b4add8b0aa8749d80a34ca5d941c355-Paper-Conference.pdf.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Tomas Jackson, Noah Brown, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In *Proceedings of the 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 1769–1782. PMLR, 2023. URL https://proceedings.mlr.press/v205/huang23c.html.

IBM Granite Team. Granite-3.3-8B-Instruct model card. Hugging Face model card, 2025. URL https://huggingface.co/ibm-granite/granite-3.3-8b-instruct.

Dongwei Jiang, Bowei Zhang, Andrew Wang, Nicholas Andrews, and Daniel Khashabi. Feedback friction: LLMs struggle to fully incorporate external feedback. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 54746–54778. Curran Associates, Inc., 2025. doi: 10.52202/085713-1831. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/4eb5cc598a6271528ed9b84bb9879e9b-Paper-Conference.pdf.

Ryo Kamoi, Yusen Zhang, Nan Zhang, Jiawei Han, and Rui Zhang. When can LLMs actually correct their own mistakes? a critical survey of self-correction of LLMs. *Transactions of the Association for Computational Linguistics*, 12:1417–1440, 2024. doi: 10.1162/tacl_a_00713. URL https://aclanthology.org/2024.tacl-1.78/.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. OpenAssistant conversations—democratizing large language model alignment. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 47669–47681. Curran Associates, Inc., 2023. doi: 10.52202/075280-2064. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/949f0f8f32267d297c2d4e3ee10a2e7e-Paper-Datasets_and_Benchmarks.pdf.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, JD Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal Behbahani, and Aleksandra Faust. Training language models to self-correct via reinforcement learning. In Y. Yue, A. Garg, N. Peng, F. Sha,

and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 54523–54549, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/871ac99fdc5282d0301934d23945ebaa-Paper-Conference.pdf.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. LLMs get lost in multi-turn conversation. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 54738–54778, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/59f6421e64707225fdf5b28840679a07-Paper-Conference.pdf.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2024. URL https://arxiv.org/abs/2411.15124.

Young-Jun Lee, Seungone Kim, Byung-Kwan Lee, Minkyeong Moon, Yechan Hwang, Jong Myoung Kim, Graham Neubig, Sean Welleck, and Ho-Jin Choi. RefineBench: Evaluating refinement capability of language models via checklists. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 81042–81085, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/835b6c04ca2daaa1a682c2eaad4a70ea-Paper-Conference.pdf.

Jiaming Li, Lei Zhang, Yunshui Li, Ziqiang Liu, Yuelin Bai, Run Luo, Longze Chen, and Min Yang. Ruler: A model-agnostic method to control generated length for large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3042–3059, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.172. URL https://aclanthology.org/2024.findings-emnlp.172/.

Aofan Liu and Jingxiang Meng. Self-correction as feedback control: Error dynamics, stability thresholds, and prompt interventions in LLMs, 2026. URL https://arxiv.org/abs/2604.22273.

Lennart Ljung. *System Identification: Theory for the User*. Prentice Hall, Upper Saddle River, NJ, 2nd edition, 1999.

Ximing Lu, Peter West, Rowan Zellers, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. NeuroLogic decoding: (un)supervised neural text generation with predicate logic constraints. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4288–4299, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.339. URL https://aclanthology.org/2021.naacl-main.339/.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative refinement with self-feedback. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 46534–46594. Curran Associates, Inc., 2023. doi: 10.52202/075280-2019. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/91edff07232fb1b55a505a9e9f6c0ff3-Paper-Conference.pdf.

Aleksandar Makelov, Georg Lange, Atticus Geiger, and Neel Nanda. Is this the subspace you are looking for? an interpretability illusion for subspace activation patching. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 26486–26515, 2024. URL https://proceedings.ic

lr.cc/paper_files/paper/2024/file/70b8505ac79e3e131756f793cd80eb8d-Paper-Conference.pdf.

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. URL https://doi.org/10.1007/BF02295996.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372. Curran Associates, Inc., 2022. doi: 10.52202/068431-1262. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/6f1d43d5a82a37e89b0665b33bf3a182-Paper-Conference.pdf.

Mistral AI Team. Model card for Ministral-8B-Instruct-2410. Hugging Face model card, 2024. URL https://huggingface.co/mistralai/Ministral-8B-Instruct-2410.

Todd Nief, David Reber, Sean Richardson, and Ari Holtzman. Dynamic weight grafting: Localizing finetuned factual knowledge in transformers, 2025. URL https://arxiv.org/abs/2506.20746.

Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli, and Matej Balog. AlphaEvolve: A coding agent for scientific and algorithmic discovery. Google DeepMind white paper, 2025. URL https://storage.googleapis.com/deepmind-media/DeepMind.com/Blog/alphaevolve-a-gemini-powered-coding-agent-for-designing-advanced-algorithms/AlphaEvolve.pdf. May 2025.

Theo X. Olausson, Jeevana Priya Inala, Chenglong Wang, Jianfeng Gao, and Armando Solar-Lezama. Is self-repair a silver bullet for code generation? In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 36545–36593, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/9ddc141bdbf9d1db510cefff56c586ad-Paper-Conference.pdf.

OpenAI. GPT-5.6: Frontier intelligence that scales with your ambition. OpenAI product release, 2026. URL https://openai.com/index/gpt-5-6/. Accessed 2026-08-26.

Nazneen Rajani, Lewis Tunstall, Edward Beeching, Nathan Lambert, Alexander M. Rush, and Thomas Wolf. No robots. https://huggingface.co/datasets/HuggingFaceH4/no_robots, 2023.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652. Curran Associates, Inc., 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1b44b878bb782e6954cd888628510e90-Paper-Conference.pdf.

Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. On the self-verification limitations of large language models on reasoning and planning tasks. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4O0v4s3IzY.

Sankaran Vaidyanathan, David Arbour, Aaron Mueller, Scott Niekum, and David Jensen. The curse of multiple mediators: Hidden interaction effects in activation patching, 2026. URL https://arxiv.org/abs/2606.27510.

Yuh-Shyang Wang, Lily Weng, and Luca Daniel. Neural network control policy verification with persistent adversarial perturbation. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10050–10059. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/wang20v.html.

Xuening Wu, Qianya Xu, Yanlan Kang, Zeping Chen, Yubin Liu, and Shenqin Yin. Do language models converge to themselves? recursive self-refinement as textual relaxation, 2026. URL https://arxiv.org/abs/2607.22653.

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, 2025. URL https://arxiv.org/abs/2505.09388.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-agent: Agent–computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-1601. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/5a7c947568c1b1328ccc5230172e1e7c-Abstract-Conference.html.

Shunyu Yao, Howard Chen, Austin W. Hanjie, Runzhe Yang, and Karthik Narasimhan. COLLIE: Systematic construction of constrained text generation tasks, 2023. URL https://arxiv.org/abs/2307.08689.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Hf17y6u9BC.

Zhipu AI. GLM-5.2. Zhipu AI model documentation, 2026. URL https://docs.bigmodel.cn/cn/guide/models/text/glm-5.2. Accessed 2026-08-26.

Jingwei Zuo, Maksim Velikanov, Ilyas Chahed, Younes Belkada, Dhia Eddine Rhayem, Guillaume Kunsch, Hakim Hacid, Hamza Yous, Brahim Farhat, Ibrahim Khadraoui, Mugariya Farooq, Giulia Campesan, Ruxandra Cojocaru, Yasser Djilali, Shi Hu, Iheb Chaabane, Puneesh Khanna, Mohamed El Amine Seddik, Ngoc Dung Huynh, Phuc Le Khac, Leen AlQadi, Billel Mokeddem, Mohamed Chami, Abdalgader Abubaker, Mikhail Lubinets, Kacper Piskorski, and Slim Frikha. Falcon-H1: A family of hybrid-head language models redefining efficiency and performance, 2025. URL https://arxiv.org/abs/2507.22448.

A Constraint Families, Data, and Execution

Reading guide. Appendix A specifies the tasks, data, and budget replay; Appendix B gives endpoint inference and robustness checks. Appendix C supports the controlled response, recoverability, and history analyses in Sections 4.2–5.2. Appendices D and E report supplementary checkpoint interventions and interface comparisons; Appendix F covers crossover, capture dynamics, and trigger-state sensitivity, and Appendix G gives the common failure taxonomy and counts.

A.1 Verifiers and Violation Measures by Constraint Family

For every constraint family, the fixed verifier returns a joint-success indicator J and a nonnegative violation vector $\mathbf{v}_k$. When an analysis requires a scalar violation measure, we use

$$E_k(\mathbf{v}) = \sum_j v_j, \qquad E_k(\mathbf{v}) = 0 \iff J = 1. \tag{5}$$

This sum summarizes violations within a task; its components retain their distinct meanings and units.

Exact length. The frozen Unicode-aware word count $W(y)$ uses the regular expression

```
\b[\w]+(?:[-'][\w]+)*\b.
```

Each case has target N and anchor set A, extracted before generation from the human-authored prompt–reference pair. The matcher $M(q, y)$ is case-insensitive, token-boundary-aware, and permits flexible whitespace. The reported endpoint is

$$J_{\min}(y) = \mathbf{1}[W(y) = N \ \wedge \ \forall q \in A : M(q, y) = 1]. \tag{6}$$

For this constraint family, $J_t = J_{\min}(y_t)$.

Its violation vector contains normalized absolute length error followed by one missingness indicator per anchor,

$$\mathbf{v}_{\text{len}}(y) = \left(\frac{|W(y) - N|}{\max(N, 1)}, \{1 - M(q, y) : q \in A\} \right). \tag{7}$$

Any stricter binary content requirement Q can only reduce the accepted set: $J_{\min}(y)Q(y) \leq J_{\min}(y) \leq \mathbf{1}[W(y) = N]$. Anchors establish minimum retention; semantic equivalence and subjective quality remain unmeasured. Energy analyses across constraint families use Equation 7; signed system-identification and exact-length reset analyses explicitly use $|d_t|$ when reporting absolute length-error contraction.

Lexical and compositional structures. Let n_w, n_s be the observed word and sentence counts and N_w, N_s their requested values. The count residuals are $r_w = n_w - N_w$ and $r_s = n_s - N_s$. Let δ_j indicate a missing or mismatched lexical condition, f_j count occurrences of forbidden word j, and n_i be the word count of observed sentence i. The five COLLIE-derived structures use

$$\mathbf{v}_{\text{c05}} = \left(|r_w| / \max(N_w, 1), \delta_1^{\text{pos}}, \ldots, \delta_q^{\text{pos}} \right), \tag{8}$$

$$\mathbf{v}_{\text{c07}} = \left(\delta_1^{\text{req}}, \ldots, \delta_q^{\text{req}} \right), \tag{9}$$

$$\mathbf{v}_{\text{c09}} = \left(|r_s| / \max(N_s, 1), f_1, \ldots, f_q \right), \tag{10}$$

$$\mathbf{v}_{\text{c10}} = \left(|r_s| / \max(N_s, 1), \{(L - n_i)_+ / \max(L, 1)\}_{i=1}^{n_s}, \{(n_i - U)_+ / \max(U, 1)\}_{i=1}^{n_s} \right), \tag{11}$$

$$\mathbf{v}_{\text{c12}} = \left(|r_s| / \max(N_s, 1), \delta_1^{\text{final}}, \ldots, \delta_q^{\text{final}} \right). \tag{12}$$

Here q is the number of lexical conditions in the task, L and U are the requested per-sentence bounds, and $(z)_+ = \max(z, 0)$. In words, c05 imposes word-position requirements under a word-count target, c07 required-word presence, c09 forbidden words under a sentence-count target, c10 per-sentence length bounds under a sentence-count target, and c12 sentence-final words under a sentence-count target. The lexical constraint family allocates 240/120/120 of its 480 cases to c05/c07/c12 and the compositional evaluation set 240/240 to c09/c10. Thus the lexical column of Table 1 is an unequally weighted composite; the compositional column equally weights its two structures. Because c05 and c12 also carry count residuals, the three family labels are reporting conventions rather than a partition by violation type. Frozen word tokenization and sentence splitting are shared by generation and audit; each case includes a human-authored witness that passes the same verifier.

Trajectory events. An unsuccessful recurrence occurs when the SHA-256 hash of a draft equals that of an earlier draft while $J = 0$. An unresolved trajectory is a persistent fixed point when its final three drafts are identical. It is a terminal period-p cycle, $p \in \{2, 3, 4\}$, when its final $2p$ drafts comprise two identical p-draft blocks and the block contains at least two distinct drafts. For a selected recurrent state, first-step escape means that the next draft differs from every draft seen before the intervention. Later recurrence checks for repetition of any previously seen draft after the first continuation step; final joint success is measured separately.

Exact-length failure categories. The additional exact-length classification covers protocol-complete trajectories without final joint success: 4,622 local and 650 API failures. Protocol-incomplete records are excluded. For a trajectory with drafts $y_0, \ldots, y_K$, let K be its last recorded revision index and $d_t = W(y_t) - N$ its signed word-count error. All checks include the initial draft. The classifier assigns the *first matching category* in the following order; the legend order is only for display.

1. **Anchor failure.** The final draft has exactly the target word count ($d_K = 0$) but is missing at least one required anchor. This condition is checked at the endpoint and takes priority over all trajectory patterns.
2. **Recurrence.** Two recorded drafts are byte-identical: $y_i = y_j$ for some $0 \le i < j \le K$, checked by SHA-256 of their UTF-8 text without normalization. The repeat need not be consecutive or terminal. This category is therefore broader than a persistent fixed point or terminal cycle, as defined above.
3. **Sign flips (≥ 2).** After removing zero errors from $(d_0, \ldots, d_K)$, the remaining sign sequence changes at least twice. For example, errors $(3, 0, -2, 1)$ give two flips. These are crossings of the word-count target, not necessarily repeated outputs or growing errors.
4. **Repeated residual.** At least one consecutive pair has $d_{t+1} = d_t$. The word count is unchanged at that step, although the text may change. Byte-identical outputs receive the earlier recurrence label.
5. **Near target (≤ 2).** The best absolute error anywhere in the trajectory satisfies $\min_{0 \le t \le K} |d_t| \le 2$. This includes zero error and does not require the final draft to remain near the target. Exact word count alone does not establish joint success.
6. **Net contraction.** The final absolute error is smaller than the initial one: $|d_K| < |d_0|$. Intermediate errors may increase, so this label does not imply monotonic improvement or that a larger budget would suffice for success.
7. **Other non-contraction.** None of the preceding conditions holds. In particular, $|d_K| \ge |d_0|$; equality here compares the endpoints and need not imply equal errors at consecutive revisions.

The assigned labels partition failures, but their underlying conditions can overlap. For example, a trajectory may both repeat a draft and finish closer to the target; it is labeled recurrence. Category shares thus describe this fixed classification rule, not the prevalence of independent causal mechanisms. The common output-repetition classification used in Figure 5 is defined separately in Appendix G.

A.2 Data Construction and Frozen Execution

Combined-480 contains 120 cases each from Dolly (Conover et al., 2023), No Robots (Rajani et al., 2023), WritingPrompts (Fan et al., 2018), and OASST1 (Köpf et al., 2023). Source partitions and case IDs are fixed across checkpoints. The target is the deterministic count of the human response; literal anchors are frozen before generation. Targets span 10–500 words. The 480 cases contain 1,907 anchors, with 3–5 per case (mean 3.973; median 4); 1,804/1,907 (94.6%) are single-token anchors under the frozen word regex.

The lexical and compositional panels each contain 480 frozen cases constructed from official COLLIE instances (Yao et al., 2023). The lexical panel covers required-word, lexical-position, and sentence-final constraints; the compositional panel combines sentence-count requirements with forbidden words or per-sentence length bounds. Each panel contains 160 cases from CC-News, 160 from

Gutenberg, and 160 from Wikipedia. COLLIE supplies the prompts, constraint targets, and human-authored feasible witnesses; we select, balance, audit, and freeze the cases before evaluation. Case IDs are shared across controllers within a constraint family and are distinct across families.

The task, target fields, prompt format, revision budget, and stopping rule are fixed before observing outcomes. Round zero supplies the task and response-only instruction. After each failure, the model receives its complete previous response and the full deterministic verifier report. The loop stops at joint success or after eight revisions.

A.3 Controller Roster and Main-Panel Inference

The 12 locally executed checkpoints are Llama 3.1 8B (Grattafiori et al., 2024), Gemma 2 9B (Gemma Team et al., 2024), GLM-4 9B (GLM Team et al., 2024), Ministral 8B (Mistral AI Team, 2024), Qwen3 1.7B, 4B, 8B, 14B, 32B, and 30B-A3B (Yang et al., 2025), Granite 3.3 8B (IBM Granite Team, 2025), and Falcon-H1 7B (Zuo et al., 2025). They span seven model families and include dense, MoE, and Transformer–SSM hybrid architectures. The seven API controllers are GPT-5.6 Sol (OpenAI, 2026), Claude Opus 5 (Anthropic, 2026), Gemini 3.6 Flash (Google, 2026), GLM-5.2 (Zhipu AI, 2026), DeepSeek V4 Flash (DeepSeek, 2026), Qwen3.7 Plus (Alibaba Cloud, 2026), and Llama 3.1 70B (Grattafiori et al., 2024).

For Qwen3.7 Plus, the saved run configurations across all three constraint families specify the OpenRouter request identifier `qwen/qwen3.7-plus`. The accompanying catalog snapshots record `qwen/qwen3.7-plus-20260602` as the canonical slug. This catalog identifier does not establish the backend weight version for each request; we do not infer a historical snapshot from the provider's current alias mapping.

Evaluating every controller on every constraint family yields 27,360 attempted trajectories. One Gemini refusal leaves 3,359 protocol-complete API exact-length records and 479 cases common to all seven API controllers for paired inference. Other combinations of controllers and constraint families have 480 protocol-complete cases. Uncertainty intervals use 20,000 source-stratified paired-case bootstrap draws (Efron & Tibshirani, 1993); paired binary contrasts use exact McNemar tests (McNemar, 1947) with Holm correction (Holm, 1979); and interaction models cluster by case (Cameron & Miller, 2015).

A.4 Revision-Budget Sensitivity

The primary endpoint compares models after at most eight revisions, using a budget fixed before the main experiments. We replay the completed trajectories offline under every observed revision cap $T \in \{0, 1, \ldots, 8\}$. The API exact-length replay uses the same 479 all-controller-common cases as the paired analysis; all other combinations of controllers and constraint families use 480 cases.

Revision Cap T	Median Cell Success	Cell Range	Local Reversals	API Reversals
0	26.0%	0.2–98.5%	28/66	4/21
1	31.9%	0.6–99.8%	21/66	7/21
2	35.8%	1.0–100.0%	18/66	8/21
3	40.6%	1.2–100.0%	23/66	7/21
4	42.5%	1.2–100.0%	20/66	7/21
5	51.6%	1.2–100.0%	19/66	8/21
6	54.8%	1.2–100.0%	18/66	8/21
7	54.8%	1.5–100.0%	17/66	8/21
8	54.8%	1.5–100.0%	17/66	8/21

Table 4: Frozen revision-budget replay across all 57 combinations of controllers and constraint families. A reversing pair has opposite strict controller orderings on at least two constraint families at the same cap; ties do not count as strict reversals.

Absolute success changes with the action budget. Both panels retain multiple cross-family ordering reversals at every inspected cap. Their existence is therefore robust to the budget cap, although individual rankings change.

Extended-budget audit. The revision-32 audit selects 144 stratified revision-8 failures. Twelve succeed by revision 32, 113 remain unsuccessful through revision 32, and 19 exhaust the native context window. Equivalently, 12 of the 125 context-feasible cases succeed. Llama succeeds on 10/24 additional cases, Gemma and Qwen3 14B on 1/24 each, and GLM, Ministral, and Qwen3 8B on none; 18/24 Gemma trajectories and 1/24 Qwen3 8B trajectories terminate at the context limit. This selected-failure cohort does not estimate population-wide gains from a longer budget.

Tables 5–7 report cumulative success for every model and constraint family at each revision cap, showing where success rates plateau and where they continue to rise late in the budget.

Controller	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Local controllers									
Llama 3.1 8B	3.5	12.5	25.8	41.3	56.5	67.3	74.6	77.3	79.4
Qwen3 32B	0.8	3.1	6.0	10.8	16.9	22.3	25.0	27.5	30.0
Gemma 2 9B	0.4	3.1	7.3	12.1	16.5	20.6	24.4	28.3	29.6
Falcon-H1 7B	0.8	1.5	3.1	6.5	9.0	12.1	15.8	17.9	20.4
Qwen3 14B	2.3	5.2	9.4	12.7	14.0	14.4	15.2	15.6	15.6
Qwen3 30B-A3B	1.9	2.3	3.3	5.8	7.3	10.2	11.5	13.3	15.4
Qwen3 8B	1.9	3.1	4.4	5.6	7.1	7.9	9.0	9.0	9.6
GLM-4 9B	0.2	0.8	1.9	2.3	3.5	5.0	6.0	8.3	10.4
Qwen3 4B	1.3	3.5	4.6	6.0	6.3	6.7	6.9	7.1	7.1
Granite 3.3 8B	1.0	2.1	4.2	6.3	7.9	9.0	10.2	11.3	12.3
Ministral 8B	0.2	0.8	1.7	3.3	4.2	5.0	5.2	5.6	5.8
Qwen3 1.7B	0.2	0.6	1.0	1.3	1.3	1.3	1.3	1.5	1.5
API controllers									
GPT-5.6 Sol	5.2	32.4	63.0	84.3	96.0	97.9	99.4	99.6	99.6
Claude Opus 5	7.3	42.4	71.8	84.3	90.6	95.6	97.3	98.3	99.0
Gemini 3.6 Flash	6.7	25.1	48.0	63.9	77.0	85.4	90.0	94.2	96.2
GLM-5.2	3.8	9.0	19.8	30.5	42.2	51.6	58.0	64.7	68.5
DeepSeek V4 Flash	0.8	6.3	14.2	24.8	41.1	55.1	64.1	67.8	70.1
Llama 3.1 70B	2.3	13.4	34.9	53.9	69.3	77.5	83.3	85.4	88.1
Qwen3.7 Plus	2.9	8.4	13.4	19.8	26.3	29.9	35.5	39.9	43.0

Table 5: Exact-length cumulative joint success (%) after each revision cap. API exact-length cells use the 479-case all-controller-common set; all other cells use 480 cases.

Controller	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Local controllers									
Llama 3.1 8B	33.3	40.2	44.6	47.1	48.1	49.4	50.2	50.6	50.8
Qwen3 32B	24.6	32.7	35.8	37.1	38.3	39.0	39.6	40.0	40.2
Gemma 2 9B	27.9	31.0	32.9	33.3	33.5	34.0	34.0	34.2	34.2
Falcon-H1 7B	25.6	27.1	28.3	29.2	30.2	30.2	30.6	30.6	30.6
Qwen3 14B	28.3	35.2	39.8	40.6	42.5	42.9	43.1	43.3	43.3
Qwen3 30B-A3B	24.2	27.5	29.4	30.0	30.4	30.4	30.6	30.6	30.6
Qwen3 8B	26.7	31.5	32.5	33.5	34.2	34.6	34.6	35.2	35.4
GLM-4 9B	22.9	24.0	24.8	25.4	25.8	26.3	26.3	26.3	26.3
Qwen3 4B	21.7	26.5	28.8	29.2	29.4	29.4	29.8	29.8	29.8
Granite 3.3 8B	20.0	21.7	23.1	23.1	23.1	23.3	23.3	23.3	23.8
Ministral 8B	17.7	22.7	26.3	27.1	27.1	27.1	27.1	27.1	27.1
Qwen3 1.7B	22.7	23.8	24.2	24.2	24.2	24.2	24.2	24.2	24.2
API controllers									
GPT-5.6 Sol	71.9	91.3	97.3	98.8	99.4	99.4	99.6	99.8	99.8
Claude Opus 5	52.9	64.8	74.0	84.6	92.3	95.8	96.9	98.1	98.8
Gemini 3.6 Flash	51.3	56.0	60.4	66.0	70.4	73.5	76.9	79.4	82.3
GLM-5.2	46.7	52.5	54.6	56.3	57.3	58.1	58.8	58.8	58.8
DeepSeek V4 Flash	44.8	50.2	52.5	53.5	53.8	54.4	54.8	54.8	54.8
Llama 3.1 70B	16.0	31.9	35.2	36.3	37.7	40.0	40.8	41.9	42.9
Qwen3.7 Plus	47.3	56.5	61.3	63.1	64.8	65.2	65.2	65.4	65.4

Table 6: Lexical cumulative joint success (%) after each revision cap. Every cell uses 480 cases.

Controller	0	1	2	3	4	5	6	7	8
Local controllers									
Llama 3.1 8B	71.3	88.5	96.3	96.5	96.9	97.1	97.3	97.3	97.3
Qwen3 32B	75.0	92.9	95.8	97.7	97.7	97.9	97.9	97.9	97.9
Gemma 2 9B	61.0	67.7	71.0	71.7	72.1	72.5	72.9	74.0	74.2
Falcon-H1 7B	36.7	41.7	69.4	73.3	76.7	78.1	80.0	80.6	80.6
Qwen3 14B	40.6	48.8	65.2	65.8	67.7	69.2	69.2	70.0	70.6
Qwen3 30B-A3B	62.1	71.7	72.7	72.9	73.3	73.8	73.8	74.0	74.0
Qwen3 8B	59.2	60.0	67.3	68.1	68.3	69.4	69.4	69.4	69.4
GLM-4 9B	48.5	50.8	56.9	62.3	62.7	62.7	62.9	62.9	62.9
Qwen3 4B	53.1	57.1	59.2	60.2	60.2	60.2	60.2	60.4	60.4
Granite 3.3 8B	31.5	37.7	39.4	39.8	40.4	40.4	40.4	40.4	40.4
Ministral 8B	28.1	28.8	28.8	28.8	28.8	28.8	28.8	28.8	28.8
Qwen3 1.7B	26.0	26.5	26.5	26.5	26.5	26.5	26.5	26.5	26.5
API controllers									
GPT-5.6 Sol	96.5	99.8	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Claude Opus 5	98.5	99.0	99.2	99.2	99.2	99.2	99.2	99.2	99.2
Gemini 3.6 Flash	96.9	99.2	99.4	99.8	99.8	99.8	99.8	99.8	99.8
GLM-5.2	85.8	96.0	98.8	99.6	99.6	99.6	99.6	99.8	99.8
DeepSeek V4 Flash	77.1	90.6	94.0	95.8	96.5	97.1	97.1	97.1	97.1
Llama 3.1 70B	56.9	70.0	78.1	81.3	83.5	84.6	85.6	86.3	86.7
Qwen3.7 Plus	91.9	95.4	97.7	97.9	98.1	98.5	98.5	98.5	98.5

Table 7: Compositional cumulative joint success (%) after each revision cap. Every cell uses 480 cases.

Budget profiles. Exact length remains budget-sensitive late: from $T = 4$ to $T = 8$, DeepSeek V4 Flash, GLM-5.2, and Llama 3.1 8B gain 29.0, 26.3, and 22.9 percentage points, respectively. Lexical cells generally flatten earlier, although Gemini 3.6 Flash gains 11.9 points after $T = 4$. Compositional cells are most front-loaded: the largest $T = 4$-to-$T = 8$ gain is 4.0 points (Falcon-H1 7B). The shared endpoint captures heterogeneous late-recovery profiles and leaves convergence unresolved.

A.5 Evidence Panels

Module	Scope	Evaluation Target
Main closed loop	19 controllers × 3 constraint families × 480 cases	Endpoint spread and failure dynamics
System identification	10 checkpoints, 12 commands, 24 discovery + 48 confirmation cases	Feedback-to-action response curves
Held-out action prediction	9 checkpoints; 3,014 eligible checkpoint–case trajectories	Model-specific versus shared prediction MAE
Conditional rescue	19 controllers, revision-4 risk set	Recoverability given recurrence and current state
History reset	83 Llama + 120 GLM recurrent states	Recurrence escape versus final success
Cross-task history	868 states across 9 model–task cells	Heterogeneity in escape and success effects
Partial history	60 GLM states, 4 categorical conditions	Escape and success under partial retention
Aligned checkpoint panel	Llama, Qwen, and Gemma Base/Instruct pairs	Error readout, restoration, activation transplantation, and grafts
Long horizon	144 stratified revision-8 failures (125 context-feasible)	Persistence beyond the primary budget

Table 8: Scope and evaluation target of each evidence panel.

B Inference and Robustness

B.1 Interaction between Controller and Constraint Family

Table 1 reports final joint success using 480 cases per cell, except API exact length, which uses 479 cases common to all seven systems. Its final column gives the mean and sample standard deviation across the three constraint-family rates. Group rows average models within each family before computing these summaries. Since the families use distinct datasets, ranking reversals describe performance across these settings rather than a causal effect of constraint structure.

The frozen seven-controller API panel uses 479 all-model-common evaluable exact-length cases after one stable Gemini refusal; lexical and compositional cells use all 480 cases. Llama 3.1 70B minus Qwen3.7 Plus changes from $+45.1$ points on exact length to -22.5 on lexical constraints, a difference-in-differences of -67.6 points (95% CI $[-74.3, -60.9]$; Holm-adjusted $p = 4.08 \times 10^{-85}$). After Holm correction, 19 of 21 exact-length pairs and 20 of 21 lexical pairs differ in final success. Compositional constraints place six controllers near the ceiling, leaving 9 of 21 pairwise separations.

The Wald test for the interaction between controller and constraint family gives $\chi^2(12) = 1214.05$ ($p = 1.63 \times 10^{-252}$), and the controller-by-structure test gives $\chi^2(16) = 3662.21$ ($p < 10^{-300}$). The largest p-values across all leave-one-controller-out and leave-one-structure-out refits are 2.49×10^{-160} and 7.84×10^{-11}, respectively. These interaction estimates compare cells across distinct datasets for the constraint families and COLLIE case sets.

Controller	Initial Joint	Early Capture	Early Recurrence	Final Joint
Local controllers				
Llama 3.1 8B	3.5	16.7	2.4	79.4
Qwen3 32B	0.8	4.2	43.6	30.0
Gemma 2 9B	0.4	4.3	17.2	29.6
Falcon-H1 7B	0.8	2.1	8.1	20.4
Qwen3 14B	2.3	3.1	67.1	15.6
Qwen3 30B-A3B	1.9	1.4	65.7	15.4
Qwen3 8B	1.9	1.4	66.8	9.6
GLM-4 9B	0.2	0.8	15.5	10.4
Qwen3 4B	1.3	1.3	59.5	7.1
Granite 3.3 8B	1.0	1.8	6.8	12.3
Ministral 8B	0.2	1.0	49.1	5.8
Qwen3 1.7B	0.2	0.3	70.9	1.5
API controllers				
GPT-5.6 Sol	5.2	42.3	0.9	99.6
Claude Opus 5	7.3	42.9	1.6	99.0
Gemini 3.6 Flash	6.7	27.4	0.2	96.2
GLM-5.2	3.8	11.4	24.9	68.5
DeepSeek V4 Flash	0.8	11.4	8.6	70.2
Llama 3.1 70B	2.3	22.7	4.1	88.1
Qwen3.7 Plus	2.9	6.6	8.2	42.9

Table 9: Canonical 19-controller exact-length panel. Initial/final are case-level $J_{\min}$ percentages; early events are per still-active transition into revisions 1–4. API entries use protocol-complete records (479 for Gemini, 480 for each other system); Table 1 and the interaction between controller and constraint family use the 479 all-model-common evaluable cases.

Table 10 averages controllers within each panel and then summarizes the three constraint-family means. The lexical column pools structures at 240/120/120; composition pools its two structures at 240/240.

B.2 Content Contract Robustness Check (Descriptive)

We apply seven additional conditions separately to the existing exact-length API endpoint annotations. This post-hoc sensitivity check remains outside the registered inference family and reports each condition separately. Missing annotation fields are treated as not applicable and do not fail the added condition; the denominator remains 480 cases per controller.

Llama remains ahead under every measured condition, including a 54.0-point gap under full applicable-entity retention. This pattern is inconsistent with an advantage sustained solely by the minimal anchor contract. These observables measure content retention; semantic equivalence and subjective quality remain unmeasured.

Controller	Exact	Lexical	Comp.	Mean ± SD
Local controllers ($n = 12$)	19.8	33.0	65.2	39.3 ± 23.4
API controllers ($n = 7$)	80.6	71.8	97.3	83.3 ± 12.9
All controllers ($n = 19$)	42.2	47.3	77.1	55.5 ± 18.8
Range across controllers	1.5–99.6	23.8–99.8	26.5–100.0	

Table 10: Revision-8 joint success (%) by panel and constraint family. Mean and SD summarize the three constraint-family columns.

Observable	Stored Audit Field / Check	Llama 3.1 70B	Qwen3.7 Plus
Complete entity retention	`draft_entity_retention` $= 1$	85.2	31.3
Complete proper-name retention	`draft_proper_name_retention` $= 1$	87.1	42.5
Complete symbolic-unit retention	`draft_symbolic_retention` $= 1$	87.7	42.9
No parser-fragment increase	`fragment_rate_change` ≤ 0	85.0	41.7
No parser-agreement decrease	`agreement_rate_change` ≥ 0	80.4	42.3
No parser-dangling increase	`dangling_rate_change` ≤ 0	87.9	42.5
No NLI-contradiction increase	`adjacent_contradiction_mean_change` ≤ 0	42.1	19.4

Table 11: Post-hoc descriptive check over single stricter content requirements. Values are joint success rates (%) for Llama 3.1 70B and Qwen3.7 Plus after adding each requirement to $J_{\min}$, over 480 cases per controller.

B.3 Inference and Execution Provenance

Local runs use greedy decoding (`do_sample=false`), bfloat16, an EOS padding token, and `max_new_tokens=max(128, 3N)`. Each checkpoint uses its own tokenizer and documented chat template with a generation prompt; thinking mode is disabled when the interface supports that option. The six-model original Combined-480 panel has command-level provenance for every one of its 18 model–split cells. Primary-120 and Replication-120 use the serial runner; Extension-240 uses a batch-one equivalent, with Qwen3 14B executed as two ordered 120-case shards. A guarded batch-four Llama trial failed exact serial equivalence on seven of eight gate cases, was aborted, and is excluded from the canonical output. The batch-one gate matched every case, round, and generated text.

Immutable analysis manifests record each input path, row count, and SHA-256 digest. The Qwen3 vLLM scale panel uses one frozen 480-case interface at 1.7B, 4B, 8B, 14B, 32B, and 30B-A3B; an independent audit verifies 480 unique case IDs for all twelve primary checkpoints. The environment record contains Python 3.10.19, PyTorch 2.9.1, and Transformers 4.57.6. Tokenizers 0.22.2 and Accelerate 1.12.0 were recovered post hoc. Launch records omit driver versions, full weight hashes, and a single immutable Qwen3 14B revision; the manifest marks these fields as missing. Exact reproduction of the Ministral row preserves the historical `fix_mistral_regex=false` tokenizer behavior.

B.4 Prompt and Decoding Robustness

The same 60 case IDs were evaluated for Llama 3.1 8B, Gemma 2 9B, and Qwen3 14B under three feedback prompt forms and three sampled decoding seeds. Llama ranks above Gemma, which ranks above Qwen3 14B, in every condition (Table 12). Absolute rates vary with the interface; controller ordering is stable.

C Closed-Loop and History Interventions

C.1 Response Estimation, Prediction, and Recoverability

Controlled response estimation. The controlled response experiment holds source text fixed and varies only the synthetic target and verifier message over requests $\{\pm1, \pm2, \pm5, \pm10, \pm20, \pm50\}$.

Checkpoint	Baseline	Structured	Concise	Sample Mean	Sample Range
Llama 3.1 8B	83.3	65.0	76.7	83.9	81.7–86.7
Gemma 2 9B	36.7	25.0	36.7	38.9	33.3–46.7
Qwen3 14B	18.3	11.7	10.0	13.3	11.7–15.0

Table 12: Final $J_{\min}$ (%) on the matched 60-case robustness panel. Sample statistics summarize three decoding seeds.

Unlike naturally occurring revision states, this design exposes every checkpoint to the same drafts and correction requests. The 24 discovery and 48 confirmation texts are disjoint. All ten checkpoints use the same grid and median-action estimator. Seven infeasible case–command cells imply targets below the 10-word floor and are excluded for every checkpoint, leaving 286 discovery and 571 confirmation actions per checkpoint (8,570 actions overall). Qwen3 32B and 30B-A3B were collected later under this same protocol; the analysis here integrates all ten checkpoints rather than reporting separate cohort aggregates. Model-specific curves and the shared curve are fit only on discovery data and then held fixed. The shared curve is the median of all ten checkpoints' discovery actions at each request. No confirmation or natural trajectory outcomes are used to fit either predictor. This is an expanded analysis of held-out data, not a new prospective confirmation gate.

Behavioral rates first average actions within each text and then average texts equally, for every checkpoint. Under this common weighting, Qwen3 30B-A3B has 59.5% zero word-count change, 35.0% correct direction, and 26.9% error contraction on confirmation texts. Its discovery–confirmation median-curve correlation is $\rho = 1.000$, with nine of twelve request levels tied at zero in both splits; this reflects rank reproducibility rather than correction accuracy. The corresponding correlation for Qwen3 32B is $\rho = 0.951$. The MoE comparison describes one checkpoint and does not isolate an architecture effect from training or computation differences.

Table 13 reports the Base/Instruct comparison separately for discovery and confirmation under this same text-balanced weighting. For requested change $c = -d_t$ and realized edit a_t, correct direction means $a_t c > 0$, error expansion means $|a_t - c| > |c|$, and zero action means $a_t = 0$ (unchanged word count, not necessarily identical text). These events are not a partition: an edit in the correct direction can overshoot far enough to increase absolute error. On confirmation texts, Qwen instruction tuning reduces error expansion from 47.3% to 14.2% while increasing zero action from 35.9% to 49.8%; discovery has the same directions. These are descriptive checkpoint contrasts under the controlled protocol, not an isolation of training from all other checkpoint differences.

Checkpoint	Split	Correct direction (%)	Error expansion (%)	Zero action (%)
Llama 3.1 8B Base	Discovery	27.4	41.8	53.0
Llama 3.1 8B Instruct	Discovery	87.5	24.0	5.9
Qwen3 8B Base	Discovery	39.5	40.6	35.3
Qwen3 8B Instruct	Discovery	44.9	13.8	52.5
Llama 3.1 8B Base	Confirmation	30.7	51.2	46.2
Llama 3.1 8B Instruct	Confirmation	90.1	21.1	5.8
Qwen3 8B Base	Confirmation	39.1	47.3	35.9
Qwen3 8B Instruct	Confirmation	47.4	14.2	49.8

Table 13: Controlled Base/Instruct behavior. Discovery uses 24 texts and 286 feasible text–request pairs per checkpoint; confirmation uses 48 disjoint texts and 571 pairs. Rates average requests within each text, then weight texts equally.

Held-out action prediction. The independent natural-trajectory evaluation includes nine checkpoints; Qwen3 8B Base has no matched Combined-480 trajectory panel. For each included checkpoint, all 72 controlled-study texts are excluded from its 480 unique cases, leaving 408 cases before transition filtering. We score only transitions with $0 < |c| \leq 50$ words. The final sample contains 3,014 checkpoint–case trajectories with at least one eligible transition and 20,069 transitions; the same source case may contribute to multiple checkpoints. Both predictors use piecewise-linear interpolation between measured requests, with $(0, 0)$ added as a construction convention rather than an observed response.

For each predictor, we average absolute prediction errors across eligible transitions within a case, across cases within a checkpoint, and then equally across the nine checkpoints. Let $\overline{\text{MAE}}$ denote this last mean. The single primary aggregate compares the model-specific and shared predictors:

$$R = \frac{\overline{\text{MAE}}_{\text{shared}} - \overline{\text{MAE}}_{\text{specific}}}{\overline{\text{MAE}}_{\text{shared}}}.$$

The resulting MAEs are 17.45 and 15.54 words, respectively: a reduction of 1.91 words (95% CI 1.58–2.25), or 11.0% (95% CI 8.5–13.9). Relative reduction is computed after averaging MAEs, not by averaging checkpoint-level percentages. All prediction intervals use 10,000 paired bootstrap samples of shared case IDs within source families, carrying each sampled case across checkpoints and repeating the same aggregation.

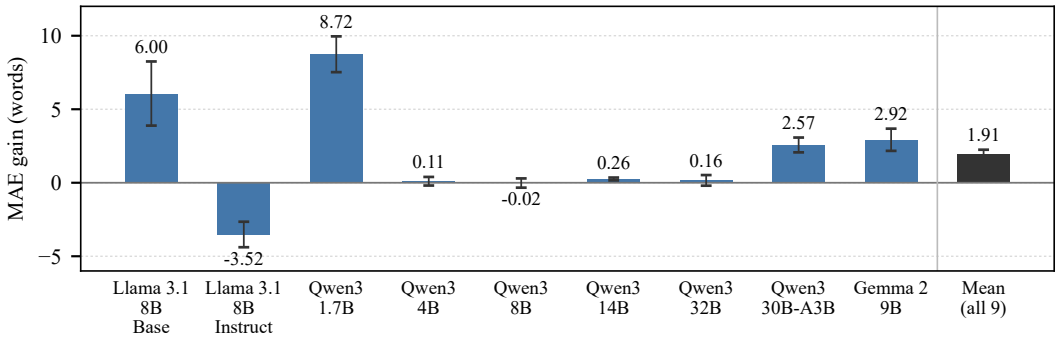

Figure 8: Natural-trajectory MAE gains (shared minus model-specific) for all nine eligible checkpoints, with paired-bootstrap 95% intervals. Discovery–confirmation curves appear in Figure 4.

Table 14 gives the per-checkpoint results and the zero-edit baseline on exactly the same transitions. The zero-edit baseline has lower point-estimate MAE than the model-specific curve for Llama Instruct, Qwen3 1.7B, 8B, 14B, 32B, and 30B-A3B. No intervals for those baseline contrasts are reported. A low prediction error can reflect small actual edits, even when the requested correction remains unmet. The comparison with the shared curve tests the predictive value of model identity, but does not establish that the full curve outperforms a fitted model-specific bias or gain.

Model	**Zero edit**	**Shared**	**Specific**	Gain [**95% CI**]
Llama 3.1 8B Base	86.23	88.67	82.67	+6.00 [3.89, 8.25]
Llama 3.1 8B Instruct	8.00	6.26	9.78	−3.52 [−4.39, −2.65]
Qwen3 1.7B	5.68	14.57	5.85	+8.72 [7.52, 9.96]
Qwen3 4B	12.10	11.27	11.16	+0.11 [−0.19, 0.40]
Qwen3 8B	7.17	7.80	7.82	−0.02 [−0.33, 0.29]
Qwen3 14B	4.28	5.83	5.58	+0.26 [0.16, 0.35]
Qwen3 32B	3.79	5.08	4.92	+0.16 [−0.20, 0.52]
Qwen3 30B-A3B	3.24	6.17	3.60	+2.57 [2.07, 3.07]
Gemma 2 9B	15.30	11.41	8.49	+2.92 [2.17, 3.68]

Gain = Shared − Specific; positive values favor model-specific prediction.

Table 14: Prediction MAE (words) on independent natural revision trajectories for all nine eligible checkpoints. Intervals for the shared-minus-specific gain are paired, source-stratified case bootstrap intervals. Lower MAE is better.

Contraction means $|d_{t+1}| < |d_t|$. It can occur even when an edit crosses the target, and net trajectory contraction does not require contraction at every step.

The case is the inferential unit. Endpoint intervals use source-stratified bootstrap; matched binary contrasts use exact McNemar tests and Holm correction; recurrence models cluster by case. Cross-checks recount stored text, recompute anchors, enforce zero split overlap, and require byte-identical crossover drafts. Because greedy trajectories can differ across backends, the Qwen scale pattern is reported under one uniform vLLM panel.

Conditional recoverability. The conditional-rescue analysis freezes revision 4 before API extension, includes only trajectories still unresolved with at least one later revision, and predicts capture through revision 8 from exact-state repeat fraction. The reported odds ratio compares repeat fractions of 1 versus 0. Its adjustment set contains current signed residual, absolute residual, log absolute residual, target, missing anchors, minimum absolute residual, improvement from the initial to current absolute residual, prior contractions, sign changes, no-op actions, largest action, length-basin exit, checkpoint, source, and length band. Sandwich covariance clusters identical case IDs across controllers; the retained Gemini provider refusal is excluded under the endpoint audit rule.

Table 3 in the main text reports the adjusted associations.

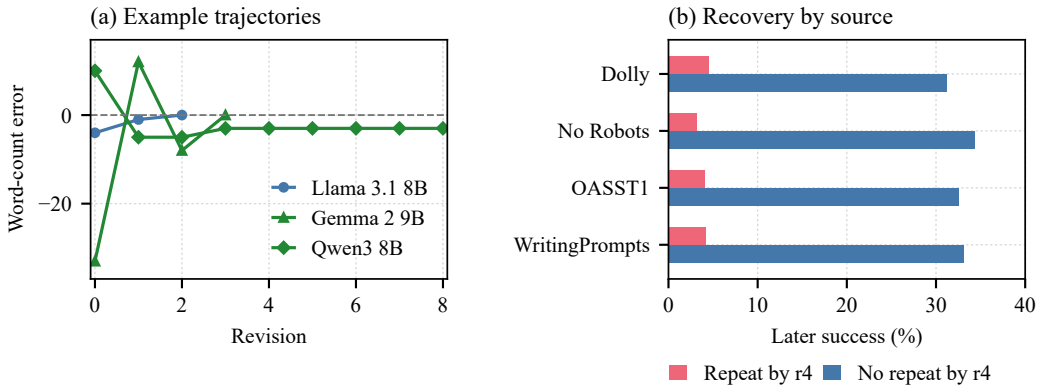

Figure 9: Matched exact-length trajectories show divergent controller dynamics, while recurrence by revision 4 is associated with lower later recoverability across all four sources.

Recurrence robustness and descriptive summaries. In the local panel, the unadjusted revision-4 association is negative in 11 of 12 checkpoints and all four sources and remains negative at revision-3 and revision-5 landmarks. Qwen3 1.7B is the ceiling-limited exception: 470 exposed and four unexposed cases yield a $+0.21$-point raw difference. In the API panel, the adjusted coefficient remains negative at both adjacent landmarks, the raw contrast is negative in all four sources, and both over-length and under-length subsets preserve the negative combined association. The raw within-controller direction is negative in four of seven API controllers. GPT, Gemini, and Claude have only 3, 1, and 3 recurrence-exposed risk-set trajectories, respectively; all four API controllers with at least 28 exposed cases are negative.

A preregistered early-capture/recurrence ratio ranks final success in both the local ($\rho = 0.783$) and API ($\rho = 0.857$) panels. Early capture alone ranks at least as strongly, and recurrence adds no held-family Brier-score improvement (Brier, 1950). We use the ratio as a descriptive controller summary.

A length-basin exit is any observed transition from $0 < |d_t| \leq 20$ to $|d_{t+1}| > 50$ by the landmark. Observed stationary failure means an unresolved trajectory whose final two outputs are identical, distinct from the three-output persistent fixed-point criterion above; its response-curve simulation analogue is unresolved capture with final predicted action magnitude below 0.5 words.

C.2 State-Matched History Intervention

The held-out first-recurrence screen produces 113 Llama and 194 GLM candidates. The paired contrast varies retained dialogue as a block and does not separately identify recency, position, or semantic content, which change together. Applying the frozen revision-≥ 3 rule excludes 28 Llama and 55 GLM states; the discovery-selected polarity rule then excludes 2 non-over-length Llama and 19 non-under-length GLM states, leaving 83 and 120. For each retained state, the two arms have the identical current draft, target, count, anchors, verifier message, decoding configuration, and remaining revision budget. The full-history arm preserves earlier turns; the reset arm preserves only the task, current draft, and current verifier state. Removing earlier dialogue jointly changes its content, length, turn structure, token positions, recency, and attention context.

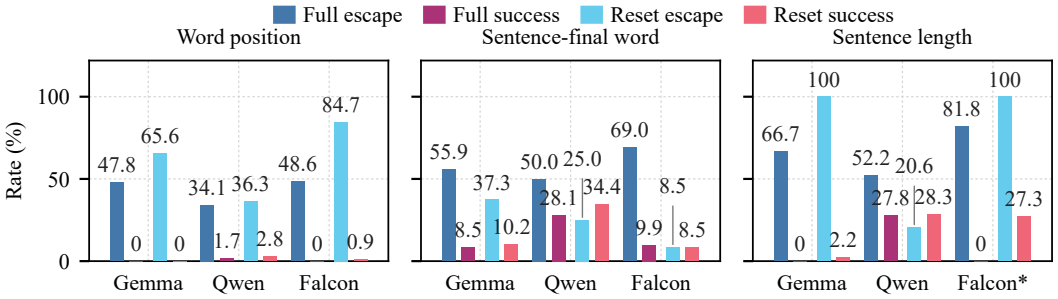

Figure 10: Escape and final success with full or reset history at matched recurrent states. Sentence length: Gemma $n = 45$; Falcon* $n = 11$.

Estimands are paired state-level differences in first-step recurrence escape, correct direction, absolute length-error contraction, later recurrence, final exact length, all-anchor retention, and final $J_{\min}$. Confidence intervals use 20,000 paired bootstrap resamples within source while preserving observed source sizes; they target the unweighted mean paired difference. Exact McNemar tests target the same paired marginal but condition on discordant pairs (19 for Llama final $J_{\min}$). The percentile interval and exact test differ in construction, so its exclusion of zero can coexist with $p = 0.063568$. The frozen treatment gate requires a final-$J_{\min}$ gain of at least 10 points, a positive bootstrap lower bound, and a nonnegative point estimate for all-anchor retention.

Figure 6 displays the exact-length paired treatment effects with stratified paired-bootstrap 95% intervals. Improvements in escape and final success are separate intervention effects, not estimates of mediation through escape. In absolute terms, Llama final joint success rises from 9/83 (10.8%) with full history to 18/83 (21.7%) after reset; GLM rises from 4/120 (3.3%) to 10/120 (8.3%). Thus 65/83 Llama and 110/120 GLM reset continuations still fail within the remaining budget. These absolute endpoints, rather than the magnitudes of the paired gains alone, establish that most reset continuations remain unsuccessful. The four partial-history conditions are categorical interventions on the same states, not a continuous history-dose scale.

C.3 Cross-Constraint History Effects

We compare full history and reset on 868 matched states from Gemma 2 9B, Qwen3 14B, and Falcon-H1 7B. For each case, we select the first eligible repeated draft before the trajectory ends. Selection is capped at 60 states for each combination of model, task structure, and source. The tasks require specified words at given word positions, specified words at sentence ends, or per-sentence word counts within given bounds. Word-position tasks also constrain total word count; the other two constrain sentence count.

History can help or hinder escape from repetition, depending on the model and task. Changes in escape need not be matched by changes in final success (Figure 10). For Gemma, escape increases by 17.8 percentage points on word-position tasks, with no change in final success. On sentence-length tasks, escape rises by 33.3 points and final success by 2.2 points. On sentence-final word tasks, escape instead falls by 18.6 points, while final success rises by 1.7 points. Final success is low in these Gemma comparisons. Qwen's sentence-length task provides a comparison with higher success rates: reset lowers escape by 31.7 points, while final success is 27.8% with full history and 28.3% with reset. The reset-minus-full success difference is 0.6 points (95% CI -2.8 to 3.9; Table 15).

State selection also matters. Some outputs already repeat at the first revision; resetting these states reconstructs the earlier prompt that produced the repeated output. Excluding these states reverses some escape effects, including Qwen on sentence-length tasks (Appendix F.3).

The targeted confirmation freezes 868 recurrent states and 1,736 paired arms: 284 Gemma 2 9B states, 391 Qwen3 14B states, and 193 Falcon-H1 7B states across c05, c10, and c12. Both arms share the current task, draft, verifier report, decoding, and remaining budget; only earlier dialogue is removed. Runtime manifests, pair keys, state hashes, and deterministic verifier replay pass for all arms. Falcon's fused-Mamba output was generated in disjoint resumable shards and merged only after an exact expected-arm and duplicate audit.

Primary effects are within-state reset-minus-full-history differences. Intervals use 20,000 constraint-by-source-stratified paired bootstrap draws. Family summaries macro-average estimable constraint cells with equal cell weight; the cross-family gate and c05/c12 controller-by-structure interaction were frozen before outcome aggregation. The interaction uses HC3 covariance and tests escape, contraction, and final success separately. Sparse cells are excluded from formal pooled estimates.

Model	Task	n	Escape Δ	Success Δ [95% CI]
Gemma 2 9B	Word position	180	+17.8	+0.0 [0.0, 0.0]
Gemma 2 9B	Sentence-final word	59	-18.6	+1.7 [-3.4, 6.8]
Gemma 2 9B	Sentence length	45	+33.3	+2.2 [0.0, 6.7]
Qwen3 14B	Word position	179	+2.2	+1.1 [-1.7, 3.9]
Qwen3 14B	Sentence-final word	32	-25.0	+6.2 [-9.4, 21.9]
Qwen3 14B	Sentence length	180	-31.7	+0.6 [-2.8, 3.9]
Falcon-H1 7B	Word position	111	+36.0	+0.9 [0.0, 2.7]
Falcon-H1 7B	Sentence-final word	71	-60.6	-1.4 [-7.0, 2.8]
Falcon-H1 7B	Sentence length*	11	+18.2	+27.3 [9.1, 45.5]

Table 15: Reset-minus-full differences (percentage points). Only final success has supplied paired 95% intervals; the starred Falcon sentence-length cell is descriptive.

C.4 Partial-History Matched Intervention

The GLM pilot selects 60 states first observed in deep recurrence (revision ≥ 4), balanced across c05, c10, and c12. Each state has four continuations: full history, the last two draft–feedback pairs, the last pair, and current task–draft–verifier state only. The 240 condition–state rows share the identical current state, model, decoding, and remaining budget and pass deterministic verifier replay.

The frozen primary family contains two paired escape contrasts, last-one minus full-history and last-two minus full-history, tested by exact paired tests with Holm correction. Confidence intervals use 20,000 constraint-by-source-stratified paired bootstrap draws and macro-average the three structures equally. Contraction, later recurrence, and final success are secondary endpoints. Table 16 reports all four conditions; Table 17 gives the primary contrasts.

History Retained	Escape	Contraction	Later Recurrence	Final Success
Full history	10.0	8.3	86.7	0.0
Last two pairs	25.0	16.7	88.3	0.0
Last pair	51.7	25.0	76.7	3.3
Reset	66.7	25.0	60.0	5.0

Table 16: GLM outcomes (%) across four history conditions on the same 60 states, equally weighted across structures.

Primary Escape Contrast	Difference (Points)	95% CI	Holm p
Last pair − full history	+41.7	30.0–53.3	1.19×10^{-7}
Last two − full history	+15.0	3.3–26.7	0.0352

Table 17: Frozen primary paired contrasts in the GLM partial-history pilot. The unit is the recurrent state; values macro-average c05, c10, and c12.

D Aligned Checkpoint Localization

This supplementary study uses Llama as a worked example and includes aligned Qwen3 8B and Gemma 2 9B checkpoints, with the input interface held fixed within each pair. The studies test error readout, Base-weight restoration into Instruct, activation transplantation, and Instruct-weight grafts into Base. They do not assume that corresponding layer ranges implement the same function across architectures.

Mechanistic localization. Causal tracing (Meng et al., 2022), activation patching (Zhang & Nanda, 2024), and weight grafting (Nief et al., 2025) provide tools for locating information and testing how model components affect behavior. Work on subspace choice (Makelov et al., 2024) and interactions

among mediators (Vaidyanathan et al., 2026) shows how intervention design affects causal attribution. We use probes and interventions to examine error information and revision behavior, with an aligned Llama Base–Instruct pair as the main case and complementary Gemma and Qwen interventions. We evaluate individual edits and complete revision trajectories separately to determine whether changes in local behavior accompany gains in final success.

D.1 Operations and Readout Conventions

Blocks and parameter groups. A block is one decoder layer, indexed from zero: blocks 0–31 in Llama, 0–35 in Qwen, and 0–41 in Gemma. A block intervention selects every checkpoint tensor under `model.layers.{i}.*`, including the attention, MLP, and within-block normalization parameters. The output package separately selects `model.norm.*` and `lm_head.*`. Unselected tensors retain the recipient checkpoint's values.

Probes. We read the representation at the final prompt token, before generating an edit, from the model's returned `hidden_states` tuple. Probe labels L0, L1, ... index this tuple, whose first entry is the embedding representation; they are not zero-based parameter-block labels. The prespecified readouts are L32 for Llama, L36 for Qwen, and L42 for Gemma. We standardize features using the training split and fit ridge readouts for error sign and required-correction magnitude (2 versus 10 words). Regularization is selected by case-grouped cross-validation on discovery data and frozen before confirmation. Readability does not establish that generation uses the decoded information.

Restoration and grafting. Restoration uses Instruct as recipient and Base as donor. For each selected tensor, half restoration sets $\theta' = 0.5\theta_{\text{Instruct}} + 0.5\theta_{\text{Base}}$; interpolation is computed in float32 and cast back to the recipient dtype. Full restoration copies the Base tensor, while sham preserves Instruct weights. Half restoration mixes every selected tensor, rather than replacing half the selected blocks. Grafting reverses the direction: selected Instruct tensors replace their counterparts in a Base recipient. These are weight interventions, without additional training.

Activation transplantation. This operation instead replaces donor activations in a recipient forward pass on identical tokenized inputs. The Llama experiments intervene on post-block residuals over the feedback-message span during prompt prefill, in both Base-to-Instruct and Instruct-to-Base directions; generated-token states are untouched. The subsections below specify the sites, samples, and behavioral criteria for each experiment.

D.2 Study Coverage and Evaluation Splits

For Qwen and Gemma, fixed-state discovery uses 24 cases (96 states), and confirmation uses 48 disjoint cases (192 states). The 480-case closed-loop comparisons include the discovery subset and are not wholly held out. Activation and graft findings below are restricted to completed discovery comparisons. A failed transplantation test does not establish the absence of a mechanism outside the tested sites, spans, or intervention scheme.

Pair	Probes and Restoration	Activation Transplantation	Weight Grafts
Llama	Held-out probes; restoration confirmed on 360 disjoint cases	Discovery complete; mediation criterion unmet	Discovery complete; direction improves, calibration criterion unmet
Qwen	Held-out probes and fixed-state dose confirmation; 480-case loop comparison	Discovery complete; no selected group passes	Discovery complete; no selected group passes
Gemma	Held-out probes and fixed-state dose confirmation; 480-case loop comparison	Discovery complete; no selected group passes	Partial transfer in discovery; confirmation incomplete

Table 18: Coverage and confirmation status of the aligned mechanism studies. Passing criteria are study-specific.

D.3 Llama Residual Probes

Meta Llama 3.1 8B Base and Instruct receive byte- and token-identical Meta serialization on 288 fixed feedback states from 72 cases. Each case supplies residuals $d \in \{-10, -2, +2, +10\}$; 24 cases form discovery and 48 disjoint cases form confirmation. Representations are captured from the returned hidden-state tuple at the final prompt token, using the readout convention defined above. Ridge regularization is selected separately for each checkpoint and readout by six-fold GroupKFold over discovery cases at layer 32 and then frozen across layers; confirmation data are not used to select regularization. Layer-32 estimates are prespecified; the layer sweep below is descriptive. Primary readouts are layer-32 error-sign accuracy and required-correction magnitude classification (2 versus 10 words); best-layer values are descriptive.

Model	Checkpoint	Error Sign	Required Magnitude
Llama 3.1 8B	Base	90.1	62.5
Llama 3.1 8B	Instruct	88.0	85.4
Qwen3 8B	Base	80.7	80.7
Qwen3 8B	Instruct	88.0	80.7
Gemma 2 9B	Base	94.3	69.3
Gemma 2 9B	Instruct	92.2	65.6

Table 19: Prespecified final-layer probe accuracy (%) on 192 held-out states per checkpoint. Magnitude distinguishes 2 from 10 words.

The probe table reports the prespecified final hidden-state readout: L32 for Llama, L36 for Qwen, and L42 for Gemma. All use the final prompt token; required-correction magnitude refers to the requested 2- versus 10-word change, not the magnitude of the generated edit. Readout accuracy does not by itself show that generation uses the decoded information.

A descriptive layer sweep shows the output-layer correction-magnitude deficit is not an absence of the underlying quantity: R^2 rises from 0.08 at layer 32 to 0.62 for Base at layer 14, and from 0.44 to 0.93 for Instruct at layer 8. These layers are selected on the split used to report them, so the values are upward-biased and not comparable to the preregistered layer-32 estimates.

D.4 Llama Parameter Restoration

Base weights are restored into an otherwise fixed Instruct checkpoint for four contiguous eight-block bands, final norm plus LM head, a sham copy, and an eight-block noncontiguous parameter-count control. The discovery screen uses 96 fixed states and Main-120. Selected blocks 8–15 and the output package are then evaluated at sham, half, and full restoration and confirmed on 360 cases absent from discovery. Fixed-state intervals resample 24 case clusters; closed-loop contrasts are paired by case and source. Figure 11 reports final-success losses relative to sham with source-stratified paired-bootstrap 95% intervals. These tests establish necessity under the intervention; unique contiguity remains unresolved.

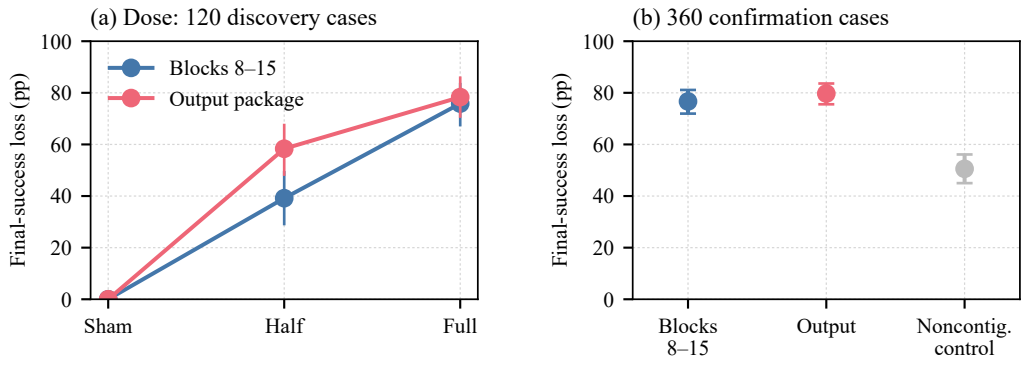

Figure 11: Base-weight restoration: discovery dose response (a) and disjoint confirmation (b). Bars show paired-bootstrap 95% intervals.

D.5 Llama Activation Transplantation

On the 96 discovery states, donor and recipient share identical token IDs. During prompt prefill only, the complete feedback-message span is transplanted in both directions either across post-block residuals 8–15 or at the post-block-31 output boundary; generated-token states are untouched. For each unit and residual family (magnitude 10 correction or magnitude 2 near-target), we report the clean checkpoint contrast, Instruct-to-Base denoising effect, Base-to-Instruct noising loss, and their interaction on direction and clipped calibration utility. Instruct-to-Base transplantation raises correct revision direction by 20.8–29.2 points while worsening magnitude calibration. No unit passes the frozen bidirectional mediation gate.

D.6 Llama Controller Graft

Five Base-recipient conditions share the Meta interface and greedy decoding: sham, blocks 8–15, output package, their joint graft, and a matched noncontiguous-plus-output control. Each runs 96 fixed states and Main-120; intervals resample case or source strata. The joint graft improves correct editing direction but fails the prespecified calibration criterion. Calibration utility is $-\min(|a/c-1|, 10)$ for actual word-count change a and requested change $c \neq 0$. The gate requires nonnegative mean utility differences against both Base and the noncontiguous-plus-output control; the observed differences are -0.268 and -2.199, respectively. The direction gain therefore does not meet the criterion for transferring effective revision behavior.

On the 96 fixed states, direction accuracy is 25.0% for Base, 59.4% with the block graft, 46.9% with the output graft, and 81.25% with their joint graft. The joint graft succeeds on 10 of 120 separate closed-loop cases (Figure 12).

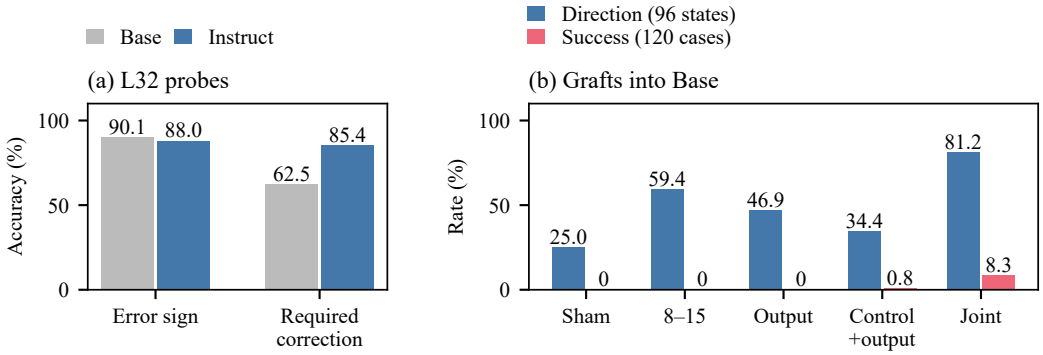

Figure 12: Llama probe readouts and all graft conditions. Direction and final success use separate fixed-state and closed-loop samples.

D.7 Qwen: Restoration Effects Without Successful Transfer

Qwen3 8B Base and Instruct share a 36-block architecture and the same Instruct serialization. Tokenizer and tensor-shape audits pass. Probe regularization is selected using discovery cases and frozen before confirmation (Table 19). Error direction remains readable in both checkpoints. Error expansion denotes an increase in absolute word-count error after an edit, $|d_{t+1}| > |d_t|$. The stored fields call this outcome `overshoot` (or its complement `overshoot_avoidance`); it need not involve crossing the target.

The restoration screen tests four nine-block bands, the output package, a sham, and a parameter-count-matched noncontiguous control. Blocks 0–8 and 9–17 are selected in discovery. On the 192 held-out fixed states, full restoration increases error expansion by 62.0 and 21.4 percentage points, respectively; paired 95% intervals are [50.0, 72.9] and [10.9, 32.8]. Half-restoration effects are weaker. Both groups satisfy the frozen restoration criteria, including comparison with the noncontiguous control. The 480-case closed-loop comparisons show lower success and larger terminal error (Table 20).

Bidirectional activation transplantation covers 384 patched discovery cells across the two selected groups. Neither group passes the required behavioral-direction and calibration criteria. Seven graft conditions compare Base, output only, noncontiguous blocks plus output, and each selected block group with and without output. Neither joint graft passes the selected behavioral-signature criteria against Base and the control. On fixed discovery states, the two joint grafts increase error expansion relative to Base by 28.1 and 31.3 percentage points, respectively. No activation or graft confirmation split is opened. Thus, the restoration results establish sensitivity to these weight changes, without showing that the tested components transfer effective revision behavior.

D.8 Gemma: Different Effects on Repetition and Correction

Gemma 2 9B Base and Instruct use a common interface and matched tokenized fixed states. The discovery restoration screen selects blocks 21–30 and 31–41 for half- and full-dose comparisons with sham and a parameter-count-matched noncontiguous control. Both groups meet the prespecified criteria on the 192 held-out fixed states. Full restoration of blocks 21–30 lowers correct direction by 65.6 percentage points [57.8, 73.4], whereas restoring blocks 31–41 increases error expansion by 33.9 points [23.4, 44.3]. These interventions therefore produce different forms of poor revision, rather than a uniform loss of activity.

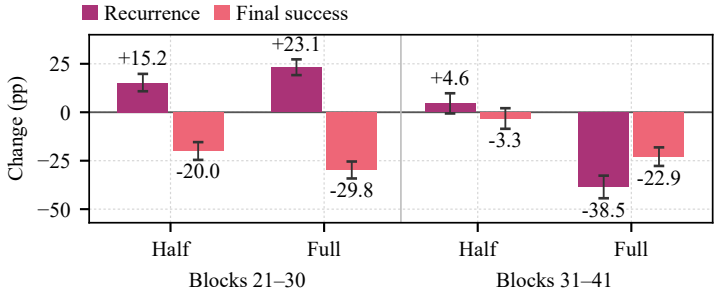

Figure 13: Gemma Base-weight restoration into Instruct. Changes relative to sham in recurrence and final success across 480 cases, with source-stratified paired-bootstrap 95% intervals. Half and full denote interpolation weights 0.5 and 1.

Model	Blocks	Repeat Δ (pp)	Success Δ (pp)	Error Δ (words)
Qwen3 8B	0–8	-5.4 [-9.0, -1.9]	-6.5 [-9.0, -4.0]	+77.9 [+71.2, +85.0]
Qwen3 8B	9–17	+3.5 [+1.0, +6.0]	-5.2 [-7.9, -2.5]	+15.6 [+11.6, +20.1]
Gemma 2 9B	21–30	+23.1 [+19.2, +27.3]	-29.8 [-34.2, -25.4]	+80.5 [+70.7, +90.6]
Gemma 2 9B	31–41	-38.5 [-44.4, -32.7]	-22.9 [-27.7, -18.1]	+118.8 [+107.6, +130.6]

Table 20: Full restoration minus sham on 480 cases per model, with source-stratified paired-bootstrap 95% intervals.

Here recurrence means any exact repeated output among the recorded drafts, including the initial draft, across all 480 cases; it is not conditioned on final failure as in Figure 5. Restoring blocks 21–30 increases recurrence by 23.1 points and lowers success by 29.8 points. Restoring blocks 31–41 instead reduces recurrence by 38.5 points while lowering success by 22.9 points and increasing terminal absolute error by 118.8 words. The latter success rates are 32.5% for sham and 9.6% after restoration. These are joint consequences of the weight intervention; they do not identify a causal effect of recurrence on success. Broad disruption from weight replacement also limits anatomical interpretation.

Neither selected group passes the bidirectional activation-transplantation criteria on 384 discovery cells. Grafting Instruct blocks 21–30 plus output into Base does improve direction from 51.0% to 74.0% on 96 fixed states, and success from 1/120 to 11/120 on separate discovery trajectories (Table 21). Calibration improves relative to Base but remains below the noncontiguous-plus-output control.

The Gemma discovery rule requires improvement on a selected behavior relative to Base and the control, with no contradiction on other selected behaviors relative to Base. Unlike Llama's calibration

Condition	Direction (%)	Calibration Utility	Final Successes
Base	51.0	-8.99	1/120
Output	50.0	-8.82	0/120
Noncontiguous + output	63.5	-3.54	3/120
Blocks 21–30	75.0	-5.46	8/120
Blocks 21–30 + output	74.0	-5.13	11/120
Blocks 31–41	36.5	-3.18	2/120
Blocks 31–41 + output	34.4	-3.21	1/120

Table 21: Gemma graft discovery: 96 fixed states and 120 separate closed-loop cases. Higher calibration utility is better.

rule, it does not require calibration to be at least as good as the control; discovery also does not require an interval excluding zero. Blocks 21–30 pass this screen, whereas blocks 31–41 fail because direction deteriorates. The subsequent graft confirmation is incomplete and is excluded from all reported estimates and conclusions. We therefore describe the 21–30 result as discovery-only partial transfer, not confirmed recovery.

E Aligned Post-Training and Interface Comparisons

The Llama comparison in Appendix D.3 uses Base and Meta Instruct under byte-identical Meta serialization. As a complementary behavioral comparison, we evaluate five released Llama–Tülu checkpoints (Lambert et al., 2024) under a common plain-text interface held fixed across this aligned panel. The four chat-capable checkpoints are also evaluated with their own bundled native templates; Base has no native chat template and is excluded from that comparison. Cases are shared across checkpoints within each constraint family, while the three constraint families retain their distinct datasets. Table 1 retains the canonical controller panel; this appendix reports a separately frozen aligned-checkpoint ablation.

Checkpoint	Interface	Exact Length	Lexical	Compositional
Llama 3.1 8B Base	Common	0.6	22.1	0.4
Tülu 3 8B SFT	Common	4.4	21.9	29.0
	Native	4.6	22.9	33.1
Tülu 3 8B DPO	Common	15.0	25.2	42.5
	Native	15.4	24.0	64.6
Tülu 3.1 8B RLVR	Common	12.5	28.5	45.6
	Native	16.7	25.8	48.8
Llama 3.1 8B Instruct	Common	47.7	46.5	91.0
	Native	79.0	50.8	97.3

Table 22: Final joint success (%) across aligned checkpoints under common and native interfaces.

Structure-level rates explain the strong interaction between checkpoint and constraint family under the common interface ($\chi^2(8) = 1094.58$, $p = 5.66 \times 10^{-231}$). Base's 22.1% lexical average comes entirely from c07 (88.3%), with 0% on c05 and c12. SFT leaves exact length and the lexical average nearly unchanged, yet reaches 55.4% on c09 and only 2.5% on c10. DPO reaches 26.3% on c10 and 15.0% on exact length; RLVR is higher on c12 and c10 but lower than DPO on exact length and c09. Performance across the released checkpoints therefore changes differently for different task structures (definitions are in Appendix A.1).

The effect of native formatting also varies across task structures. SFT's modest $+4.2$-point compositional change combines a 13.3-point decrease on c09 with a 21.7-point increase on c10, whereas DPO's $+22.1$-point gain combines increases of 13.8 and 30.4 points. This changes the DPO–RLVR ordering: RLVR leads by 3.1 points under the common interface, but DPO leads by 15.8 under native serialization, a paired difference-in-differences of 19.0 points (95% bootstrap CI 12.9–25.2). Meta Instruct has a different signature, gaining 31.2 points on exact length (26.0–36.3). Interface-by-checkpoint tests reject a uniform template bonus for exact length ($\chi^2(3) = 112.72$, $p = 2.85 \times 10^{-24}$), lexical constraints ($\chi^2(3) = 22.40$, $p = 5.40 \times 10^{-5}$), and compositional constraints ($\chi^2(3) = 46.03$, $p = 5.58 \times 10^{-10}$). The common interface compares checkpoints under the same formatting; native templates measure their deployed behavior. Neither comparison isolates the causal effect of a training stage.

F Additional Revision Diagnostics

F.1 Fixed-Draft Crossover

The crossover uses 240 common cases, with 60 from each source. Three Llama–peer pairs form ten unique draft-generator–reviser combinations: four diagonal cells and six off-diagonal cells. All 2,400 trajectories pass task, anchor, target, initial-text, and endpoint recount checks. The six within-draft reviser contrasts are tested by exact McNemar tests with Holm correction. Table 2 groups final-success rates by draft source and identifies each reviser explicitly. Draft-source effects are not uniformly zero: two point estimates favor peer drafts by 8–9 points, but no draft-source contrast survives six-test Holm correction at .05. Table 23 reports the six paired contrasts and the analysis conditional on initial failure. Excluding initially successful drafts leaves 225 cases for Llama drafts, 234 for Qwen drafts, and 240 each for Gemma and GLM drafts. The conditional advantage ranges from 45.3 to 77.1 percentage points. Each excluded case succeeds initially under both revisers because the draft is identical, so exclusion changes the denominator but not the discordant-pair counts or exact test results. The paired intervals in the all-case panel pertain only to that panel; the conditional panel reports its frozen rates, differences, and tests.

Draft source	Peer	n	Llama (%)	Peer (%)	Difference (pp)	Holm p
All cases; differences include paired-bootstrap 95% intervals						
Llama	Qwen	240	76.2	17.1	59.2 [52.5, 65.8]	4.59×10^{-37}
Qwen	Qwen	240	84.6	15.8	68.8 [62.5, 75.0]	2.78×10^{-45}
Llama	Gemma	240	76.2	33.8	42.5 [35.0, 50.0]	5.10×10^{-22}
Gemma	Gemma	240	81.2	25.4	55.8 [49.2, 62.5]	1.65×10^{-37}
Llama	GLM	240	76.2	12.9	63.3 [56.7, 69.6]	1.23×10^{-39}
GLM	GLM	240	85.4	8.3	77.1 [71.2, 82.5]	4.44×10^{-51}
Initially unsuccessful drafts only; conditional point estimates						
Llama	Qwen	225	74.7	11.6	63.1	4.59×10^{-37}
Qwen	Qwen	234	84.2	13.7	70.5	2.78×10^{-45}
Llama	Gemma	225	74.7	29.3	45.3	5.10×10^{-22}
Gemma	Gemma	240	81.2	25.4	55.8	1.65×10^{-37}
Llama	GLM	225	74.7	7.1	67.6	1.23×10^{-39}
GLM	GLM	240	85.4	8.3	77.1	4.44×10^{-51}

Table 23: Fixed-draft reviser contrasts. Llama: 3.1 8B; Qwen: Qwen3 14B; Gemma: 2 9B; GLM: GLM-4 9B. Differences are Llama minus peer. Frozen all-case intervals use 20,000 case-paired bootstrap resamples. Intervals apply only to the all-case panel. Exact McNemar tests are Holm-corrected over six comparisons separately in each panel.

The crossover changes the deployed reviser and native interface jointly.

F.2 Capture and Budget

The API trajectory audit contains 3,359 protocol-complete trajectories: 479 Gemini records and 480 for each other system. The budget replay uses 479 all-system-common cases instead. Conditional capture is computed among trajectories unresolved before each revision. Its mean over revisions 5–6 is no lower than over 1–2 for every system, but the late means range from 6.4% to 58.7%. A low fixed-budget endpoint need not imply a decreasing per-round capture probability. Among the 650 failures, 90.2% end at their own best absolute residual. This does not rule out intermediate oscillation or instability. Five trajectories attain exact word count and later leave it; this is not loss of joint success. The exact-length diagnostic labels follow the priority rule in Appendix A.1. Figure 15 retains the dense Qwen scale comparison and residual-distance diagnostic. The latter uses disjoint integer bins; zero denotes a best-at-terminal trajectory, not stable dynamics.

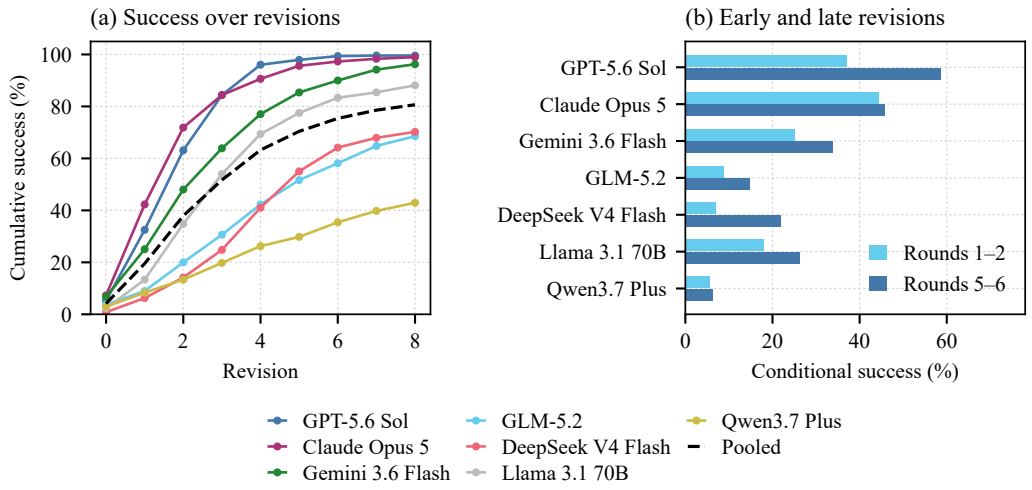

Figure 14: Cumulative capture and conditional capture under the eight-revision budget on the protocol-complete API panel.

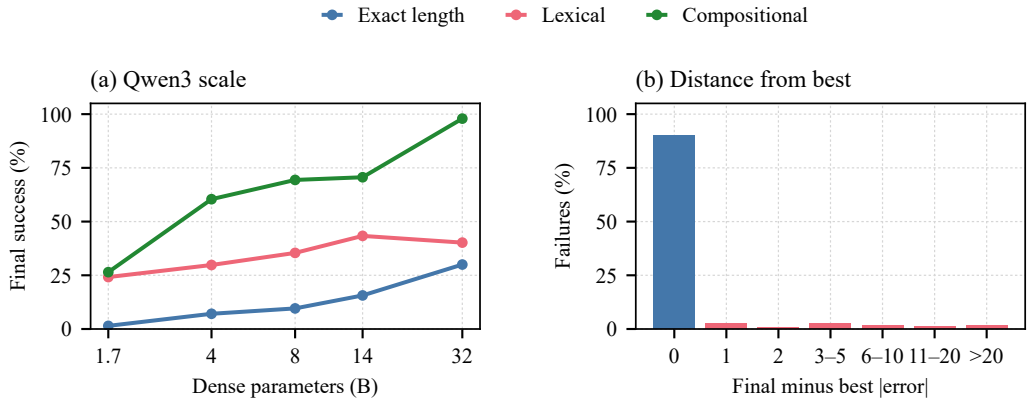

Figure 15: Additional diagnostics: dense Qwen scale across constraint families and final distance from each failed trajectory's best residual.

F.3 Trigger-State Sensitivity of History Reset

All 209 first-revision reset states replay the previously failing prompt; 199 repeat the trigger text. For a first-step fixed point, the current draft and feedback equal the initial draft and feedback, so deleting the history reconstructs the earlier input. This protocol property contributes to mixed-cohort contrasts. Table 24 reports all controller–structure cells, including small cells, rather than selecting only sign reversals. The subset analysis is retrospective and does not replace the frozen full-cohort estimand. Removing first-revision states changes the population; it is not a causal effect of intervention timing.

For Qwen c10, the later-state subset has 92 states: escape changes from 19.57% to 32.61%, and final success from 10.87% to 14.13%. The final-success difference is 3.26 points, with a descriptive 95% interval of $[-3.26, 9.78]$, recomputed using 5,000 source-stratified paired bootstrap draws (seed 20260908). The original cell-rate CSV supplies no escape intervals; no interval is imputed from its marginal rates.

G Failure Trajectories across Constraint Families

We classify unsuccessful protocol-complete trajectories separately for exact length, lexical constraints, and compositional constraints. All 19 model configurations are included. There are 27,359 complete

Model	Task	n	Escape Δ	Later-state n	Escape Δ
Gemma	Word position	180	+17.8	162	+25.9
Gemma	Sentence-final word	59	-18.6	48	-4.2
Gemma	Sentence length	45	+33.3	45	+33.3
Qwen	Word position	179	+2.2	150	+12.7
Qwen	Sentence-final word	32	-25.0	18	+5.6
Qwen	Sentence length	180	-31.7	92	+13.0
Falcon	Word position	111	+36.0	109	+37.6
Falcon	Sentence-final word	71	-60.6	24	-16.7
Falcon	Sentence length	11	+18.2	11	+18.2

Table 24: Reset-minus-full escape effects (percentage points), in all states and after excluding first-revision triggers.

trajectories and 12,170 failures: 5,272 exact-length, 4,805 lexical, and 2,093 compositional failures. One protocol-incomplete Gemini exact-length record is excluded. These descriptive counts use all complete records, rather than restricting every API model to the 479 cases shared for paired endpoint comparisons.

A common classification rule. Let $y_0, \ldots, y_K$ be the recorded drafts, including the initial draft, where K is the last recorded revision index. A trajectory is a failure when its final draft does not satisfy all constraints for its task. Text equality means exact UTF-8 equality without normalization; equal word counts or equal verifier residuals do not establish equal outputs. Among failed trajectories, assign the first matching category below:

1. **Terminal fixed point.** At least three drafts are recorded and $y_K = y_{K-1} = y_{K-2}$. Earlier drafts need not be identical. This is an observed terminal pattern, not a claim that repetition would continue forever.
2. **Terminal cycle.** For at least one $p \in \{2, 3, 4\}$, the last $2p$ drafts consist of two identical consecutive blocks of p drafts, each containing at least two distinct outputs. For example, a trajectory ending in A, B, A, B satisfies the period-2 test when $A \neq B$. We do not require the cycle to begin at the initial draft or claim to identify its minimal period. The fixed-point category takes priority if both conditions hold.
3. **Other repeat.** Some $y_i = y_j$ with $i < j$, but neither terminal condition holds. For example, A, B, A, C contains repetition without a terminal fixed point or tested cycle. A trajectory can receive this label even if its final output differs from every earlier one.
4. **No repeat.** All recorded drafts are distinct. This label describes text novelty only: it does not imply error reduction, correct editing direction, or proximity to satisfying the constraints.

These four categories are mutually exclusive and exhaustive under the stated priority rule. They describe output trajectories, not distinct causal mechanisms. Unlike the seven exact-length categories in Appendix A.1, they do not use word-count thresholds, anchor status, or signed-error flips. Consequently, a repeated trajectory ending at exact word count with missing anchors remains in a repetition category here, although the seven-category classifier assigns anchor failure. The two classifications should not be compared as if their labels were interchangeable.

Reconstruction and aggregation. Local labels are reconstructed from the existing frozen per-trajectory indicators for fixed points, terminal cycles, and any output recurrence. API labels are recomputed from the original recorded drafts after checking input hashes against the existing analysis manifest. Each model contributes 480 complete trajectories per constraint family, except Gemini exact length with 479. Lexical summaries preserve the original 240/120/120 allocation to c05/c07/c12; compositional summaries preserve the 240/240 allocation to c09/c10. Figures 16–18 show within-family compositions. Each bar is normalized by that model's failure count, marked n; an empty bar marked "No failures" has no defined failure composition. Panel (a) pools failed trajectories within API, local, or all models, so models with more failures contribute more. They are not averages that weight models or constraint structures equally.

Constraint Family	Terminal Fixed Point	Terminal Cycle	Other Repeat	No Repeat	Total
Exact length	2,922	715	833	802	5,272
Lexical constraints	2,348	418	1,017	1,022	4,805
Compositional constraints	1,491	172	225	205	2,093

Table 25: Failure counts under the common output-trajectory classification.

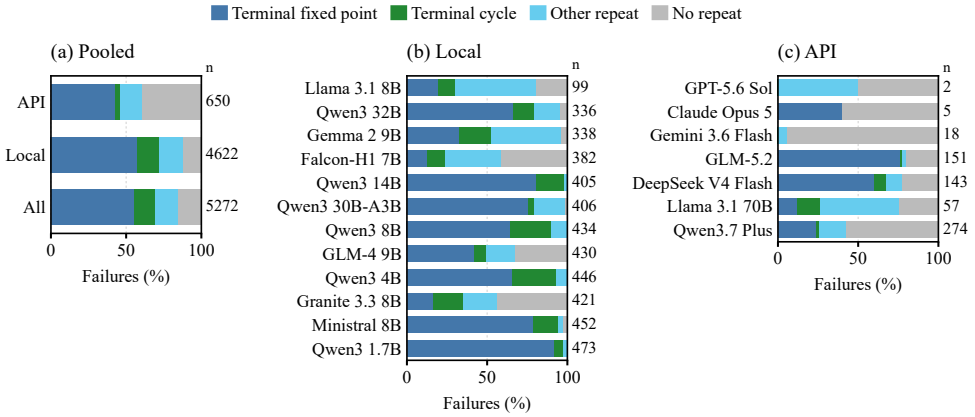

Figure 16: Exact-length failures: API, local, and pooled compositions (a), and all 19 models (b,c). Colors and within-panel model order match Figure 5 and Table 1.

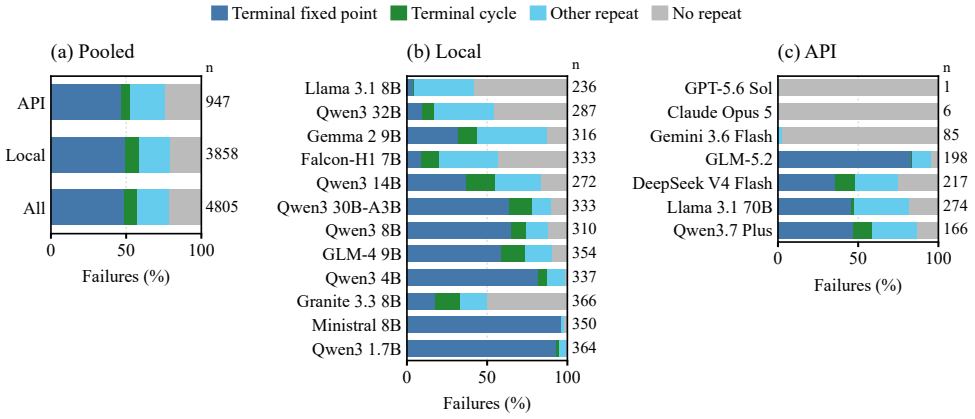

Figure 17: Lexical failures: pooled compositions (a) and all 19 models (b,c); colors and model order follow Figure 16.

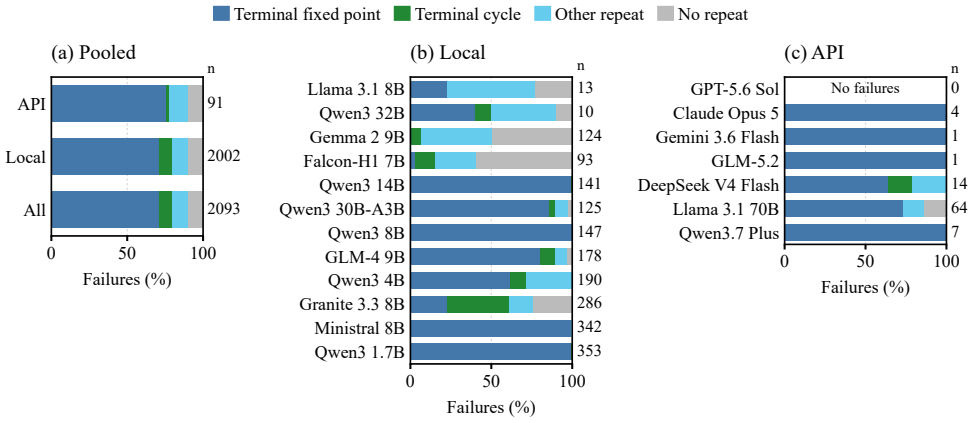

Figure 18: Compositional failures: pooled compositions (a) and all 19 models (b,c); colors and model order follow Figure 16.