WHEN CAN A SKILL LIBRARY TRUST ITS EXECUTION RECORDS?

Anonymous authors
Paper under double-blind review

ABSTRACT

A skill library may use a successful execution record to predict an update's effect, decide whether to deploy it, or justify replacing the full skill with shorter text. These are different judgments. On sixteen known QA questions, records of the same proposal at other receivers reduce squared F1-effect risk by 0.00797 relative to equal-budget records of other proposals at the same receivers. That information raises the value of a fixed retrospective gate by 0.324 F1 points over its matched control, yet the gate's own value relative to rejecting every update is estimated at -0.591 points (95% interval $[-1.425, 0.042]$). Receiver-only and pooled histories also attain lower absolute prediction risk than same-proposal history. In a separate crossed study, three local trials beat pooled records in two settings and lose in two others as the balance of local execution noise and nonlocal error reverses. Finally, *performance substitution*—replacing a full skill with one section while maintaining its mean score within a prespecified margin—changes answer distributions and source outcomes under GLM. Execution records do transfer useful information, but prediction, deployment, and retention of particular successes require distinct comparisons.

1 INTRODUCTION

A skill library has retained a textual update after it improved its source. Should the library install that update in another document? It can consult archived executions or pay for local trials; it may also store a shorter passage in place of the full skill. Each choice appeals to the update's history, but each asks that history to support a different judgment. Reflection, reasoning memories, evolving playbooks, and skill optimizers all turn execution feedback into reusable instructions (Shinn et al., 2023; Ouyang et al., 2026; Zhang et al., 2026; Yang et al., 2026). The practical question is what their execution records warrant at the next use.

Three judgments are easy to conflate. A record may help *predict* the effect of installing an update in a receiving document. A good prediction may still lead to a poor *deployment* decision if the predicted gain is small or the available policy is weak. And a shorter instruction may preserve the *mean score* while changing which earlier answers survive. We separate these claims experimentally. The result is a sharper account of reuse: a successful record is evidence for a specified effect, action, or outcome, not a general credential for all three.

Our central test asks whether information travels with a fixed proposal across receivers. A *record* averages repeated paired execution differences for one update, receiver, and question. We predict an update's effect at a target receiver from its records at two other receivers, then compare against records of other proposals at those exact receivers. Receiver identity and the two-record budget are held fixed; all target-pair executions are excluded. The target question is already known to every predictor through records at other text combinations. This design isolates proposal-associated predictive information rather than a claim about unseen questions. Figure 1 shows who can read which executions.

The proposal's history does carry information. Same-proposal records reduce squared F1-effect risk by 0.00797 relative to the matched control on the sixteen known questions (Section 3). Yet receiver-only and pooled histories attain lower absolute risk, and the better matched prediction does not turn the resulting gate into an established profitable policy. That gate improves F1 decision value by 0.324 points over its matched counterpart, while its own value relative to rejecting every update is

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.

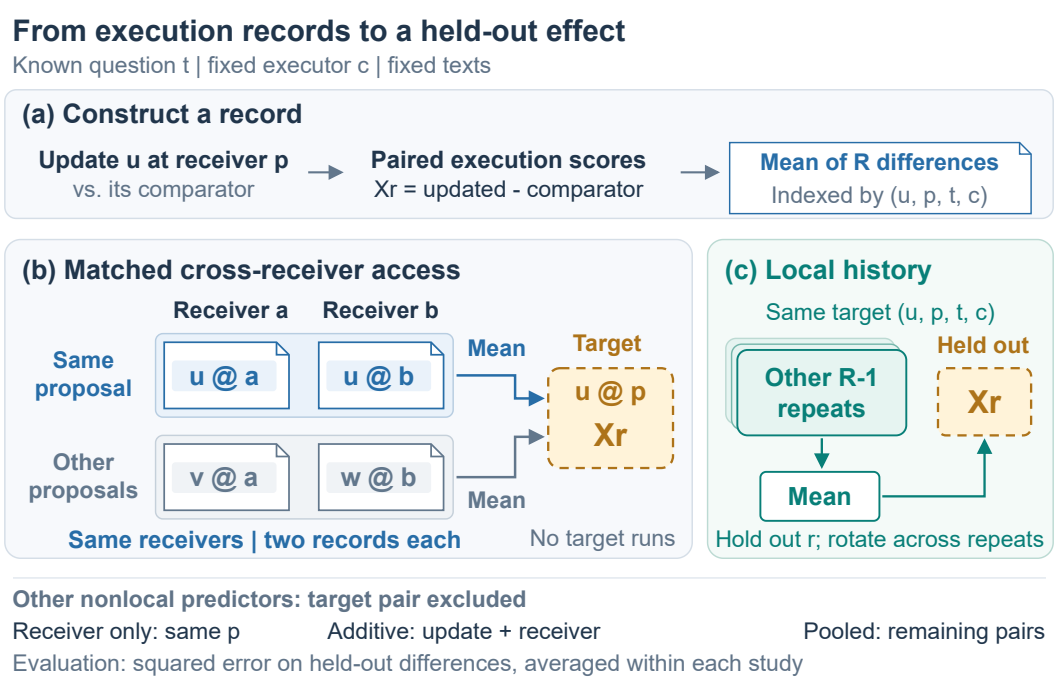


Figure 1: Information available for a receiving-effect prediction. (a) Paired scores yield a difference X_r; a record averages R repeats. (b) The natural-queue comparison changes proposal identity while matching receivers and record count; all target-pair runs are excluded. (c) Local history predicts a held-out difference from the other $R - 1$ target repeats. Each study specifies its own comparator and repetitions.

estimated at -0.591 points. The positive predictive finding and the unfavorable decision estimate belong together: they show exactly why information in a record is not a deployment guarantee.

Complementary studies separate what the successful source contributes. A source endpoint combines inherited performance with the update's current increment (Section 2); content controls find a benefit from instructional text over inert filler without resolving a utility premium for its exact source correspondence (Section 5).

Buying local evidence introduces a separate budget decision. In a fully crossed panel, three local observations beat pooled history in two thinking-off configurations and lose in two high-thinking configurations. A finite-record identity explains this reversal by comparing local sampling noise with the nonlocal predictor's error (Section 4). Locality supplies relevance; repetitions supply precision. Their balance determines whether spending a small budget locally improves the prediction.

Shortening a skill poses another question. Under GLM, *performance substitution*—replacing a full skill with one section while maintaining its mean within a prespecified margin—coexists with changed answer distributions and source outcomes (Section 6). A general counterexample shows why even high mean score and answer agreement cannot guarantee retention of the full skill's new successes. The three judgments therefore call for different comparisons. A reusable library record should retain the intervention, comparator, receiver, question, executor, and individual outcomes needed to make them.

2 What a Reuse Decision Needs to Estimate

A source score can reward work done by the previous instruction. To evaluate reuse, the library needs the change produced by an update against an explicit comparator. Let u denote a fixed update, p a receiving document, t a question, and c an executor configuration. We call p a *receiver*: the document an update is installed in for measurement. A document pool can supply both source and receiver

roles. The conditional effect is

$$\theta_{uptc} = \mathbb{E}[s_t(Y^u_{ptc}) - s_t(Y^0_{ptc})], \tag{1}$$

where Y^u uses the installed update and Y^0 its comparator. In the natural queue the comparator is the unchanged receiver; in the fully crossed study it is another substantive replacement text. Scores are token F1 or exact match (EM) in $[0, 1]$. The expectation concerns repeated execution of fixed texts. Positive effects represent gains and negative effects losses.

An endpoint and an increment answer different questions. For no-skill instruction N, source parent P, and updated text U, expected score satisfies

$$V(U) - V(N) = \underbrace{V(P) - V(N)}_{\text{inherited performance}} + \underbrace{V(U) - V(P)}_{\text{current increment}}. \tag{2}$$

Here P and U are measured on their own workflow; no second receiver enters the decomposition. In a natural-revision source panel separate from the prediction queue, Executor A (an archived executor configuration) has inherited performance of -16.02 F1 points, a retained update adding 4.92 (95% interval $[0.43, 9.30]$), and an endpoint contrast with no skill of -11.11. The original-interface analysis uses 128 common questions and six source states with two events each; the full six-configuration panel appears in Appendix I.

An independent DeepSeek queue illustrates the converse. Across 24 events from twelve sources, inherited performance is $+4.21$, the retained increment is -0.22 ($[-1.27, 0.77]$), and the endpoint is $+3.99$ ($[0.64, 7.67]$). These are descriptive 95% source-block/question intervals under the tag-cleaned interface (Appendix E). The positive endpoint primarily reflects inherited performance. The endpoint describes the final instruction; the increment describes what this update added. Installing it elsewhere calls for another increment in the receiving context.

A record and a prediction target. Suppress c and write $X_{uptr} = s_t(Y^u_{ptr}) - s_t(Y^0_{ptr})$ for repeat r. A *record* is the mean $\bar{X}_{upt}$ of R such paired differences. We predict a held-out execution difference and average squared error over target pairs and questions. Each nonlocal predictor reads records for the *same question* at other text combinations, with the entire target pair excluded. Local history predicts repeat r from the other $R - 1$ executions of that exact combination. Figure 1 summarizes these access rules.

The matched comparison changes which update supplies two records: the target update at two other receivers, or other updates at those same receivers. It identifies information associated with proposal identity. Receiver-only and pooled histories provide broader context; local history measures what direct observations add.

Units and uncertainty. The prediction panels use supplied-context QA with four question categories, separate from optimizer training and development. Prediction risk is squared error for effects on the $[-1, 1]$ scale; utility differences are F1 or EM points on the 0–100 scale. Prediction-risk intervals are individual descriptive bootstrap intervals over each panel's specified support, with predictions fixed. Supporting studies retain their own text collections, score interfaces, and comparison families (Appendix N).

3 Proposal Information Across Receivers on Known Questions

On a known question, a proposal's records at other receivers predict its effect better than records of other proposals matched on receiver identity and history size. The comparison isolates information associated with proposal identity while keeping the prediction target explicit.

A natural proposal queue. Twelve source blocks each generate twelve proposals using DeepSeek V4 Flash with thinking disabled. Each proposal uses eight optimizer-training and four optimizer-development questions. All 144 proposals enter the analysis, including the 143 rejected by development selection. The common evaluation set has sixteen held-out questions, four per task category; the records are archived executions of the fixed deployed texts.

Each proposal is installed in three receivers other than its source. The assignment varies: a source block contains 36 observed proposal–receiver pairs over eleven receiver documents, each used in three or four pairs, so the panel is sparse rather than a complete crossing with three fixed receivers. Installation appends the identical proposal in a delimited update field and preserves the receiver outside it; the deployed text is fixed for three executions per proposal–receiver–question combination. Comparing each execution with a corresponding execution of the unchanged receiver yields 20,736 paired score differences. All predictors read records for the *same known question* as their target, so the transfer tested here runs across text combinations.

Matching receivers and record budgets. The same-proposal predictor averages the proposal's records at its two other receivers; its matched control reads a different proposal at each of those exact receivers, again two records each averaging three executions. For every admissible pair of alternative records we compute the prediction and its risk, then average those risks, holding receiver identity and record count fixed while changing whether the records concern the target proposal.

Table 1: Prediction risks in the natural proposal queue. Lower is better. Entries are squared F1- or EM-difference risks for scores in $[0, 1]$. A record averages three executions; local history uses individual repetitions of the target combination. All predictors use the target question. The first two rows share receiver identity and record count, forming the matched contrast. The remaining rows draw on different histories and budgets, providing absolute-risk context for the matched result.

Predictor	Available history	F1 risk	EM risk
Same proposal	2 records	0.092998	0.115901
Matched other proposals	2 records	0.100968	0.131498
Receiver only	2 or 3 records	0.068065	0.084309
Additive	35 records	0.099505	0.117547
Pooled	35 records	0.068875	0.088412
Local history	2 executions	0.027337	0.028356

Same-proposal records reduce F1 prediction risk by 0.00797 (interval $[0.00131, 0.01797]$) relative to the matched control, and EM risk by 0.01560 $[0.00648, 0.02704]$; Table 1 gives both absolute risks. With the donor receivers and record budget held fixed, knowing which proposal produced a record improves the forecast under both measures.

Other records can predict better. Proposal identity is informative, but receiver context and broader history also matter (Table 1). Receiver-only prediction averages the two or three other proposals at the target receiver; pooled prediction averages the other 35 pair records in its source block. Additive prediction fits proposal and receiver main effects after deleting the entire target pair, including all three repetitions. The sparse design stays full rank after every deletion, so the fitted effects remain identified.

Receiver-only and pooled prediction have lower observed risks than same-proposal prediction. Local history uses the other two executions of the exact proposal–receiver–question combination, reducing F1 risk further to 0.02734. Its improvements over same-proposal, receiver-only, and additive prediction all have individual 95% intervals excluding zero (Appendix F). It has already observed the target deployment.

Intervals use 2,000 source-block/question bootstrap draws with predictions fixed. The twelve source blocks and sixteen shared questions determine their resampling support.

Does better prediction help a deployment decision? A post-hoc replay accepts an update at a receiver when its mean predicted effect across the sixteen known questions is positive; a zero prediction rejects it. We define *decision value* as the mean score change relative to the unchanged receiver across all 432 placements, counting rejected updates as zero. The target pair's three repetitions evaluate each accepted update. Each admissible matched donor pair supplies its own gate, with the same donors across questions; outcomes are then averaged over donor pairs (Appendix C).

Same-proposal evidence improves decision value by 0.324 F1 points (95% interval $[0.013, 0.717]$) over matched other-proposal evidence. The same-proposal rule has a negative point estimate of -0.591 $[-1.425, 0.042]$ relative to rejecting every update; the matched rule's value is -0.914

$[-1.894, -0.144]$. A receiver-only gate has value $+0.202$ $[-0.631, 1.124]$, using a different history. Always accepting yields -2.656, whereas the pooled F1 gate rejects all updates and yields zero.

Intervals reweight evaluation outcomes across source blocks and questions with decisions fixed on the original sixteen-question panel. The gate is chosen and evaluated on that same question panel, although target-pair executions never enter its predictions. Acceptance rates and the separately defined EM gate appear in Appendix C.

A post-hoc question-category holdout sets each gate on twelve questions and assesses the four withheld questions. Its same-minus-matched F1 decision-value difference is -0.044 points, with a descriptive interval of $[-0.550, 0.551]$ (Appendix D). This internal stress test asks about cross-question action rather than the original same-question prediction, and its contrast is imprecise.

The original gate makes the matched information useful relative to a weaker gate, but its own absolute value is not established as positive. Predictive information is real; a beneficial deployment still needs its own comparison.

4 When Are Local Trials Worth Their Budget?

A local trial observes the deployment of interest, but three noisy executions need not beat a broad history. In a separate fully crossed panel, local history outperforms pooled records in two thinking-off configurations and underperforms them in two high-thinking configurations. The reversal turns the choice of evidence into a measurable budget question.

A fully crossed replacement panel. Forty paired replacement texts span four instruction families and replace a designated section of each of twelve receivers. Both alternatives, g_0 and g_1, contain substantive instructions, so the effect is $s(g_1) - s(g_0)$. Four executions per alternative on twenty common questions produce 76,800 responses per configuration. The analysis uses sixteen natural questions, four per category, excluding four diagnostic probes.

The configurations are DeepSeek V4 Flash and Pro, each with thinking off or high. At each target question, pooled prediction reads the other 479 replacement–receiver records, replacement-only reads the other eleven receivers, and receiver-only reads the other 39 replacements. Additive prediction fits replacement and receiver effects with the entire target cell excluded. Each record averages four paired observations; local history predicts a target repetition from the other three observations of that exact combination. Complete risks appear in Appendix G.

Separating the two costs. A short local mean carries sampling noise; a pooled prediction carries error in approximating the target effect. Their balance determines when the local mean improves prediction. For a fixed prediction z, target mean $\bar{X}$, and sample variance s_X^2 of R repetitions,

$$\widehat{\mathcal{R}}(z) = (z - \bar{X})^2 + \frac{R-1}{R} s_X^2, \qquad \widehat{\mathcal{R}}_{\text{local}} = \frac{R}{R-1} s_X^2. \tag{3}$$

With $R = 4$, average over target cells to define $V = \overline{s_X^2}$ and $D = \overline{(z_{\text{pool}} - \bar{X})^2}$. The observed comparison is exactly

$$\widehat{\mathcal{R}}_{\text{pool}} - \widehat{\mathcal{R}}_{\text{local}} = \underbrace{(D - V/4)}_{\widehat{A}} - \underbrace{V/3}_{\widehat{N}_3}. \tag{4}$$

Under iid target execution and independence of nonlocal predictions from target executions, $\widehat{N}_3$ estimates the local mean's noise cost $\sigma^2/3$, and $\widehat{A}$ estimates the pooled prediction's error for the latent target effect, $(\mathbb{E}Z - \theta)^2 + \mathrm{Var}(Z)$, averaged over this panel. The finite-table equalities themselves require no independence assumption (Appendices A and B).

The observed reversal has two components. Relative to thinking off, the estimated local noise cost is about twice as large in each high configuration, while pooled error is much smaller (Table 2). All four pooled-minus-local intervals exclude zero, and receiver-only and additive comparisons have the same directions (Appendix G). This comparison describes the deployed settings rather than isolating a causal effect of thinking.

A risk-based budget condition. For n independent local paired observations, their mean has lower expected squared prediction risk than Z precisely when $\sigma^2/n < (\mathbb{E}Z-\theta)^2+\mathrm{Var}(Z)$. The inequality states what another local trial must overcome: sampling noise must fall below the nonlocal estimate's error for the target effect. The observed $\widehat{A}$ and $\widehat{N}_3$ locate this comparison at three observations; in the high configurations $\widehat{A}$ is near zero with an interval spanning zero. Further observations could also be combined with the nonlocal estimate, as discussed in Section 8.

Table 2: Numerical explanation of the local-history reversal. Positive $\widehat{A}-\widehat{N}_3$ favors local history. Entries use squared F1-effect units; Figure 2 shows individual descriptive 95% intervals.

Configuration	Local noise $\widehat{N}_3$	Pooled error $\widehat{A}$	Pooled − local
Flash, off	0.00521	0.02176	+0.01655
Pro, off	0.00271	0.02212	+0.01941
Flash, high	0.01120	0.00127	−0.00993
Pro, high	0.00582	0.00027	−0.00555

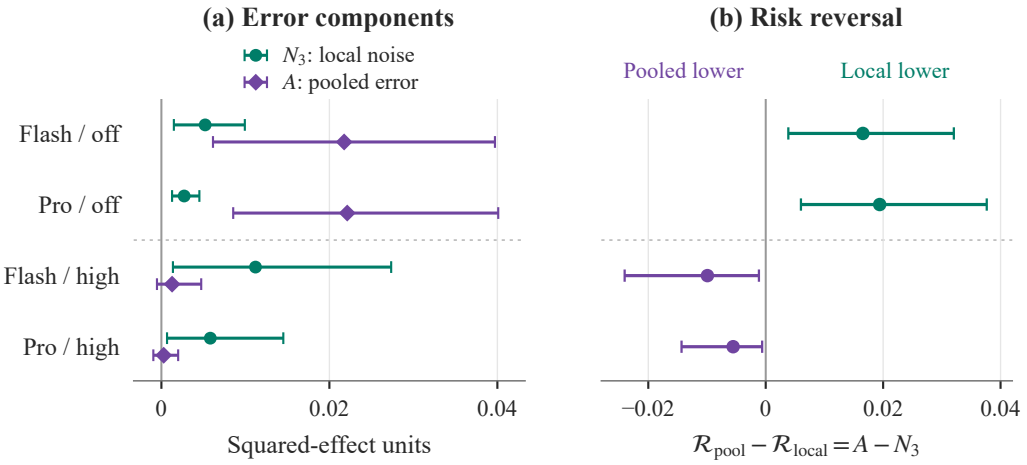


Figure 2: The two costs behind finite local evidence. Left: estimated local-mean noise and pooled error for the target effect. Right: their difference gives the observed prediction-risk comparison. Intervals use 5,000 crossed-bootstrap draws with predictors fixed. Untruncated $\widehat{A}$ intervals cross zero in the high configurations; a negative bound reflects estimation uncertainty, not a negative true squared error.

5 Predictive Information Is Not a Reusable Gain

The matched study found information attached to proposal identity. Does an update's instructional content help at a new receiver, and does correspondence with its own source add utility? Two studies using text collections separate from the prediction panels test these questions and the role of the receiving instruction.

Content, source correspondence, and net benefit. A five-arm study fixes the text outside an update field and varies what fills it. Twenty-eight new source lineages produce 56 proposals, nine passing development selection; all enter evaluation at three receivers each, none the proposal's own source. On sixteen questions with three executions under historical DeepSeek V4 Flash, the arms are the source update, a shuffled update from another source event, matched unrelated prose, empty delimiters, and the unchanged receiver. Inert prose matches characters and whitespace-token counts within 5%, with identical paragraph and heading counts (Appendix H).

Table 3: Content controls on 168 placements. F1-point differences have individual descriptive 95% intervals, without adjustment across the six planned comparisons. Source updates are evaluated at receivers other than their own source. *Unchanged receiver* removes the update region.

Comparison	Difference	95% interval
Source update − matched inert prose	$+3.610$	$[0.047, 8.134]$
Shuffled update − matched inert prose	$+3.393$	$[0.071, 7.634]$
Source update − shuffled update	$+0.217$	$[-1.475, 2.349]$
Source update − unchanged receiver	$+2.117$	$[-0.831, 6.355]$
Matched inert prose − empty delimiters	-0.720	$[-2.485, 0.811]$
Empty delimiters − unchanged receiver	-0.772	$[-2.061, 0.553]$

Both source and shuffled updates beat unrelated filler in their individual comparisons: instructional content matters. The source-minus-shuffled estimate is $+0.217$ points with an interval spanning zero, and the source-minus-unchanged comparison is similarly unresolved (Table 3). The 2,000 bootstrap draws resample source lineages, events, receiver slots, and stratified questions; recurring receiver identities and shuffled-donor links are not separately cross-resampled.

This result gives the matched prediction finding a sharper meaning. Proposal identity helps forecast a particular effect (Section 3); the content controls do not establish an additional utility premium for matching an update to its own source event. Predictive provenance and receiving utility call for different comparisons.

The receiving instruction supplies part of the effect. A frozen natural repair adds passages that improve Executor A F1 by 9.03 points on the parent document and 4.16 after the old reasoning body is removed (paired 95% intervals $[5.02, 13.31]$ and $[1.07, 7.52]$). Under DeepSeek V4.1 Flash, the same insertions yield -1.43 and -0.98, with both intervals spanning zero. These four-repeat, 128-question comparisons use tag- and polarity-normalized scores (Appendix Q). Passage utility is a property of the receiving intervention: the background instruction and executor belong in the comparison that supports deployment.

6 Preserved Scores, Changed Answers

A shorter skill can match a full skill's average score while answering different questions. We call *performance substitution* replacing a full skill with part of it at effectively unchanged mean score, operationalized here by a prespecified ± 1 F1-point margin. The retention question is whether the full skill's newly successful answers survive.

Mean score and even high answer agreement cannot certify that retention. Proposition J.2 constructs, for every $0 < \varepsilon < 1/2$, two equally improved deployments that agree on $1 - 2\varepsilon$ of answers while retaining none of the same newly achieved successes: they agree on a large baseline-correct set and repair disjoint errors. We therefore measure both mean score and source outcomes in an extraction study. It compares a starting instruction G, twelve full generated skills F, and the designated *Answer Format* sections S each skill contributes; extraction retains 18.4% of a skill's characters on average. A hash-selected 128-question HotpotQA panel (Yang et al., 2018) receives four executions per condition and block, with starting-prompt calls shared across blocks and a primary interface that canonicalizes eligible yes/no answers. The generator, its fixed five-section schema, and the disjoint training groups are given in Appendices K and L.

Mean replacement and distributional change coexist. Under GLM-5.3-Flash (GLM hereafter), section-minus-full F1 is -0.002 points, with a 98.75% question-by-block interval of $[-0.89, 0.82]$, inside the prespecified ± 1 margin. This meets the performance-substitution criterion: one section matches the full skill's mean within the chosen margin. Full-minus-start is $+0.58$ (95% interval $[-0.67, 2.05]$); the full skill's improvement over G remains unresolved.

Independent repeats distinguish instruction change from ordinary output variability. At a fixed question–block input x, let $p_T(a \mid x)$ be the normalized-answer distribution under text T, B_x the

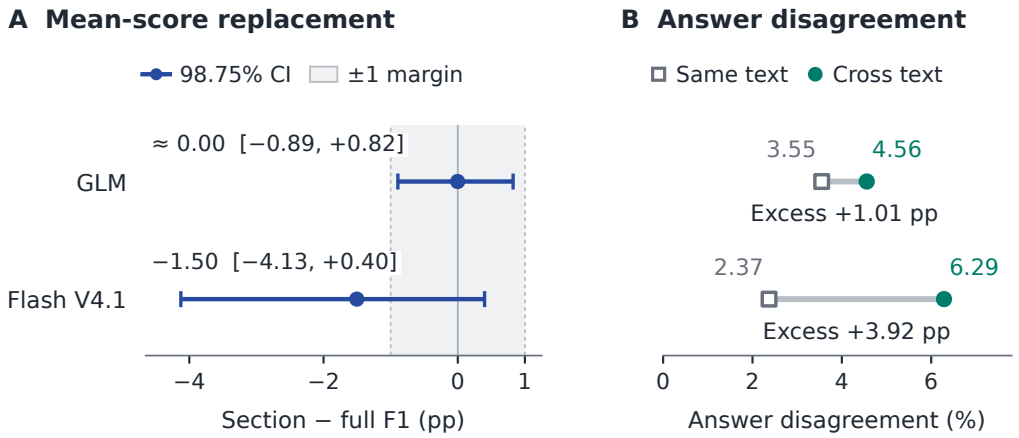


Figure 3: Extraction on twelve skill blocks, 128 questions, and four executions. Left: section-minus-full F1, 98.75% intervals, and the prespecified margin. Right: cross-text answer disagreement and the mean within-text reference. Both configurations show distributional change (Holm-adjusted $p < 0.001$); only GLM establishes score replacement.

cross-text disagreement rate, and $W_{F,x}, W_{S,x}$ the within-text rates. Then

$$E_x = B_x - \tfrac{1}{2}(W_{F,x} + W_{S,x}) = \tfrac{1}{2}\sum_a [p_F(a \mid x) - p_S(a \mid x)]^2. \tag{5}$$

This is half the squared equality-kernel MMD (Gretton et al., 2012). GLM's cross-text rate is 4.56%, against a 3.55% same-text reference, giving 1.01 percentage points of excess disagreement. The within-input permutation test detects different answer distributions (Holm-adjusted $p < 0.001$). DeepSeek V4.1 Flash shows the same distributional change, with 3.92 points of excess, and a score difference of -1.50 (98.75% interval $[-4.13, 0.40]$). Both executors show distributional change; only GLM establishes score replacement (Figure 3).

Fresh source controls give the change a direction. Equal means can hide gains replacing losses. On the historical GLM records, the section retains 99 of the full skill's 143 exact-match gains over G, loses 44, recovers 41 source losses, adds 45 new successes, and loses 34 shared successes: a net change of eight successful cells, describing recorded answers rather than stable response types (Appendix J).

Fresh executions revisit the frozen 143 gain and 120 loss cells. Relative to a freshly executed full skill, the section's EM changes are -9.79 and $+10.00$ points; fixed-input 95% execution bounds are $[-17.82, -1.76]$ and $[1.23, 18.77]$, while wider question-by-block intervals $[-20.45, 0.60]$ and $[-2.48, 23.96]$ both include zero. Re-executing F controls for departures from retrying historically selected responses, so on this recorded panel the fixed-input contrasts support both lost source matches and recovered failures (Appendix M).

Under GLM, mean-score replacement and changes in particular answers coexist. The source-outcome accounting shows the losses and recoveries hidden by the mean; the general construction explains why no aggregate score can certify retention of the full skill's new successes.

7 Related Work

Learning and reusing textual guidance. Feedback supports iterative refinement (Shinn et al., 2023; Madaan et al., 2023), prompt optimization (Pryzant et al., 2023; Yuksekgonul et al., 2025; Agrawal et al., 2026), and reusable memories (Ouyang et al., 2026; Zhang et al., 2026). SkillOpt and Trace2Skill examine evolving skills and their transfer (Yang et al., 2026; Ni et al., 2026). Our object is the paired execution record of a fixed update at a receiving document.

Prediction from prior evaluations. PromptEval shares sparse prompt–example observations to estimate performance (Polo et al., 2024); INSTINCT learns a surrogate for instruction search from evaluation history (Lin et al., 2024). We compare paired update effects using identical donor receivers and record counts, then test what the matched information changes in a deployment gate.

Interpreting effects under repeated execution. Aggregate scores can conceal trace changes (Zhou et al., 2026); component studies examine prompt optimization (Melcer et al., 2025), while multi-round studies examine skill evolution (Liu et al., 2026). Repeated-agent evaluation and score-interface analysis motivate checking more than one run and one metric (Yao et al., 2025; Hua et al., 2025). We combine a prespecified score-replacement margin, a same-text reference for answer change (Gretton et al., 2012), and accounting of source outcomes.

8 Discussion

Execution history supports reuse through distinct comparisons. Same-proposal records contain predictive information across receivers, yet the resulting gate's relative improvement does not establish a positive deployment payoff. A library must connect the predicted effect to the value of its available action. Source provenance supplies another kind of information: both source and shuffled instructions beat inert prose in the separate content panel, while the utility premium for exact source correspondence remains unresolved. Remembering where an instruction succeeded and measuring what it does in the receiving context serve different decisions.

The budget result makes this distinction operational. Locality supplies relevance; repetitions supply precision. With three local observations, sampling noise exceeds nonlocal prediction error in the two high-thinking configurations and falls below it in the two thinking-off configurations. More relevant evidence can therefore be a worse use of a small budget. The finite-record identity identifies the comparison an evidence-allocation policy must make: the precision bought by another local trial against the error of the available nonlocal predictor. Evaluating such a policy would also charge the paired comparator calls used to acquire its evidence.

Maintaining a successful correction adds an outcome requirement. Under GLM, an extracted section meets the full skill's mean-score criterion while changing particular answers and source outcomes. Retained gains, lost gains and recovered failures show whether a replacement preserves the success that motivated storing the skill. The counterexample establishes why high score and overall answer agreement cannot supply this accounting. Redundant successful coverage can also produce negative additive interaction while preserving successes (Appendix T). Mean utility, incremental benefit and success retention thus have distinct roles in maintaining a library.

These results motivate records that keep the proposal and installed-text versions, receiver, comparator, question, executor, scoring interface, paired outcomes and aggregation rule together. Such records would let a later decision choose evidence for its target and inspect which source successes survive substitution. This is a design implication of the comparisons, not a measured benefit of an implemented library.

The matched test concerns 144 development proposals, 143 of them rejected, on sixteen known questions. The 20,736 paired differences sharpen comparisons within that panel; category holdout supplies an internal cross-question check. An independent question panel would test the same-question prediction contrast, while evaluation of a gate frozen before those questions would test deployment. The configuration and extraction studies address complementary judgments.

9 Conclusion

A skill's successful record is evidence for a particular inference, not a general warrant for reuse. On known questions, same-proposal executions carry information across receivers, but their predictive advantage does not itself establish a beneficial deployment. A mean-preserving section can change the answers a full skill produced. Skill libraries can make better use of their histories by asking three separate questions: what effect will this update have, is installing it worthwhile, and which earlier successes must survive?

Ethics Statement

The study uses public QA and simulated business-task benchmarks, without human-subject experiments or actions in real business accounts. Reused instructions can introduce deployment errors; this analysis makes their effects observable but does not certify safe autonomous deployment.

Reproducibility Statement

The appendices specify text constructions, task supports, executor configurations, scoring interfaces, call aliases, resampling units, and comparison families. They provide full condition means, contrasts, intervention texts, diagnostic records, and estimator derivations. The source package includes portable paired-score inputs, scripts, and full results for the budget analysis, gate replay, and post-hoc question-family holdout; none of these reanalyses makes new model calls. Statistical figures are reproducible from machine-readable values and the included analysis scripts. Shared calls and fixed diagnostic inputs retain their distinct roles in uncertainty calculations. Natural-proposal, controlled-replacement, and repeated-execution panels are documented separately.

AI Use Statement

Language-model tools assisted with manuscript drafting and restructuring, conceptual framing, hypothesis organization, methodological and mathematical review, interpretation of existing results, code development, data inspection, and figure preparation. Model-assisted answer annotations are identified separately from benchmark scores and checked as described in Appendix R. The authors are responsible for the scientific claims, analysis, citations, and final manuscript.

References

Lakshya A Agrawal, Shangyin Tan, Dilara Soylu, Noah Ziems, Rishi Khare, Krista Opsahl-Ong, Arnav Singhvi, Herumb Shandilya, Michael J Ryan, Meng Jiang, Christopher Potts, Koushik Sen, Alexandros G. Dimakis, Ion Stoica, Dan Klein, Matei Zaharia, and Omar Khattab. GEPA: Reflective prompt evolution can outperform reinforcement learning. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/0e9e708b6f48e14fd0ac29e167413f76-Abstract-Conference.html.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL https://www.jmlr.org/papers/v13/gretton12a.html.

Andong Hua, Kenan Tang, Chenhe Gu, Jindong Gu, Eric Wong, and Yao Qin. Flaw or artifact? rethinking prompt sensitivity in evaluating LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 19889–19899. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1006. URL https://aclanthology.org/2025.emnlp-main.1006/.

Xiaoqiang Lin, Zhaoxuan Wu, Zhongxiang Dai, Wenyang Hu, Yao Shu, See-Kiong Ng, Patrick Jaillet, and Bryan Kian Hsiang Low. Use your INSTINCT: INSTruction optimization for LLMs usIng neural bandits coupled with transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 30317–30345. PMLR, 2024. URL https://proceedings.mlr.press/v235/lin24r.html.

Yuxuan Liu, Zhaochen Su, Yuhao Zhang, Jiahe Guo, Zhongwei Xie, Huihao Jing, Lingyun Xie, Qing Zong, Yauwai Yim, Zhixiong Zhang, Haoran Li, and Yangqiu Song. Rethinking self-evolving agent skills: Feedback dynamics over multiple rounds. *arXiv preprint arXiv:2608.02636*, 2026. URL https://arxiv.org/abs/2608.02636v1.

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 *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

Daniel Melcer, Qi Chen, Wen-Hao Chiang, Shweta Garg, Pranav Garg, and Christian Bock. Textual gradients are a flawed metaphor for automatic prompt optimization. *arXiv preprint arXiv:2512.13598*, 2025. URL https://arxiv.org/abs/2512.13598v1.

Jingwei Ni, Yihao Liu, Xinpeng Liu, Yutao Sun, Mengyu Zhou, Pengyu Cheng, Dexin Wang, Erchao Zhao, Xiaoxi Jiang, and Guanjun Jiang. Trace2skill: Distill trajectory-local lessons into transferable agent skills. *arXiv preprint arXiv:2603.25158*, 2026. URL https://arxiv.org/abs/2603.25158v5.

Siru Ouyang, Jun Yan, I-Hung Hsu, Yanfei Chen, Ke Jiang, Zifeng Wang, Rujun Han, Long T. Le, Samira Daruki, Xiangru Tang, Vishy Tirumalashetty, George Lee, Mahsan Rofouei, Hangfei Lin, Jiawei Han, Chen-Yu Lee, and Tomas Pfister. ReasoningBank: Scaling agent self-evolving with reasoning memory. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.25140v2.

Felipe Maia Polo, Ronald Xu, Lucas Weber, Mírian Silva, Onkar Bhardwaj, Leshem Choshen, Allysson Flavio Melo de Oliveira, Yuekai Sun, and Mikhail Yurochkin. Efficient multi-prompt evaluation of LLMs. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/28236482f64a72eec43706b6f3a6c511-Abstract-Conference.html.

Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with "gradient descent" and beam search. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7957–7968. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.494. URL https://aclanthology.org/2023.emnlp-main.494/.

Daniel Shepard and Robin Salimans. Automationbench. *arXiv preprint arXiv:2604.18934*, 2026. URL https://arxiv.org/abs/2604.18934.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Yifan Yang, Ziyang Gong, Weiquan Huang, Qihao Yang, Ziwei Zhou, Zisu Huang, Yan Li, Xuemei Gao, Qi Dai, Bei Liu, Kai Qiu, Yuqing Yang, Dongdong Chen, Xue Yang, and Chong Luo. SkillOpt: Executive strategy for self-evolving agent skills. *arXiv preprint arXiv:2605.23904*, 2026. URL https://arxiv.org/abs/2605.23904v2.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=roNSXZpUDN.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Pan Lu, Zhi Huang, Carlos Guestrin, and James Zou. Optimizing generative AI by backpropagating language model feedback. *Nature*, 639:609–616, 2025. doi: 10.1038/s41586-025-08661-4. URL https://www.nature.com/articles/s41586-025-08661-4.

Qizheng Zhang, Changran Hu, Shubhangi Upasani, Boyuan Ma, Fenglu Hong, Vamsidhar Kamanuru, Jay Rainton, Chen Wu, Mengmeng Ji, Hanchen Li, Urmish Thakker, James Zou, and Kunle Olukotun. Agentic context engineering: Evolving contexts for self-improving language models. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2510.04618v3.

Xiaolin Zhou, Jinbo Liu, Li Li, Ryan A. Rossi, and Xiyang Hu. Counterfactual trace auditing of LLM agent skills. *arXiv preprint arXiv:2605.11946*, 2026. URL https://arxiv.org/abs/2605.11946v2.

GUIDE TO THE APPENDICES

Table 4 maps each appendix to the main-text result it supports, so a reader who needs one protocol can locate it without reading the guide in full. Prediction and decision appendices come first because they carry the derivations, access rules, and reanalyses behind Sections 3 and 4.

Table 4: Appendix contents and the main-text result each supports. Appendices are grouped by the panel they document; the prediction panels, the natural-revision source panel, and the extraction panel are separate text collections and are not successive stages of one update trajectory.

Appendix	Contents	Supports
A Finite-record prediction	Risk identities, leave-one-out local risk, bias–variance comparison	Eq. (3); Sec. 4
B Local-observation budget	Numerical decomposition, intervals, portable reanalysis	Sec. 4; Fig. 2
C Deployment-gate replay	Fixed decision rule, absolute and relative value, conditional uncertainty	Sec. 3
D Question-category holdout	Post-hoc cross-question gate, fold values, refitted intervals	Sec. 3
F Proposal records	Queue construction, predictor access rules, contrasts, resampling	Sec. 3; Table 1
G Fully crossed replacements	Panel design, complete five-predictor risks, bootstrap scheme	Sec. 4; Fig. 2
H Five-arm content control	Arms, matching criteria, nested receiver-slot analysis	Sec. 5; Table 3
E Independent source queue	Endpoint and increment contrasts under the tag-cleaned interface	Sec. 2
I Natural-revision sources	Source bank, six configurations, selection procedure, scoring	Eq. (2)
Q Frozen natural repair	Ten frozen text objects, interfaces, comparison families	Sec. 5
J Outcome accounting	Source-outcome accounting, replacement scores, combination laws	Sec. 6
K Fixed-section generation	Five-section schema, designated *Answer Format* section, deployment	Sec. 6; Fig. 3
L Repeated execution	Historical versus fresh executions, estimation targets	Sec. 6
M Repeated-execution results	Complete condition means, contrasts, interface sensitivity	Sec. 6
N Scores and uncertainty	Every score interface and interval family used in the paper	All tables and figures
O, P	Answer changes, block variation, selected response examples	Sec. 6
R, S, T	Answer annotations, frozen instructions in multi-step automation, combination and placement results	Sec. 8

Appendix T interprets combination and placement results without assuming that a negative additive interaction means lost success. Appendix N is the reference for readers checking whether two numbers in different tables are directly comparable, since the panels use different score interfaces and interval families.

A FINITE-RECORD PREDICTION AND LOCAL EXECUTION NOISE

Fix a text pair, receiving document, question, and configuration. Let $X_1, \dots, X_R$ be the recorded score differences, $\bar{X} = R^{-1} \sum_r X_r$, and $s_X^2 = (R-1)^{-1} \sum_r (X_r - \bar{X})^2$, for $R > 1$. For a prediction z that is fixed across the R target executions,

$$\frac{1}{R} \sum_r (z - X_r)^2 = (z - \bar{X})^2 + \frac{1}{R} \sum_r (X_r - \bar{X})^2 = (z - \bar{X})^2 + \frac{R-1}{R} s_X^2.$$

The cross term vanishes because $\sum_r (X_r - \bar{X}) = 0$. Local history has a different prediction for each held-out execution:

$$z_{-r} = \frac{R\bar{X} - X_r}{R-1}, \qquad z_{-r} - X_r = \frac{R}{R-1}(\bar{X} - X_r).$$

Consequently

$$\frac{1}{R}\sum_r (z_{-r} - X_r)^2 = \frac{R}{R-1} s_X^2.$$

Both equalities are identities of the recorded table and need no independence assumption.

For an expectation-level interpretation, suppose the target executions are independent and identically distributed with mean θ and variance σ^2. Suppose also that the nonlocal predictor Z is independent of a target execution, conditional on this fixed deployment. Then

$$\mathbb{E}[(Z - X)^2] = \sigma^2 + (\mathbb{E}Z - \theta)^2 + \mathrm{Var}(Z), \qquad \mathbb{E}[(z_{-r} - X_r)^2] = \sigma^2 + \frac{\sigma^2}{R-1}.$$

Local averaging has lower expected risk exactly when

$$\frac{\sigma^2}{R-1} < (\mathbb{E}Z - \theta)^2 + \mathrm{Var}(Z).$$

This is the usual squared-error bias–variance comparison applied to the specified record access. It explains why a finite local mean may outperform or underperform a nonlocal predictor. It neither identifies the cause of a configuration difference nor proves that any nonlocal predictor is optimal. An optimal predictor allowed to ignore extra information has a different decision problem from the fixed averaging rules evaluated here. The reported bootstrap analyses reweight observed risks; they do not estimate a causal effect of thinking settings or implement a shrinkage policy.

B Numerical Decomposition of the Local-Observation Budget

This reanalysis uses the fully crossed panel in Appendix G, restricted to sixteen natural questions. Its portable array contains F1 differences with axes configuration, replacement (40), receiver (12), question (16), and repeat (4). The original score arrays are converted to float64 before subtraction. The target pair is excluded from the pooled predictor at every question. No new executions are collected.

For each target cell i, write $v_i = s^2_{X,i}$ and $d_i = (z_i - \bar{X}_i)^2$, and average equally over all $40 \times 12 \times 16$ cells. Then

$$V = \overline{v_i}, \quad D = \overline{d_i}, \quad \widehat{N}_3 = V/3, \quad \widehat{A} = D - V/4,$$
$$\widehat{\mathcal{R}}_{\mathrm{pool}} = D + 3V/4, \qquad \widehat{\mathcal{R}}_{\mathrm{local}} = 4V/3, \qquad \widehat{\mathcal{R}}_{\mathrm{pool}} - \widehat{\mathcal{R}}_{\mathrm{local}} = \widehat{A} - \widehat{N}_3.$$

These are exact finite-record identities. Under the independent-execution conditions in Appendix A, V estimates the mean target variance, $\widehat{N}_3$ estimates the three-observation local mean's variance, and $\widehat{A}$ estimates the mean of $(\mathbb{E}Z_i - \theta_i)^2 + \mathrm{Var}(Z_i)$. This last quantity includes nonlocal prediction variance and squared bias. It is not a direct measure of between-deployment heterogeneity.

The 5,000 bootstrap draws use the original seed and resampling order: sample ten replacement pairs with replacement within each of the four instruction families, twelve receiver slots, and four natural questions within each of the four categories. Independently drawn multiplicities are crossed to reweight the fixed cell statistics. Predictors are not refit and target repetitions are not resampled. This construction reproduces the original pooled-minus-local intervals while also giving intervals for the components.

In the high configurations, the $\widehat{A}$ intervals span zero. A negative moment estimate or interval endpoint represents finite-sample uncertainty, not a negative latent squared error. It is retained rather than truncated or interpreted as evidence of negative noise. The point estimates show both a larger local-noise cost and a smaller pooled-to-target error than in the corresponding off configuration. This is an observed configuration comparison, not a causal estimate of the effect of thinking.

For n independent local observations of a fixed deployment, expected risk is $\sigma^2 + \sigma^2/n$, compared with $\sigma^2 + A$ for the nonlocal rule. Hence local prediction is better iff $\sigma^2/n < A$. The experiment

Table 5: Components of the budget comparison. Entries use squared F1-effect units. The moments $\widehat{A}$ and their intervals are untruncated. Intervals are individual descriptive 95% intervals.

Configuration	V	D	$\widehat{N}_3$	$\widehat{A}$
Flash, off	.01563	.02566	.00521 [.00148, .00993]	.02176 [.00614, .03971]
Pro, off	.00814	.02416	.00271 [.00127, .00452]	.02212 [.00855, .04010]
Flash, high	.03361	.00967	.01120 [.00137, .02736]	.00127 [−.00053, .00474]
Pro, high	.01747	.00464	.00582 [.00067, .01451]	.00027 [−.00096, .00199]

measures this balance for $n = 3$; it does not validate extrapolated trial counts, an adaptive acquisition policy, or a cost-optimal gate. In particular, dividing by the small, uncertain high-configuration estimates of A would produce unstable thresholds.

Portable reproduction and verification. The source package includes the portable budget analysis, its paired-score inputs, and the reference results; the artifact README gives the reproduction commands. The reduced input preserves anonymous axis identities, instruction-family and batch/carrier structure, natural-question indices, and reference statistics; the artifact manifest hashes the supplied inputs. Default execution reads only the adjacent portable input. All five original F1 risks in all four configurations are reproduced within 7×10^{-18}, and the original pooled-minus-local interval endpoints within 7×10^{-17}. Tests cover the finite identities, local leave-one-out risk, target-pair exclusion, resampling weights, and archived-reference reproduction. Re-running from a separate directory produces byte-identical result JSON.

C A Retrospective Deployment-Gate Replay

This reanalysis asks what the matched information contrast changes in a specified reuse decision. It uses the natural queue in Appendix F: twelve source blocks, 144 proposals, 432 update–receiver placements, sixteen known questions, and three paired executions per cell. There are no new model calls. The zero threshold, deployment-level primary unit, and question-conditional secondary analysis were fixed for this retrospective analysis before inspecting their results; they were not endpoints of the original collection.

C.1 Decision rule, comparator, and evaluation

Let $i = (u, p)$ identify a placement and $z_{it}^{(k)}$ its effect prediction from history rule k. The primary gate is

$$a_i^{(k)} = \mathbf{1}\left\{\frac{1}{16}\sum_{t=1}^{16} z_{it}^{(k)} > 0\right\}, \qquad \widehat{J}_k = \frac{1}{432}\sum_i a_i^{(k)}\left(\frac{1}{48}\sum_{t=1}^{16}\sum_{r=1}^{3} X_{itr}\right). \tag{6}$$

Thus the gate makes one decision per placement, using the original sixteen-question panel. A rejection retains the unchanged receiver and has incremental value zero. All three paired target repetitions evaluate the fixed decision; none enters a nonlocal prediction. Reported values multiply $\widehat{J}$ by 100 to obtain F1 or EM points. F1 is the primary utility measure. EM uses its own predictions and threshold, so it evaluates a separate score-specific gate rather than EM outcomes under the F1 gate.

For the matched comparator, select one other proposal at each of the target proposal's two other receivers. Keep those donor identities fixed across all sixteen questions. For each admissible donor pair, average its predictions over questions, apply the gate, and evaluate its outcome. Average these values over all admissible pairs, then over placements. The expected acceptance indicator may therefore be fractional. Averaging all donor predictions before thresholding would give a different policy with a larger information budget.

Receiver-only, additive, and pooled retain their original target-deleted constructions. Always accept reports the collection's mean effect; always reject reports zero. The contextual predictors read different histories and budgets from the matched pair.

Table 6: Primary deployment gates on the fixed sixteen-question panel. Values are points relative to the unchanged receiver, with individual descriptive 95% intervals. Acceptance rates concern the score-specific gate.

Policy	F1 accept (%)	EM accept (%)	F1 value	EM value
Same proposal	24.537	34.722	-0.591 $[-1.425, 0.042]$	-0.231 $[-1.977, 1.582]$
Matched other proposals	24.003	32.131	-0.914 $[-1.894, -0.144]$	-0.711 $[-2.148, 0.797]$
Receiver only	22.917	34.259	$+0.202$ $[-0.631, 1.124]$	$+0.275$ $[-1.177, 1.881]$
Pooled	0.000	5.556	0	-0.183 $[-0.955, 0.068]$
Additive	34.491	43.056	-0.317 $[-1.204, 0.509]$	$+0.140$ $[-1.683, 2.181]$
Always accept	100.000	100.000	-2.656 $[-5.607, -0.180]$	-1.244 $[-5.397, 3.420]$
Always reject	0.000	0.000	0	0

The paired same-minus-matched value contrast is $+0.32365$ F1 points ($[0.01291, 0.71697]$) and $+0.47957$ EM points ($[-0.19747, 1.29812]$). This fixed gate translates the matched information advantage into a relative F1-value improvement. Its same-proposal value remains below zero at the point estimate, with an interval spanning zero. The replay therefore supports the relative information contrast without establishing a profitable deployment rule.

C.2 Conditional uncertainty and numerical arithmetic

We reproduce the original 2,000 source-block/question bootstrap draws. Whole source blocks are resampled, and the common questions are resampled within four categories. Predictions and actions are computed once on the original sixteen-question panel and held fixed. Bootstrap draws reweight the evaluation outcomes in Eq. (6); they do not refit a gate or simulate decisions on newly sampled questions. Intervals are individual descriptive percentile intervals, conditional on this fixed rule and decision panel. Source blocks and questions supply the resampling support, not the 20,736 raw differences.

The gate tests a strictly positive mathematical sum and rejects ties. Implementation sums original donor execution differences with stable summation before dividing by the positive count. This avoids treating exact-zero sums as positive because separately rounded means leave a residual near 10^{-18}. No epsilon changes the primary threshold. The original mean-of-rounded-means calculation and $\pm 10^{-12}$ perturbations are retained as numerical diagnostics in the result file. An independent exact-rational check of 7,392 primary same, receiver, pooled, and matched donor-pair signs agrees with the stable implementation. Additive primary predictions have no near-zero ambiguity.

Three noisy target repetitions do not provide a noise-free label of whether an update is beneficial. We consequently evaluate action-weighted held-out effects, rather than call observed sign disagreements true false-accept or false-reject rates. Prospective evaluation on independently collected deployments would test the rule beyond this replay.

C.3 Secondary question-conditional rule

The secondary rule applies $a_{it}^{(k)} = \mathbf{1}\{z_{it}^{(k)} > 0\}$ separately to each known question. This changes the deployment decision by allowing question-specific installation. The matched comparator still thresholds each admissible donor pair before averaging outcomes.

Table 7: Secondary question-conditional gate values. Units and conditional intervals follow Table 6; this rule is distinct from the primary placement-level gate.

Policy or paired contrast	F1 value	EM value
Same proposal	-0.122 $[-1.052, 0.699]$	$+0.897$ $[-0.878, 3.502]$
Matched other proposals	-0.440 $[-1.280, 0.266]$	$+0.323$ $[-1.159, 2.488]$
Receiver only	$+1.173$ $[0.248, 2.207]$	$+2.546$ $[0.256, 5.768]$
Pooled	$+0.112$ $[-0.666, 0.934]$	$+1.538$ $[-0.685, 5.237]$
Additive	$+0.715$ $[-0.298, 1.945]$	$+2.315$ $[-0.068, 5.787]$
Same − matched	$+0.318$ $[-0.030, 0.891]$	$+0.574$ $[-0.009, 1.494]$

Always-accept and always-reject values are unchanged. The secondary additive fits contain near-zero solver residuals that affect acceptance under tiny threshold perturbations; the positive perturbation diagnostic leaves their aggregate value unchanged. Full acceptance rates, contextual contrasts, and numerical diagnostics accompany the script output.

Portable reproduction. The source package includes the portable gate analysis, its paired-score inputs, and the reference results; the artifact README gives the reproduction commands. The reduced input carries paired score differences, all donor indices, anonymous source and question identities, signed predictions, and reference statistics. Running the script on the supplied input reconstructs the signed predictions and reproduces the gate, original six-predictor F1/EM risks, and their original risk intervals. The maximum discrepancy from archived risks and intervals is below 5×10^{-16}.

D Post-Hoc Question-Category Holdout

The main matched risk in Section 3 predicts an effect on a *known* target question from records for that same question at other text combinations. The placement-level gate in Appendix C also decides and evaluates on the original sixteen-question panel. This internal stress test changes the target: it asks whether records on three question categories support a deployment decision on the fourth category. It makes no new model calls and is not an independent-question validation.

Fold-specific decisions. For each of the four categories, withhold its four questions. Recompute each placement's same-proposal mean prediction from the other twelve questions and accept only if the signed mean is strictly positive. For the matched comparison, keep each admissible donor pair at the same two receivers fixed across the twelve training questions, threshold its prediction separately, then average realized values over donor pairs. All target-pair runs remain excluded from both nonlocal predictors. Evaluate each decision on the four withheld questions, averaging their three paired target executions. Rejection contributes zero relative to the unchanged receiver. F1 and EM define separate gates, as in Appendix C.

Table 8: Post-hoc leave-one-question-category-out gate replay on the original sixteen-question bank. Entries are same-proposal minus matched-other-proposal decision value in score points on the held-out category. Each fold trains its action on the other twelve questions.

Held-out category	F1 value difference	EM value difference
Evidence acquisition	-0.172	$+0.314$
Multi-hop reasoning	-0.141	$+0.199$
Source comparison	-0.101	-0.363
Verification	$+0.239$	$+0.975$
Equal-category average	-0.044	$+0.281$

Uncertainty and interpretation. The mean same-minus-matched F1 difference is -0.0438 points, with a descriptive paired 95% interval $[-0.5504, 0.5506]$; the EM difference is $+0.2813$ $[-0.4484, 0.8900]$. For F1, the same-proposal and matched policies have absolute values -0.9171

and -0.8733 points, respectively, against zero for always rejecting. Intervals come from 2,000 draws that resample twelve whole source blocks and questions within each of the four fixed categories, recomputing the fold-specific gates. They do not resample new categories, receiver banks, or proposals. The analysis was chosen after the original results and has only four held-out categories. Its imprecise cross-question decision contrast neither negates the original same-question prediction-risk result nor establishes transfer to unseen questions.

The artifact includes the portable holdout analysis, reduced input hash, numerical checks, fold-level values, and full interval outputs. The artifact README gives the reproduction commands. Stable summation of stored donor differences enforces the original strict-positive threshold without a tuned tolerance.

E An Independent Source Queue

The independent DeepSeek queue applies the endpoint decomposition in Appendix I to twelve historical parent skills. Each source has two branches. The common branch retains the historical skill. The adapted branch first performs one model-specific optimization step and keeps its changed candidate only if it improves on a separate adaptation-development set; otherwise it retains the historical skill. Each branch then produces a fresh main update, giving 24 main events paired into twelve source blocks. An adapted and a common branch from the same source remain members of the same block.

The adaptation stage uses 16 training and 32 development questions per source. The main stage uses 32 training and 32 development questions per event, followed by a shared 128-question source-evaluation set. Question identities are disjoint across these roles. Candidate generation uses failure reflection, a maximum of four edits, and the one-, two-, and four-edit prefixes. Main-development F1 selects the candidate; strict improvement over the current parent is required for acceptance. Fifteen main updates are accepted, spanning ten of the twelve sources.

This study removes residual opening and closing answer tags after the original answer extractor, repeatedly until the extracted string no longer changes. It uses the resulting answer for F1 throughout training feedback, selection, and evaluation. This tag-cleaned interface does not canonicalize yes/no polarity. On the reported source panel, the original and tag-cleaned interfaces give identical decomposition estimates.

Table 9: Endpoint decomposition in the independent DeepSeek source queue, using tag-cleaned F1. Differences are in F1 points with descriptive 95% source-block/question bootstrap intervals. "Adapted" identifies the branch with a preceding model-specific optimization step. The accepted rows condition on the separate development decision.

Support	Events	Sources	Parent − none	Retained − parent	Retained − none
All events	24	12	$+4.21$ $[0.65, 8.02]$	-0.22 $[-1.27, 0.77]$	$+3.99$ $[0.64, 7.67]$
Accepted	15	10	$+4.52$ $[1.04, 8.29]$	-0.36 $[-1.98, 1.22]$	$+4.16$ $[0.76, 7.84]$
Common, accepted	7	7	$+4.78$ $[1.04, 8.84]$	-0.45 $[-2.55, 1.70]$	$+4.33$ $[1.04, 7.94]$
Adapted, accepted	8	8	$+4.29$ $[0.75, 8.12]$	-0.28 $[-2.64, 1.87]$	$+4.02$ $[0.25, 8.02]$

The decomposition uses 20,000 shared bootstrap draws of the twelve original sources crossed with question draws stratified by the four task categories. Both branches of a sampled source move together. Accepted-subset estimates renormalize weights over accepted events in the sampled blocks. The 128 questions were collected to evaluate the source premise for later representation and receiver comparisons; those later comparisons required an established positive source increment. Table 9 describes the completed source evaluation. Across its all-event and accepted supports, the point estimates assign the endpoint advantage to the parent, with little additional contribution from the current update.

F Predicting a Proposal's Effects from Execution Records

F.1 Proposal collection and receiving documents

A historical DeepSeek Flash configuration with thinking disabled generates twelve proposals from each of twelve source blocks. Each proposal uses eight optimizer-training and four optimizer-development questions; held-out evaluation uses sixteen common questions, four from each task category. All 144 proposals enter the record analysis, including the 143 rejected by development selection.

Each proposal is installed in three receivers other than its source. The receiver assignment varies across proposals: within each source block, twelve proposals form 36 observed proposal–receiver pairs involving eleven receiver documents. Each receiver occurs in three or four pairs. Thus the design samples a sparse set of pairs from a fixed bank of documents. It does not cross every proposal with the same three receivers.

Installation places the same proposal text into a delimited update field while preserving the receiving document outside that field. The receiver bank has no pre-existing update field, so installation appends the literal proposal content. The resulting text is fixed across the three executions of each proposal–receiver–question combination. Each own-update execution is compared with a fresh execution of the unchanged receiver on the same question, producing 20,736 paired score differences.

F.2 Prediction target and information available to each predictor

Let u index a proposal, p a receiver, t a held-out question, and $r \in \{1, 2, 3\}$ an execution. Write

$$x_{uptr} = s(Y^u_{uptr}) - s(Y^0_{uptr}), \qquad \bar{x}_{upt} = \frac{1}{3}\sum_{r=1}^{3} x_{uptr}, \tag{7}$$

where Y^u and Y^0 are responses with and without the installed proposal, respectively. Scores s are token F1 or exact match on the $[0, 1]$ scale. A record is the three-execution mean $\bar{x}_{upt}$. All predictors use records for the same question t as the target. The experiment therefore asks how observations at other text combinations predict an effect on a known question.

For a prediction z formed without the target pair's repetitions, the finite-record risk at that pair and question is

$$\widehat{R}_{upt}(z) = \frac{1}{3}\sum_{r=1}^{3}(z - x_{uptr})^2 = (z - \bar{x}_{upt})^2 + \frac{2}{3}\widehat{\sigma}^2_{upt}, \tag{8}$$

where $\widehat{\sigma}^2_{upt} = \frac{1}{2}\sum_r (x_{uptr} - \bar{x}_{upt})^2$. The equality is an empirical identity. Its last term is common to the record predictors evaluated on the same target; their risk differences also compare prediction of the target's three-execution mean. Reported risks average equally over the observed pairs and questions. Their units are squared F1 or squared EM differences, rather than score percentage points.

The six predictors differ in the records they read. *Same proposal* averages the target proposal's records at its two other receivers. *Matched other proposals* instead chooses one record from another proposal at each of those exact same two receivers. We enumerate every admissible pair of choices, compute its two-record prediction and risk, and then average the risks. This matches receiver identities and the two-record budget; averaging the candidate predictions first would define a different predictor.

Receiver only averages the two or three records from other proposals at the target receiver. *Additive* fits proposal and receiver effects to the other observed pairs within the same source block and question. It deletes all three repetitions of the target pair before fitting and predicts their mean from the remaining pairs. The sparse design has 36 pair cells and 22 identifiable regression coefficients; deleting any target cell leaves the design full rank. *Pooled* averages the other 35 pair records in that block and question.

Local history uses the target combination's own repetitions. It predicts repetition r by $\frac{1}{2}\sum_{j\neq r} x_{uptj}$ and averages squared error over the three held-out repetitions. Its prediction varies with r, so the fixed-z decomposition in Eq. (8) does not directly apply to this predictor. Unlike the other predictors, it has already observed the exact proposal, receiver, and question together. Its role is to measure the

value of local execution history. Implementing that access on a previously unobserved combination would require local trial executions.

Table 10: Finite-record prediction risks in the natural proposal queue. F1 and EM scores lie in $[0, 1]$; the table reports squared score-difference risk. A record contains three executions. Local history instead reads the other two individual repetitions of the target combination.

Predictor	Available records	F1 risk	EM risk
Same proposal at other receivers	2	0.092998	0.115901
Matched other proposals	2	0.100968	0.131498
Receiver only	2 or 3	0.068065	0.084309
Additive, target pair deleted	35	0.099505	0.117547
Pooled, target pair deleted	35	0.068875	0.088412
Local history	2 executions	0.027337	0.028356

Table 11: Paired risk differences for the natural proposal queue. A positive value favors the predictor named on the right. Intervals are descriptive 95% source-block/question bootstrap intervals; they describe prediction error, not improvement in deployed task scores.

Risk difference	F1	EM
Matched other − same proposal	0.007970 $[0.001313, 0.017974]$	0.015596 $[0.006482, 0.027043]$
Same proposal − local history	0.065661 $[0.016183, 0.128212]$	0.087545 $[0.027610, 0.152508]$
Receiver only − local history	0.040727 $[0.013612, 0.073479]$	0.055952 $[0.025125, 0.089631]$
Additive − local history	0.072167 $[0.021404, 0.131479]$	0.089191 $[0.031449, 0.151610]$

F.3 Resampling and interpretation

The record analysis uses 2,000 retrospective paired bootstrap draws. Each draw resamples the twelve whole source blocks and the common questions within their four strata. It retains every proposal, receiver assignment, and repetition within a sampled block. Predictions and cell risks are computed once; the bootstrap reweights these fixed risks. It neither refits the additive predictor nor places a repeated target cell into its training set.

Tables 10 and 11 separate two observations. With receivers and record budgets matched, proposal identity improves prediction of both F1 and EM effects. The target combination's own history provides further predictive information, including over receiver-only and additive baselines. The receiver-only and pooled predictors nevertheless attain lower absolute risk than the same-proposal predictor. The matched contrast identifies information carried by a proposal across its observed receivers, while the complete table locates that information among the other available records.

G Fully Crossed Text Replacements

The controlled panel varies replacement text, receiver document, and execution configuration. It contains forty paired replacement texts from four instruction families, with two generation batches and five carrier roots per family. Each pair is installed in a designated section of each of twelve receiver documents, leaving the other sections fixed. The replacements are not derived from these receivers. Both alternatives contain substantive instructions: if they are denoted g_0 and g_1, their score difference is an alternative-text contrast, rather than a comparison with an unmodified receiver.

Every replacement–receiver combination is evaluated on twenty common questions with four executions under each alternative. Four questions are directed diagnostic probes; the prediction analysis below uses the remaining sixteen natural questions, four per task category. The full collection comprises 76,800 responses per configuration. The four configurations are DeepSeek V4 Flash and V4 Pro, each with thinking disabled or set to high. They are analyzed as execution configurations.

For the score difference $x_{uptr} = s(g_1) - s(g_0)$, the target and risk follow the same construction as Equations (7)–(8), now with four repetitions. The common within-cell term is $\frac{3}{4}\widehat{\sigma}^2$. Pooled prediction averages the other 479 replacement–receiver records at the same question; replacement-only prediction reads the same replacement at the other eleven receivers; receiver-only prediction reads the other 39 replacements at the target receiver. Additive prediction fits replacement and receiver effects with the complete target cell deleted. Local history averages the other three repetitions of the same replacement–receiver–question combination to predict the fourth.

Table 12: Complete F1 prediction risks in the fully crossed panel, restricted to its sixteen natural questions. All five predictors use squared prediction error for effects computed from F1 scores in $[0, 1]$. Each record outside the target cell averages four executions.

Configuration	Pooled	Replacement only	Receiver only	Additive	Local history
Flash, thinking off	0.03739	0.03934	0.03804	0.04010	0.02084
Pro, thinking off	0.03026	0.03246	0.03056	0.03281	0.01085
Flash, thinking high	0.03488	0.03593	0.03513	0.03624	0.04481
Pro, thinking high	0.01775	0.01800	0.01784	0.01810	0.02329

Table 13: Risk reductions from using local history in the fully crossed panel. Positive values favor local history. Intervals are descriptive 95% crossed bootstrap intervals over replacements, receivers, and natural questions.

Configuration	Pooled − local	Receiver only − local	Additive − local
Flash, thinking off	0.01655 [0.00385, 0.03204]	0.01721 [0.00408, 0.03320]	0.01926 [0.00500, 0.03621]
Pro, thinking off	0.01941 [0.00600, 0.03765]	0.01971 [0.00613, 0.03841]	0.02197 [0.00732, 0.04227]
Flash, thinking high	−0.00993 [−0.02403, −0.00116]	−0.00968 [−0.02359, −0.00112]	−0.00857 [−0.02111, −0.00096]
Pro, thinking high	−0.00555 [−0.01434, −0.00061]	−0.00546 [−0.01427, −0.00059]	−0.00519 [−0.01374, −0.00053]

The 5,000 bootstrap draws resample ten replacement pairs within each of the four instruction families, twelve receiver documents, and four questions within each natural-question category. The predictions remain fixed while these weights are applied to the paired cell risks. Local history has lower risk than the other tested predictors in both thinking-off configurations. In both high-thinking configurations, its three local observations have higher risk than the pooled records. This reversal locates the value of a finite local history within the observed execution configuration; it does not attribute the change to an isolated reasoning mechanism.

H Separating Update Content, Added Text, and Source Correspondence

A separate five-arm experiment examines what an update contributes beyond occupying space in a receiving document. Twenty-eight newly generated source lineages each produce two proposals, yielding 56 updates. Source generation uses a shared pool of reflection tasks and retains the first schema-valid, distinct document for each lineage. Proposal generation uses separate optimizer-training and development tasks. Nine proposals are accepted by development selection and 47 are rejected; all 56 enter this content experiment. The executor is historical DeepSeek Flash with thinking disabled.

Each proposal is placed in three receivers drawn from the same 28-document bank, none of which is the proposal's own source. The 168 placements reuse each of the 28 documents six times as a receiver, and a document that served as one proposal's source can serve as another's receiver; the two roles are distinguished by the placement, not by the document. All five arms share the text outside one delimited update region. The unchanged-receiver arm removes that region; the blank

arm retains its empty delimiters; the inert arm fills it with unrelated natural prose; the shuffled arm inserts another source event's update; and the source-update arm inserts the current event's update. The inert control matches character and whitespace-token counts within 5%, and matches paragraph and Markdown-heading counts exactly. Its prose is screened for task guidance and overlap with the update's instructional content. The shuffled donor is fixed by the source assignment before target evaluation. The source update was generated for its own source document, while all three receivers are different documents.

Every event–receiver–arm combination is evaluated on sixteen held-out questions, four per task category, with three executions. This produces 40,320 responses. Table 14 gives all six planned paired contrasts, including both the content comparison and the net change from the unchanged receiver.

Table 14: Five-arm content controls. Differences are F1 points with individual 95% crossed bootstrap intervals, without a familywise adjustment. "Source update" refers to the proposal associated with the current source event; it is evaluated at receivers other than its own source. "Unchanged receiver" is the arm with the update region removed.

Comparison	F1 difference	95% interval
Source update − matched inert text	$+3.610$	$[0.047, 8.134]$
Shuffled update − matched inert text	$+3.393$	$[0.071, 7.634]$
Source update − shuffled update	$+0.217$	$[-1.475, 2.349]$
Source update − unchanged receiver	$+2.117$	$[-0.831, 6.355]$
Matched inert text − blank region	-0.720	$[-2.485, 0.811]$
Blank region − unchanged receiver	-0.772	$[-2.061, 0.553]$

The analysis first averages the three repetitions within each event–receiver–question cell. Each of 2,000 bootstrap draws samples the 28 source lineages, the two events within each sampled lineage, and the three receiver slots within each sampled event. The common questions are crossed with those draws and resampled within their four categories. These intervals follow the planned nested receiver-slot analysis on the fixed 28-document bank; repeated receiver identities and shuffled-donor links are not separately resampled as a crossed population of documents.

Both source and shuffled updates outperform matched inert prose under their individual 95% comparisons. The source-correspondence contrast and the source-update increment over the unchanged receiver have positive estimates, with both intervals spanning zero. The controls therefore distinguish the observed advantage over unrelated filler from the two stronger possibilities: an advantage specific to the originating event, and a net improvement over simply retaining the unchanged receiver. A directly written task-rule condition would address a further question—whether the feedback source contributes beyond an equally explicit task instruction—and is a different comparison from these five arms.

I Natural-Revision Protocol and Source Comparisons

Table 15: Update increments and final deployment scores have different reference points. Each configuration contains twelve natural events from six shared source states. Entries are F1 differences in points; both interfaces score the same responses. Full paired intervals appear in Appendix I.

	Update increment $F - P$		Endpoint advantage $F - N$	
Configuration	Original	Polarity	Original	Polarity
Executor A	+4.92	+4.72	-11.11	-12.78
DeepSeek Flash	+0.10	+0.05	+2.98	+1.59
Qwen 397B	+0.02	+0.02	-0.17	-0.17
GLM-5.3	+0.74	+0.09	+4.26	+0.77
GLM-5.3-Flash	+0.25	+0.25	+0.88	+0.88
Executor B	+1.55	-0.38	+10.52	-1.07

I.1 Sources, tasks, and execution configurations

The natural panel uses six source states: five historical skills from an earlier Executor A development study and an empty skill. Each source receives two generation events per configuration. The twelve events have sixteen training questions and a disjoint sixteen-question development panel each. The common held-out panel has 128 questions, thirty-two in each of four strata: verification, comparison, shared attributes, and multi-hop relations. Training, development, and test roles are disjoint. Executor A has a larger archived test panel; only the common 128-question subset enters Table 15. The source bank was selected before these tests and is not a random sample of possible parent skills.

The frozen task constructor supplies supporting documents first, followed by distractors. Execution requires no retrieval or external tools. Each of the six configurations generates updates from its own training responses and executes its own candidates. The configurations share source states and question assignments while generating their own edits. DeepSeek V4 Flash runs with thinking disabled and a 4,096-token output limit; the other five configurations have a 16,384-token limit. The GLM configurations are two variants of one named family; Executor A and Executor B denote two archived configurations whose public checkpoint identities were not established. The six configurations are not six independently controlled model families.

Table 16: Candidate coverage in the 72-event natural panel. Accepted candidates are a subset of the nonempty proposals; events without an accepted candidate retain their parent in the workflow comparison.

Configuration	Events	Nonempty proposals	Accepted
Executor A	12	12	10
DeepSeek V4 Flash	12	12	4
Qwen 397B	12	8	2
GLM-5.3	12	6	4
GLM-5.3-Flash	12	7	1
Executor B	12	8	3
Total	72	53	24

I.2 Generation, selection, and response aliases

The implementation uses the reflection, aggregation, clipping, and skill-compilation modules of SkillOpt (Yang et al., 2026). Training failures are identified by archived EM. The optimizer proposes at most four edits. Development evaluation uses prefix caps of one, two, and four, truncated to the available valid edits and deduplicated. Thus a three-edit proposal includes a three-edit candidate. The candidate with the highest development F1 is the literal candidate L. The retained skill is $F = L$ only for a strict improvement over the parent P; otherwise $F = P$. Events without an eligible candidate retain the parent. All twelve events per configuration enter the workflow comparison.

The compiler applies replacements to their literal targets, insertions at their anchors, and appends to the document's end. The natural table compares retained skills with parents and the empty no-skill condition N. Identical text objects share a recorded execution for a given question. There is one response per unique prompt–question pair; aliases such as $F = P$ are not independent repetitions. For any common score v, the endpoint decomposes exactly as

$$v(F) - v(N) = \{v(P) - v(N)\} + \{v(F) - v(P)\}. \tag{9}$$

The first term describes the starting document and the second the retained update's increment under the selection procedure.

I.3 Scoring and paired uncertainty

The archived parser reads the last complete `<answer>...</answer>` pair, falling back to the last nonempty line when no pair is complete. Token F1 uses lowercase, punctuation, article, and whitespace normalization and overlap precision and recall. EM compares normalized strings. Incomplete executions receive zero task score. Original F1 preserves each study's archived extraction interface.

The polarity sensitivity applies only to a question with a single reference whose stripped lowercase form is `yes` or `no`. It recognizes a literal leading yes/no, ignoring case and leading whitespace, followed by optional whitespace and either punctuation from `, . ; : ! ?` or the end of the answer. That answer is reduced to the lowercase binary label. For example, "Yes, they are" is canonicalized, while "Yes because..." and "true" are not. Other references are unaffected. This narrow rescoring uses the same responses. The natural and fixed-section tables report both interfaces. The repair study additionally distinguishes raw extraction, residual-tag cleanup, and its primary score after both tag and polarity cleanup.

Natural-study uncertainty uses 20,000 paired bootstrap draws. Each draw resamples the six source clusters, retaining the two events within each cluster, and resamples the 128 questions within their four strata. Source and question weights are shared across configurations; the configurations themselves are fixed. Table 17 reports 95% intervals for the three decomposition terms. These quantify variation over this source-and-question panel, with six source clusters rather than 72 independent sources.

Table 17: Source-deployment decomposition on the common 128 questions. All contrasts include twelve events per configuration; intervals are paired 95%. Original and polarity rows score the same responses.

Configuration	Score	$P - N$	$F - P$	$F - N$
Executor A	Original	-16.02 [-23.97, -8.19]	+4.92 [+0.43, +9.30]	-11.11 [-16.35, -6.54]
	Polarity	-17.50 [-25.91, -9.05]	+4.72 [+0.00, +9.24]	-12.78 [-18.23, -7.86]
DeepSeek Flash	Original	+2.88 [-0.32, +6.65]	+0.10 [-0.88, +1.10]	+2.98 [-0.08, +6.68]
	Polarity	+1.54 [-1.33, +4.86]	+0.05 [-0.94, +1.03]	+1.59 [-1.11, +4.81]
Qwen 397B	Original	-0.19 [-2.10, +1.45]	+0.02 [-0.27, +0.34]	-0.17 [-2.11, +1.51]
	Polarity	-0.19 [-2.10, +1.45]	+0.02 [-0.27, +0.34]	-0.17 [-2.11, +1.51]
GLM-5.3	Original	+3.52 [+0.84, +6.91]	+0.74 [-0.31, +2.60]	+4.26 [+1.30, +7.56]
	Polarity	+0.68 [-1.05, +2.83]	+0.09 [-0.43, +0.70]	+0.77 [-1.08, +3.00]
GLM-5.3-Flash	Original	+0.63 [-1.29, +2.88]	+0.25 [+0.00, +0.98]	+0.88 [-0.96, +3.11]
	Polarity	+0.63 [-1.29, +2.88]	+0.25 [+0.00, +0.98]	+0.88 [-0.96, +3.11]
Executor B	Original	+8.96 [+4.26, +13.88]	+1.55 [-0.47, +5.39]	+10.52 [+6.33, +14.50]
	Polarity	-0.69 [-3.57, +1.97]	-0.38 [-1.33, +0.09]	-1.07 [-4.08, +1.81]

J Score Preservation and Outcome Accounting

Mean scores summarize which outcomes occur under a deployed text. The identities below recover distinctions that this summary discards: which source corrections survive extraction, which source errors are repaired, and whether a combined instruction loses a success or repeats coverage of the same requirement.

J.1 Deployment laws and recorded outcome profiles

Let $\mathcal{X}$ be a finite set of evaluation units with weights $w_x \geq 0$ and $\sum_x w_x = 1$. A unit is a question, a generated-block–question pair, or an Automation task. The complete instruction text T and execution configuration c determine an outcome distribution at each unit. In QA, $p_T(a \mid x, c)$ is the distribution of normalized answers used in the main text; in Automation, the outcome includes the trajectory and terminal world state. A scoring function takes values in $[0, 1]$, and its expected weighted score is $V(T)$.

To describe the historical response tables, fix one recorded outcome $y_{c,T}(x)$ for each deployment and unit. Its profile is

$$\Phi_c(T) = \big(y_{c,T}(x)\big)_{x \in \mathcal{X}}. \tag{J.1}$$

Let $f_{c,T}(x) \in [0, 1]$ be its score and $z_{c,T}(x) \in \{0, 1\}$ its success indicator under a specified criterion. Suppressing fixed c, write

$$v_f(T) = \sum_x w_x f_T(x), \qquad v_z(T) = \sum_x w_x z_T(x). \tag{J.2}$$

Thus v_f denotes the score of a recorded profile, whereas V denotes an expectation over new execution. F1 is divided by 100 in this appendix; multiplying a difference by 100 recovers percentage points.

Binary success means exact match in the QA accounting and strict task success in Automation. Automation partial credit remains the task-level eligible-assertion fraction, averaged over tasks.

In the generated-section study, T represents the block-specific texts T_b, with $y_{c,T}(b,q) = y_{c,T_b}(q)$. The starting-prompt response for a question is shared across blocks. The resulting table has paired outcomes but not independent observations in every cell. All set identities below hold for this recorded table and, under a specified joint sampling scheme, after taking expectations. Marginal execution laws determine expected score contrasts; a joint outcome table is additionally needed to count successes that occur together. Repeated-execution inference is developed in Appendix J.5.

J.2 Extraction changes the deployed text

A fixed extraction rule e maps a full document F to a section $S = e(F)$. Deploying S replaces the input to the execution system and produces its own law p_S. Consequently, an extract can retain a source correction, lose it, or succeed where the source failed. In the insertion notation $D + A$, the operation specifies the receiving document, separator, and placement. These are part of the deployed text whose outcomes are evaluated.

J.3 Replacement scores and source corrections

For a recorded text T, define $C_T = \{x : z_T(x) = 1\}$ and $\mu(H) = \sum_{x \in H} w_x$. Compare the starting prompt G, full skill F, and extracted section S. The source's newly achieved successes are $C_F \setminus C_G$. Define

$$H = \mu\big((C_F \setminus C_G) \cap C_S\big), \qquad \text{source gains retained,} \tag{J.3}$$

$$L = \mu\big((C_F \setminus C_G) \setminus C_S\big), \qquad \text{source gains lost,} \tag{J.4}$$

$$R = \mu\big((C_G \setminus C_F) \cap C_S\big), \qquad \text{starting successes recovered.} \tag{J.5}$$

The retained fraction is $H/(H+L)$ when $H + L > 0$. Recovery R instead counts successes restored after a source error.

Proposition J.1 (Replacement accounting)**.** *For recorded deployments G, F, S with common evaluation weights and success criterion, let*

$$p_{ijk} = \mu\{x : (z_G(x), z_F(x), z_S(x)) = (i,j,k)\}, \qquad i,j,k \in \{0,1\}.$$

Then

$$v_z(S) - v_z(F) = p_{101} + p_{001} - p_{010} - p_{110}. \tag{J.6}$$

The positive terms count recovered starting successes and new successes absent from both G and F. The negative terms count lost source gains and lost successes shared by G and F. Retained source gains $H = p_{011}$ cancel from the difference.

Proof. The eight joint states partition $\mathcal{X}$, so the difference is $\sum_{i,j,k}(k-j)p_{ijk}$. States with $j = k$ contribute zero; the remaining four give Eq. J.6. Equivalently,

$$v_z(S) - v_z(F) = \mu(C_S \setminus C_F) - \mu(C_F \setminus C_S). \tag{J.7}$$

Partitioning each difference set by success under G gives the four terms, including $L = p_{010}$ and $R = p_{101}$. □

Proposition J.2 (High agreement with no source-gain retention)**.** *For every $0 < \varepsilon < 1/2$, there is a deterministic binary-output system in which F and S both improve on G by ε, both score $1 - \varepsilon$, and agree on their complete answers with probability $1 - 2\varepsilon$, yet S retains none of F's newly achieved successes.*

Proof. Use three units with weights $1 - 2\varepsilon, \varepsilon, \varepsilon$. Let a complete answer be a bit, with success defined as outputting one, and set

$$z_G = (1,0,0), \qquad z_F = (1,1,0), \qquad z_S = (1,0,1).$$

Then $v_z(G) = 1 - 2\varepsilon$ and $v_z(F) = v_z(S) = 1 - \varepsilon$. The first unit supplies agreement of mass $1 - 2\varepsilon$. The second is the only source gain, and S fails there, giving $H = 0$ and $L = \varepsilon$. □

As ε decreases, score and agreement approach one while retention remains zero. Setting $z_S = z_F$ instead gives complete retention with the same three mean scores. Thus a replacement margin constrains the net score difference without determining the fate of source-specific corrections.

The historical GLM table illustrates the accounting. It contains 99 retained source gains, 44 lost source gains, 41 recovered starting successes, 45 new section matches, and 34 lost shared matches. Hence

$$v_z(S) - v_z(F) = \frac{41 + 45 - 44 - 34}{12 \times 512} = \frac{8}{6144}.$$

These are recorded block–question counts with shared questions and starting responses (Appendix O). The repeated diagnostic comparisons use fresh source and section executions on the fixed historical gain and loss sets.

For a graded score, $[a]_+ = \max(a, 0)$ gives

$$v_f(S) - v_f(F) = \sum_x w_x [f_S(x) - f_F(x)]_+ - \sum_x w_x [f_F(x) - f_S(x)]_+. \tag{J.8}$$

Applied to the historical GLM full skill versus the starting prompt, the positive and negative contributions are 1.960 and 1.396 polarity-normalized F1 points, leaving 0.564 points. Exact-match and F1 accounting follow different score functions: eight additional exact matches can coexist with a slightly lower mean F1.

J.4 Combination, repeated coverage, and lost successes

Consider background D and the prescribed deployments $D + A$, $D + B$, and $D + B + A$, with fixed insertion order and placement. Let their recorded success sets be C_0, C_A, C_B, C_{AB}. Write $U = C_A \cup C_B$ and $I = C_A \cap C_B$. The binary-score interaction is

$$J_z = v_z(D + B + A) - v_z(D + A) - v_z(D + B) + v_z(D). \tag{J.9}$$

Proposition J.3 (Success-set decomposition of interaction)**.** *For these four recorded deployments,*

$$\begin{aligned} J_z = {} & \mu(C_{AB} \setminus U) - \mu(U \setminus C_{AB}) \\ & + \mu(C_0 \setminus I) - \mu(I \setminus C_0). \end{aligned} \tag{J.10}$$

Proof. The sum of two success indicators equals the sum of their union and intersection indicators. Therefore $J_z = \mu(C_{AB}) - \mu(U) + \mu(C_0) - \mu(I)$. Applying Eq. J.7 to each difference gives the result. □

The first pair compares the combination with the union of separate successes: it counts successes outside that union and successes lost from it. The second pair compares baseline success with success shared by both additions. In particular, $\mu(I \setminus C_0)$ counts new successes achieved by each separate addition, while $\mu(C_0 \setminus I)$ counts baseline successes missed by at least one addition, whether or not the combination recovers them.

These terms need not describe disjoint units. Pattern $(z_0, z_A, z_B, z_{AB}) = (0, 1, 1, 0)$ contributes -1 for repeated new coverage and -1 for its subsequent loss. Conversely, $(1, 0, 0, 0)$ has interaction $+1$ although every modified deployment fails. The sign of the interaction alone therefore does not determine whether composition preserves successes.

Corollary J.4 (Preserved successes can yield negative interaction)**.** *If $C_0 \subseteq I$ and $U \subseteq C_{AB}$, then*

$$J_z = \mu(C_{AB} \setminus U) - \mu(I \setminus C_0). \tag{J.11}$$

If also $C_{AB} = U$, interaction equals the negative mass of shared new successes.

Proof. The two inclusions remove the second and third terms of Eq. J.10. Equality $C_{AB} = U$ removes the first. □

The corollary describes negative interaction with every separate success intact. When the combination loses successes, $U \setminus C_{AB}$ records them directly. Its subset $I \setminus C_{AB}$ contains cases where both individual deployments succeed and the combination fails, the historical pattern inspected in Automation.

Repeated executions assess the corresponding success probabilities rather than treating one recorded pattern as a stable response type.

For graded scores, define the pointwise better and worse separate scores

$$u_f(x) = \max(f_A(x), f_B(x)), \qquad i_f(x) = \min(f_A(x), f_B(x)).$$

Since $f_A + f_B = u_f + i_f$,

$$J_f = \sum_x w_x(f_{AB}(x) - u_f(x)) + \sum_x w_x(f_0(x) - i_f(x)). \tag{J.12}$$

With $\mathbb{E}_w[a] = \sum_x w_x a(x)$, splitting each term by sign yields

$$\begin{aligned} J_f = \mathbb{E}_w[(f_{AB} - u_f)_+] - \mathbb{E}_w[(u_f - f_{AB})_+] \\ + \mathbb{E}_w[(f_0 - i_f)_+] - \mathbb{E}_w[(i_f - f_0)_+]. \end{aligned} \tag{J.13}$$

These are pointwise comparisons before averaging. For binary scores, they reduce to Proposition J.3.

Corollary J.5 (Repeated gains under graded scoring)**.** *If $f_0 \le \min(f_A, f_B)$ and $f_{AB} = \max(f_A, f_B)$ on every unit, then*

$$J_f = -\mathbb{E}_w[\min(f_A - f_0, f_B - f_0)]. \tag{J.14}$$

Proof. The first sum in Eq. J.12 is zero, and $i_f - f_0 = \min(f_A - f_0, f_B - f_0) \ge 0$. Substitution gives the identity. If instead $f_{AB} \ge u_f$, the nonnegative term $\mathbb{E}_w[f_{AB} - u_f]$ records gains beyond the better separate score. □

On the eighteen historical Executor A questions with EM pattern $(0, 1, 1, 1)$, each separate and combined F1 score is one, so each interaction is $-(1 - f_0)$. With weights $1/512$, their total contribution is -2.94 F1 points. The other questions contribute $+0.14$, giving the recorded -2.80 interaction. This calculation accounts for the baseline's partial credit and explains the selected pattern without assuming that the corollary's conditions hold on every question.

J.5 Repeat-calibrated discrepancy and execution uncertainty

Conditional answer distributions. Fix a task or block–question input x and execution configuration c. Let p_F and p_S be the distributions of its normalized answers under the full skill and section, suppressing x, c in this paragraph. Calls are assumed independent conditional on these inputs, with a common distribution across repetitions of each text. Define

$$B = \Pr(A_F \ne A_S), \qquad W_X = \Pr(A_X \ne A'_X), \quad X \in \{F, S\},$$

where every pair uses independent calls. Their calibrated difference is

$$E = B - \frac{W_F + W_S}{2} = \frac{1}{2}\sum_a \big(p_F(a) - p_S(a)\big)^2. \tag{J.15}$$

Indeed, $B = 1 - \sum_a p_F(a)p_S(a)$ and $W_X = 1 - \sum_a p_X(a)^2$; substitution gives the square. This is half the squared maximum mean discrepancy for the discrete delta kernel (Gretton et al., 2012). It equals zero precisely when the two conditional normalized-answer laws agree. Averaging E_x with positive weights preserves this characterization on the evaluated inputs. The comparison depends on the answer-normalization interface; changed strings need not represent changed factual judgments.

Unbiased estimation. For $R \ge 2$ fresh repetitions of each text, use all cross-text pairs and all distinct within-text pairs:

$$\widehat{B} = \frac{1}{R^2}\sum_{r=1}^{R}\sum_{s=1}^{R} \mathbf{1}\{A_{F,r} \ne A_{S,s}\}, \tag{J.16}$$

$$\widehat{W}_X = \frac{1}{R(R-1)}\sum_{r \ne s} \mathbf{1}\{A_{X,r} \ne A_{X,s}\}, \tag{J.17}$$

$$\widehat{E} = \widehat{B} - \frac{\widehat{W}_F + \widehat{W}_S}{2}. \tag{J.18}$$

Each summand has the stated expectation, so linearity gives unbiasedness even though pairs share outputs. The pairs are not independent observations for inference. Finite estimates can be negative and remain untruncated. Applying the same construction to binary match indicators gives $E = (q_F - q_S)^2$, where q_X is the exact-match probability.

Conditional distribution tests. The QA statistic averages $\widehat{E}_x$ over the weighted block–question inputs. The permutation test exchanges text labels within repeat-indexed pairs at each input and recomputes this statistic. Under the null of equal conditional answer distributions at every input, independent calls are exchangeable within these pairs. The two configuration tests form one Holm family. A repetition index groups calls for this permutation; it does not couple their execution randomness or define a shared latent response type across texts.

For Automation placement, we compute discrepancy for each fixed assertion's passed flag, average assertions within a task, and then weight tasks equally. Label exchanges move whole trajectories, retaining the dependence among a task's assertions. The permutation null is equality of the conditional assertion-vector laws, and the two configurations form a separate Holm family. The statistic measures marginal assertion discrepancies: unequal joint laws with identical marginals can have zero discrepancy. The primary analysis uses every recorded passed flag; a sensitivity analysis distinguishes eligible failure, eligible success, and exclusion.

Execution variance of score contrasts. Let u index distinct text–task–configuration inputs. Assume all new calls are independent, and let $Y_{u,1}, \ldots, Y_{u,R_u} \in [0,1]$ be identically distributed scores within each input, with $R_u \geq 2$. Merge aliases of the same input before writing a contrast as $\widehat{\theta} = \sum_u c_u \overline{Y}_u$. Conditional on the fixed inputs,

$$\mathrm{Var}(\widehat{\theta}) = \sum_u \frac{c_u^2 \sigma_u^2}{R_u}, \qquad \widehat{\mathrm{Var}}(\widehat{\theta}) = \sum_u \frac{c_u^2 s_u^2}{R_u}, \tag{J.19}$$

where s_u^2 is the ordinary unbiased sample variance. Independence removes cross-input covariance terms. A starting response shared across twelve generation blocks receives its combined coefficient before squaring; its aliases do not supply independent executions. Scores, contrasts, and standard errors are multiplied by 100 for percentage-point reporting.

These execution standard errors condition on the selected inputs. The stratified-question and crossed-block bootstrap instead resamples their per-input means to assess variation over questions and generation blocks; it adds no repetition bootstrap. Four identical outcomes give zero sample variance but do not establish a deterministic execution law.

Bounds on frozen discovery sets. Historical source-gain and source-loss sets are fixed before new execution. Conditional on their membership and texts, fresh calls estimate $\theta = \mathbb{E}[\widehat{\theta}]$ for those inputs. Define $\mathcal{V} = \sum_u c_u^2 / R_u$. Each independent summand $c_u Y_{u,r} / R_u$ has range length $|c_u|/R_u$, so for $\mathcal{V} > 0$, Hoeffding's inequality gives

$$\Pr\left(\left|\widehat{\theta} - \theta\right| \geq t\right) \leq 2\exp\left(-\frac{2t^2}{\mathcal{V}}\right). \tag{J.20}$$

A conservative $1 - \alpha$ interval is therefore

$$\widehat{\theta} \pm \sqrt{\frac{\mathcal{V}}{2} \log \frac{2}{\alpha}}, \tag{J.21}$$

intersected with the contrast's possible score range. The reported individual 95% bounds use $\alpha = 0.05$. They cover new-execution uncertainty on the frozen set without estimating variance from four runs. Question-and-block intervals address variation across tasks and texts, so a fixed-set bound can exclude zero while the corresponding resampling interval includes it.

K Fixed-Section Generation and Deployment

The study combines a broad single-execution panel with repeated execution on an input-selected panel and fixed diagnostic sets. This section gives the broad panel and its text constructions; Appendix L specifies repeated sampling and estimation.

K.1 Generation and construction of deployment objects

The study uses twelve disjoint groups of eight training questions. For each group, DeepSeek V4 Flash generates a feedback skill from execution experience and a generic skill from a general QA proposer instruction and schema without those execution trajectories. The first schema-valid candidate is retained within the allowed attempts, without a utility screen. All twelve paired blocks are valid. Generation uses temperature 0.7, thinking disabled, and a 4,096-token output limit. The common root G is:

> # Question Answering Skill
> (No learned rules yet. Rules will be added through the reflection process.)

The five sections are Evidence Acquisition, Source Comparison, Multi-Hop Reasoning, Answer Format, and Verification. This shared schema permits a fixed extraction boundary across generations. The Answer Format section was designated before test outcomes. It occupies 18.4% of the feedback document's characters on average.

For each block, F denotes the full feedback skill, S its literal Answer Format section with marker and heading, and R its remainder. The generic document is C, and C^- removes its own corresponding section. Both $C^- + S$ and $C + S$ append the same donor with the same separator. They involve no receiver search or text rewriting. Removing a named section does not remove semantically related instructions elsewhere in the document. Generic generation is matched on root, general QA framing, and schema, without exact wording or length matching.

The study also evaluates the feedback title and headings, a common title alone, G, and no skill N. With shared controls deduplicated, the set contains 99 unique text objects. Each is executed once on each of 512 test questions by both DeepSeek V4 Flash and GLM-5.3-Flash. These questions are disjoint from the generation groups and identical to the repair study's test panel. The studies therefore have separate text objects but shared questions. DeepSeek execution uses temperature zero, thinking disabled, and a 4,096-token limit. GLM-5.3-Flash uses temperature zero and a 16,384-token limit. The configuration contrast includes these settings.

K.2 Within-configuration comparisons

The block-by-question bootstrap uses 20,000 paired draws. Each draw resamples twelve paired feedback/generic blocks and questions within four strata. The same draws produce the 95%, 97.5%, and 98.75% intervals. The representation family uses 98.75% intervals, including the fixed ± 1 F1-point equivalence tolerance for $F - S$. Source-usefulness and recipient-modulation families use 97.5%. These two family names retain the wording of the archived analysis plan, where "recipient" denotes the document receiving the appended section; it is the receiver role of Section 2, not the source. The planned representation intervals distinguish mean replacement from source usefulness. The nonbinary analysis and its 95% intervals are retrospective.

The original plan also specified a stronger joint criterion for preserving a useful generation increment. It requires positive full-skill gains over both G and N, positive corresponding section-only gains, and a positive lower bound for

$$\{v(S) - v(G)\} - 0.75\{v(F) - v(G)\}. \tag{K.22}$$

The difference form avoids a ratio with an uncertain denominator. This criterion is not established for either executor. The GLM $F - S$ equivalence instead establishes preservation of mean deployment performance within its margin. Positive increments on C^- and C concern those receivers. In particular, neither receiver is the root from which F was generated. Tables 18 and 19 provide the relevant absolute means and paired comparisons under both scoring interfaces.

K.3 Direct configuration differences

For contrast k, executor m, block b, and question q, let $d^{(k)}_{mbq}$ be the within-cell score difference between its two text conditions. The direct GLM-minus-DeepSeek comparison is

$$\widehat{\Gamma}_k = \frac{1}{12 \cdot 512} \sum_{b=1}^{12} \sum_{q=1}^{512} \big(d^{(k)}_{\mathrm{GLM},bq} - d^{(k)}_{\mathrm{DS},bq}\big). \tag{K.23}$$

Table 18: Fixed-section mean scores. Original: the existing answer-tag interface; polarity: additional leading yes/no normalization. The same frozen texts and questions are used by both executors.

Object	DS original	DS polarity	GLM original	GLM polarity
N	76.36	78.97	81.49	82.09
G	76.31	79.57	82.12	82.70
F	78.34	78.34	83.27	83.27
S	77.75	77.75	83.23	83.23
R	78.91	79.04	82.30	82.50
C	76.01	78.15	80.35	80.99
C^-	76.77	78.49	80.10	81.00
$C^- + S$	78.31	78.32	83.15	83.15
$C + S$	78.36	78.37	82.82	82.82
Title and headings	77.80	78.90	81.47	82.43
Title only	74.35	78.04	81.37	82.06

Table 19: Fixed-section paired contrasts with 95% intervals. The main representation decisions instead use the wider 98.75% intervals specified in the text.

Contrast	Score	DeepSeek	GLM-Flash
$F - G$	Original	+2.03 [-0.06, +4.11]	+1.15 [+0.03, +2.34]
	Polarity	-1.22 [-2.94, +0.41]	+0.56 [-0.42, +1.59]
$F - N$	Original	+1.98 [+0.05, +3.92]	+1.78 [+0.61, +3.05]
	Polarity	-0.63 [-2.24, +0.92]	+1.18 [+0.14, +2.28]
$F - C$	Original	+2.33 [+0.12, +4.88]	+2.92 [+1.60, +4.40]
	Polarity	+0.19 [-1.20, +1.55]	+2.28 [+1.12, +3.60]
$S - G$	Original	+1.44 [-0.92, +3.75]	+1.11 [-0.07, +2.37]
	Polarity	-1.82 [-3.87, +0.13]	+0.52 [-0.53, +1.63]
$F - S$	Original	+0.59 [-0.50, +1.77]	+0.04 [-0.59, +0.67]
	Polarity	+0.59 [-0.50, +1.77]	+0.04 [-0.59, +0.67]
$F - R$	Original	-0.57 [-1.78, +0.59]	+0.97 [+0.32, +1.73]
	Polarity	-0.70 [-1.90, +0.44]	+0.77 [+0.18, +1.40]
$(C^- + S) - C^-$	Original	+1.55 [+0.24, +2.90]	+3.05 [+1.80, +4.47]
	Polarity	-0.16 [-1.20, +0.88]	+2.15 [+1.08, +3.38]
$(C + S) - C$	Original	+2.34 [+0.24, +4.80]	+2.47 [+1.23, +3.87]
	Polarity	+0.22 [-0.98, +1.40]	+1.83 [+0.78, +3.03]
Absent–present append	Original	-0.80 [-2.99, +1.18]	+0.58 [-0.46, +1.55]
	Polarity	-0.39 [-1.65, +0.89]	+0.31 [-0.44, +1.06]
$(S - G) - 0.75(F - G)$	Original	-0.09 [-1.36, +1.10]	+0.24 [-0.40, +0.90]
	Polarity	-0.90 [-2.14, +0.25]	+0.10 [-0.53, +0.74]

Shared block and stratified-question bootstrap weights preserve the alignment of conditions and configurations. This retrospective analysis discloses six contrasts under each of the two scoring interfaces: $F - G$, $F - S$, $F - R$, the two append increments, and their absent-minus-present difference. Table 20 gives all twelve, with ordinary 95% intervals and conservative 99.5833% intervals for this family. The correction covers these twelve comparisons.

Table 20: Direct GLM-minus-DeepSeek differences in the same fixed-section contrasts. The retrospective family contains all six contrasts under both interfaces; its correction covers these twelve comparisons. All units are F1 points.

Contrast	Interface	Difference	95% interval	99.5833% interval
$F - G$	Original	-0.88	[-3.12, +1.43]	[-4.13, +2.58]
	Polarity	+1.79	[-0.07, +3.76]	[-0.94, +4.75]
$F - S$	Original	-0.55	[-1.82, +0.62]	[-2.48, +1.18]
	Polarity	-0.55	[-1.82, +0.62]	[-2.48, +1.18]
$F - R$	Original	+1.53	[+0.28, +2.84]	[-0.32, +3.49]
	Polarity	+1.47	[+0.25, +2.75]	[-0.35, +3.37]
$(C^- + S) - C^-$	Original	+1.50	[-0.11, +3.18]	[-0.81, +4.05]
	Polarity	+2.31	[+0.85, +3.90]	[+0.14, +4.76]
$(C + S) - C$	Original	+0.13	[-2.04, +2.15]	[-3.09, +3.09]
	Polarity	+1.61	[+0.25, +3.05]	[-0.41, +3.76]
Absent minus present	Original	+1.38	[-0.39, +3.22]	[-1.19, +4.29]
	Polarity	+0.70	[-0.68, +2.12]	[-1.44, +2.85]

L Repeated-Execution Design and Estimation

L.1 From a recorded change to a text-condition difference

The initial experiments record one response or trajectory for each distinct deployment input. They locate candidate gains, losses, and changes in task requirements. The repeated panel holds these texts fixed and runs every selected input four times. It compares two sources of variation: changing the deployed text, and running the same text again. The historical executions are kept separate from these four new executions.

There are two evaluation panels. The main panel selects questions or tasks using their inputs alone. A diagnostic panel revisits gains and losses found in the historical records. Selection on an earlier success can itself produce an apparent loss on re-execution, while selection on an earlier failure can produce an apparent recovery. Accordingly, every diagnostic comparison uses newly executed controls. For example, an extracted section is compared with a newly executed full skill on the old loss set, rather than with the old full skill's failed response.

L.2 QA questions, texts, and execution settings

The main QA panel contains 128 questions: 32 from each of the four original question families. Within each family, selection sorts a stable hash of the task identifier, question, and supplied context. It does not use answers, scores, or historical transition labels. Of these questions, 29 have a single canonical yes/no reference and 99 are nonbinary. All twelve generated skill blocks are retained.

For each generated block, we run the full skill F, its Answer Format section S, and the common starting prompt G. An identical G input is executed once per repetition and shared across blocks. The main panel therefore requires $128 \times (1 + 12 + 12) \times 4 = 12{,}800$ calls per configuration. We additionally freeze the historical source-gain and source-loss block–question sets. Under GLM these contain 143 and 120 cells; under the original DeepSeek configuration they contain 232 and 256. Each selected cell receives new F and S executions, with G shared by question. Exact overlaps with the main panel reuse the same new calls. The diagnostic sets are analyzed separately and do not enlarge the representative sample.

For the natural revision, the same 128-question panel receives seven texts: no attached skill, parent, parent with the old body removed, the latter with either component or both components, and parent

with the new additions. Each text is executed four times. This covers repair, use after withdrawing the old body, and the complete four-condition combination. The eighteen historical Executor A questions on which either component and their combination matched after a baseline miss are also fixed before collection. Questions outside the main panel receive the four combination conditions four times, yielding a separate fixed diagnostic set.

All QA repeated-execution configurations use a 16,384-token maximum output and requested temperature zero. GLM-5.3-Flash and Executor A retain their original executor configurations. The repeated panel uses DeepSeek V4.1 Flash with thinking disabled; the initial panel used V4 Flash. Its results therefore evaluate the frozen texts on the current configuration, rather than extend the original V4 executions as if they came from an unchanged model. No execution seed is supplied. Conditions, blocks, and repetition indices are interleaved in a fixed randomized order.

The natural-repair primary score removes residual answer tags and then canonicalizes eligible leading yes/no answers. The generated-section primary score preserves its original polarity-normalized interface. Both studies additionally report the original, tag-cleaned, polarity-normalized, and jointly cleaned scores. A completed response that exhausts its output budget receives the original protocol's zero score and is not redrawn. The frozen design prescribes 3,792 Executor A calls, 14,776 GLM calls, and 20,148 current-Flash calls, including the diagnostic panels.

L.3 Automation tasks and fresh trajectories

The main Automation panel selects four tasks per domain from the original twelve, again by a fixed input-identifier hash. This gives 24 tasks across six domains. We also freeze the union of the six historical Flash tasks and two historical Pro tasks on which the parent and additions separately passed but their concatenation failed. After overlap removal, both configurations evaluate the same 30 tasks: 24 main-panel tasks and six additional diagnostic tasks. The main and diagnostic analyses retain their separate membership.

Each task receives no attached skill, parent, additions, literal concatenation, and the full revision with the original placement of its additions. Four fresh trajectories per text give 600 trajectories per configuration. Flash uses the current provider-default thinking setting; Pro uses the high-thinking setting. Both have a 16,384-token response limit, at most 50 agent steps, and the same task presentation. One agent step can issue several tool calls. Each trajectory starts with a fresh conversation and its own copy of the fixed initial world state. The order of tasks, text conditions, and repetitions is randomized before collection.

Strict task success and the benchmark implementation's partial-credit diagnostic are computed from each trajectory's terminal world. The paired unit is the task, including all of its assertions. Assertion eligibility can depend on the resulting state, so eligible-assertion scores and fixed-assertion descriptions are kept distinct. The auxiliary requirement comparison uses each assertion's recorded passed flag, retaining all assertions within their task. A separate sensitivity uses three states: eligible failure, eligible success, and exclusion. Executions that reach a step or output limit remain in the evaluation and are scored from their terminal worlds.

L.4 Estimating the effects of reuse

We first average the four executions within each unique input. Main-panel QA comparisons use equal question-family weights and equal generated-block weights; Automation uses equal domain and task weights. The full-skill, section, and starting-prompt contrasts share the same question and block resampling weights. Automation comparisons retain all conditions and assertions of a task together. We use 20,000 stratified resamples of these cell means for task-and-block intervals, and separately report execution standard errors for the fixed panel. Shared starting-prompt calls have their coefficients combined before calculating execution variance. We do not add an extra bootstrap of repetitions inside the task bootstrap.

For fixed discovery sets we also report individual conservative 95% execution-only bounds (Hoeffding) on expected new-execution contrasts. These use the bounded score range and independent-call assumption rather than a sample variance estimated from four runs. Appendix J.5 derives the bounds and distinguishes their fixed-input target from the question-and-block intervals.

For extraction, the principal distribution comparison subtracts the average same-text disagreement from the full-versus-section disagreement. All cross-condition pairs of the four new outputs contribute to the latter; all distinct pairs of repeats contribute to the former. Appendix J.5 gives the estimator and its expectation. Conditional-label permutations within a unique block–question input test equality of the two response distributions. The two configuration tests form one Holm-adjusted family. This comparison concerns the response distribution, while differences in EM and F1 concern correctness and task score.

On a historical source-gain set, we report both new $F - G$ and new $S - F$. On a historical source-loss set, we report both new $G - F$ and new $S - F$. The first comparison asks whether the source gain or loss persists in new execution; the second asks what extraction changes relative to a newly executed source. For a historical Automation combination-loss set, the corresponding comparisons are new parent-minus-concatenation and new additions-minus-concatenation. These are conditional diagnostic sets with fixed membership, not samples of a general loss rate.

Combination comparisons use the four marginal mean scores and direct differences between joint and separate deployment. Pairing the first execution of every condition into a pattern does not identify a shared latent causal state across different prompts. We therefore use repeated success probabilities to evaluate a loss, while the original four-bit patterns continue to describe the original response table. The exact accounting identities in Appendix J apply to that table without making a claim about the stability of each pattern.

Mean-score tables give descriptive 95% intervals. Within each metric, the two main-panel comparisons of joint versus separate deployment under two configurations form a four-comparison family; their 98.75% intervals use Bonferroni adjustment. QA and Automation are separate families, as are their different score metrics. For mean-score replacement, we also report the full-versus-section 98.75% interval and compare it with the original one-point equivalence margin. This reference comparison is separate from the two-configuration distributional test. The sample sizes and four repetitions were fixed before the new task responses were inspected. Main-panel estimates and historical diagnostic estimates are reported separately throughout.

M Scores and Outcomes under Repeated Execution

These tables report four fresh executions per selected input. Main panels and fixed historical diagnostic sets are reported separately. Scores and differences use a 0–100 scale. Brackets give descriptive 95% question-level intervals, with crossed generation-block resampling where applicable; Automation intervals resample tasks within domains. Different levels are stated explicitly. Execution standard errors condition on the fixed inputs and are reported separately.

M.1 Natural revision and component combination

Here N attaches no skill, P is the parent, and R removes its old reasoning body. A denotes the two additions and B the replacement body. The full revision is $F = R + B + A$. The main panel has 128 questions in both configurations.

Table 21: Mean scores on the repeated natural-revision panel. Residual answer tags and eligible leading yes/no expressions are normalized.

Text	Executor A F1	Executor A EM	Flash F1	Flash EM
N	81.59	71.09	83.95	73.24
P	72.54	61.72	86.75	75.39
R	77.54	66.41	85.19	74.41
$P + A$	81.57	70.70	85.32	74.22
$R + A$	81.71	72.07	84.21	73.44
$R + B$	78.81	68.36	85.95	75.59
F	80.84	70.51	81.97	71.88

Table 22: Repair, append, and interaction contrasts on the 128-question repeat panel. $J = F - (R + A) - (R + B) + R$.

F1 contrast	Executor A	Flash
$F - P$	$+8.29$ $[+4.27, +12.57]$	-4.78 $[-8.54, -1.51]$
$F - N$	-0.75 $[-3.99, +2.22]$	-1.98 $[-5.11, +0.83]$
$(P + A) - P$	$+9.03$ $[+5.02, +13.31]$	-1.43 $[-3.57, +0.23]$
$(R + A) - R$	$+4.16$ $[+1.07, +7.52]$	-0.98 $[-3.75, +1.55]$
$(R + B) - R$	$+1.26$ $[-1.48, +4.07]$	$+0.76$ $[-1.99, +3.71]$
J	-2.13 $[-5.42, +0.88]$	-3.00 $[-6.89, +0.52]$

Table 23: Joint versus separate deployment. The 98.75% intervals adjust the four comparisons within each score metric.

Configuration	Contrast	F1	EM
Executor A	$F - (R + A)$	-0.87 $[-3.50, +1.27]$	-1.56 $[-4.49, +0.59]$
Executor A	$F - (R + B)$	$+2.03$ $[-0.30, +4.93]$	$+2.15$ $[-0.78, +5.66]$
Flash	$F - (R + A)$	-2.24 $[-5.80, +0.49]$	-1.56 $[-5.27, +1.56]$
Flash	$F - (R + B)$	-3.97 $[-8.99, +0.20]$	-3.71 $[-8.79, +0.59]$

Table 24: New executions on all eighteen historical Executor A repeated-coverage questions. The membership is fixed from the initial panel.

Configuration	EM contrast	95% CI	Execution SE
Executor A	(R+A)-R	$+59.72$ $[+41.67, +76.39]$	3.85
Executor A	(R+B)-R	$+55.56$ $[+38.89, +69.44]$	4.09
Executor A	F-(R+A)	$+4.17$ $[+0.00, +12.50]$	1.39
Executor A	F-(R+B)	$+8.33$ $[+1.39, +18.06]$	1.96
Flash	(R+A)-R	-1.39 $[-4.17, +0.00]$	1.39
Flash	(R+B)-R	-2.78 $[-8.33, +0.00]$	1.60
Flash	F-(R+A)	$+1.39$ $[+0.00, +4.17]$	1.39
Flash	F-(R+B)	$+2.78$ $[+0.00, +8.33]$	1.60

M.2 EXTRACTED SECTIONS AND THEIR FULL SOURCES

The main panel crosses all twelve text blocks with 128 questions. G is the shared starting prompt, F the full skill, and S its own extracted section. Historical gain and loss sets retain their original block–question membership.

Table 25: Repeated mean scores under the generated-section polarity interface.

Configuration	Score	G	F	S	$S - F$ [95%]
GLM	F1	84.23	84.81	84.81	-0.00 $[-0.66, +0.61]$
GLM	EM	73.24	74.82	74.71	-0.11 $[-0.90, +0.62]$
Flash	F1	83.27	84.34	82.83	-1.50 $[-3.42, +0.02]$
Flash	EM	72.85	74.30	72.95	-1.35 $[-3.48, +0.44]$

Table 26: Comparison with the original mean-replacement margin. The entire interval must lie within the margin.

Configuration	$S - F$ F1 [98.75%]	Within ± 1 F1
GLM	-0.00 $[-0.89, +0.82]$	Yes
Flash	-1.50 $[-4.13, +0.40]$	Not established

Table 27: Answer disagreement percentages and excess $E = B - (W_F + W_S)/2$. The two configuration tests form one Holm family; E measures output-distribution difference, not a score gain.

Configuration	W_F	W_S	B	E	Holm p
GLM	3.61	3.48	4.56	1.01	< 0.001
Flash	2.77	1.97	6.29	3.92	< 0.001

Table 28: New executions on fixed historical source-change sets. Both contrasts use fresh responses; brackets give 95% question-and-block bootstrap intervals. SE refers to fixed-input execution uncertainty of $S - F$.

Configuration	Set	Cells	$F - G$ EM	$S - F$ EM	SE
GLM	Old gains	143	$+57.34$ $[+42.73, +70.22]$	-9.79 $[-20.45, +0.60]$	1.95
GLM	Old losses	120	-50.83 $[-66.60, -34.04]$	$+10.00$ $[-2.48, +23.96]$	2.14
Flash	Old gains	232	$+13.04$ $[+2.19, +25.30]$	$+2.26$ $[-6.03, +10.86]$	0.61
Flash	Old losses	256	-3.22 $[-16.74, +10.40]$	-14.45 $[-24.72, -4.95]$	0.76

Table 29: Individual conservative 95% execution-only bounds (Hoeffding) for new-execution uncertainty on each frozen discovery set. The preceding table separately reports uncertainty across questions and generation blocks. These bounds do not use the observed sample variance.

Configuration	Fixed set	$S - F$ EM	Execution-only bound
GLM	Old gains	-9.79	$[-17.82, -1.76]$
GLM	Old losses	+10.00	$[+1.23, +18.77]$
Flash	Old gains	+2.26	$[-4.04, +8.57]$
Flash	Old losses	-14.45	$[-20.46, -8.45]$

Table 30: Per-block F1 effects on the common 128-question repeat panel. Intervals resample questions within the four families.

Block	GLM $F - G$	GLM $S - F$	Flash $F - G$	Flash $S - F$
1	$+0.34$ $[-1.01, +1.88]$	$+0.53$ $[-0.56, +1.78]$	$+1.35$ $[-0.73, +3.79]$	-0.29 $[-2.85, +2.00]$
2	$+0.36$ $[-0.99, +1.96]$	$+0.55$ $[-0.49, +1.86]$	$+0.13$ $[-1.61, +1.93]$	-1.81 $[-5.01, +1.16]$
3	-0.57 $[-1.99, +0.79]$	$+0.45$ $[-0.46, +1.58]$	$+2.20$ $[+0.22, +4.64]$	-2.73 $[-5.64, -0.33]$
4	$+1.34$ $[-0.20, +3.17]$	-0.62 $[-1.99, +0.59]$	$+1.08$ $[-0.26, +2.68]$	-1.22 $[-3.65, +0.83]$
5	$+0.34$ $[-0.78, +1.64]$	-0.08 $[-1.55, +1.70]$	$+1.42$ $[-0.53, +3.64]$	-1.78 $[-4.39, +0.61]$
6	$+1.43$ $[-0.59, +3.91]$	-0.68 $[-2.70, +0.85]$	$+1.69$ $[-1.07, +4.67]$	-4.55 $[-8.41, -1.12]$
7	$+1.52$ $[-0.04, +3.42]$	-1.03 $[-2.82, +0.52]$	$+0.10$ $[-1.97, +2.20]$	-0.90 $[-3.78, +1.91]$
8	$+0.57$ $[-1.08, +2.38]$	$+0.49$ $[-0.26, +1.43]$	-0.28 $[-2.57, +1.95]$	-0.32 $[-2.62, +2.00]$
9	$+0.14$ $[-0.95, +1.36]$	-0.40 $[-1.11, +0.16]$	$+0.99$ $[-0.32, +2.58]$	$+0.03$ $[-1.17, +1.26]$
10	$+0.44$ $[-1.00, +2.11]$	-0.40 $[-1.58, +0.61]$	$+1.58$ $[-0.37, +3.91]$	-1.01 $[-3.48, +1.10]$
11	$+0.12$ $[-1.63, +2.01]$	$+0.66$ $[-1.14, +2.86]$	$+1.48$ $[-0.33, +3.56]$	-2.65 $[-6.02, +0.35]$
12	$+0.89$ $[-0.78, +2.76]$	$+0.51$ $[-0.72, +1.79]$	$+0.99$ $[-1.30, +3.44]$	-0.83 $[-2.85, +1.19]$

M.3 Automation combination and placement

The main panel contains 24 tasks across six domains. N uses no skill, P the parent, S the additions, A their end-appended concatenation, and F the full revision with its original placement.

Table 31: Mean eligible-assertion completion and strict task success over four trajectories per task and text.

Text	Flash partial	Flash strict	Pro partial	Pro strict
N	71.81	40.63	59.61	23.96
P	64.30	31.25	58.45	25.00
S	69.53	39.58	56.93	20.83
A	68.28	35.42	58.11	26.04
F	71.21	41.67	58.50	27.08

Table 32: Combination effects with 98.75% intervals, adjusting the four comparisons within each metric.

Configuration	Contrast	Partial credit	Strict success
Flash	$A - P$	$+3.98$ $[-5.88, +13.69]$	$+4.17$ $[-9.38, +17.71]$
Flash	$A - S$	-1.25 $[-7.83, +4.95]$	-4.17 $[-13.54, +5.21]$
Pro	$A - P$	-0.34 $[-7.70, +6.21]$	$+1.04$ $[-6.25, +8.33]$
Pro	$A - S$	$+1.18$ $[-6.68, +8.73]$	$+5.21$ $[-4.17, +15.62]$

Table 33: Placement and fixed requirements. Assertions are averaged within tasks, and label permutations move whole trajectories. The two configurations form a separate auxiliary Holm family.

Configuration	W_F	W_A	B	E	Holm p
Flash	13.54	15.04	14.71	0.42	0.662
Pro	7.61	10.97	9.32	0.02	0.662

Table 34: Every historical combination-loss task under both fresh configurations. Each four-digit string gives four independent strict outcomes. Equal positions across columns do not represent shared execution states.

Configuration	Historical task	N	P	S	A	F
Flash	Expense categorization	0000	0010	0010	0101	0000
Flash	Background-check follow-up	1111	0000	0001	1100	1001
Flash	Compensation adjustment	0101	0110	0101	0010	1101
Flash	Blog refresh queue	1111	0111	1101	1101	1110
Flash	Vendor approval	0101	1100	1111	0111	1001
Flash	Customer overview	0011	1101	1101	0100	1011
Flash	Multicurrency payment	0000	1100	1000	0000	0010
Flash	Trending topic	0000	0000	0000	0000	0000
Pro	Expense categorization	0000	0000	0000	0000	0000
Pro	Background-check follow-up	0000	0000	0000	0000	0001
Pro	Compensation adjustment	0000	0000	1000	0000	0000
Pro	Blog refresh queue	0101	0001	0101	0000	0000
Pro	Vendor approval	1111	1111	1111	1111	1111
Pro	Customer overview	0000	0001	0001	0000	0100
Pro	Multicurrency payment	0011	1110	1010	0111	0011
Pro	Trending topic	0100	1000	0100	0000	1101

M.4 Scoring-interface sensitivity of the repeats

These comparisons score the same newly executed answers. Cleaning answer tags and canonicalizing leading yes/no expressions change the measurement interface; they do not trigger another execution. All intervals below are descriptive 95% task-and-block intervals.

Table 35: Natural-revision F1 contrasts under four scoring interfaces. The main text uses joint tag and yes/no normalization.

Configuration	Interface	$F - P$	$F - N$
Executor A	Original	$+25.84$ $[+20.24, +31.53]$	$+4.07$ $[-0.28, +8.45]$
Executor A	Tag cleanup	$+8.94$ $[+4.86, +13.23]$	$+1.31$ $[-2.18, +4.68]$
Executor A	Yes/no	$+25.84$ $[+20.24, +31.53]$	$+2.60$ $[-1.56, +6.83]$
Executor A	Both	$+8.29$ $[+4.27, +12.57]$	-0.75 $[-3.99, +2.22]$
Flash	Original	-4.92 $[-8.68, -1.66]$	-2.12 $[-5.25, +0.69]$
Flash	Tag cleanup	-4.92 $[-8.68, -1.66]$	-2.12 $[-5.25, +0.69]$
Flash	Yes/no	-4.78 $[-8.54, -1.51]$	-1.98 $[-5.11, +0.83]$
Flash	Both	-4.78 $[-8.54, -1.51]$	-1.98 $[-5.11, +0.83]$

Table 36: Generated-skill F1 contrasts under four scoring interfaces. Yes/no normalization is the primary interface. These descriptive intervals do not replace the 98.75% replacement comparison.

Configuration	Interface	$F - G$	$S - F$
GLM	Original	$+2.28$ $[+0.25, +4.66]$	-0.00 $[-0.66, +0.61]$
GLM	Tag cleanup	$+2.28$ $[+0.25, +4.66]$	-0.00 $[-0.66, +0.61]$
GLM	Yes/no	$+0.58$ $[-0.67, +2.05]$	-0.00 $[-0.66, +0.61]$
GLM	Both	$+0.58$ $[-0.67, +2.05]$	-0.00 $[-0.66, +0.61]$
Flash	Original	$+1.06$ $[-0.41, +2.81]$	-1.52 $[-3.43, +0.01]$
Flash	Tag cleanup	$+1.06$ $[-0.41, +2.81]$	-1.52 $[-3.43, +0.01]$
Flash	Yes/no	$+1.06$ $[-0.41, +2.81]$	-1.50 $[-3.42, +0.02]$
Flash	Both	$+1.06$ $[-0.41, +2.81]$	-1.50 $[-3.42, +0.02]$

M.5 Detectability of an average Automation cost

This retrospective planning analysis uses only the 24 input-selected tasks, excluding historical diagnostic sets. The target is the mean strict-success contrast of concatenation against the parent

($A - P$) or additions ($A - S$), under Flash and Pro. We use two-sided $\alpha = 0.05/4$ per comparison, matching the four-comparison family behind the 98.75% intervals, and target 80% power. The minimum detectable effect (MDE) is the difference reaching that power under the variance model below, rather than a hard threshold below which detection is impossible.

Let d_i be a task's difference between four-execution text means, ν the average sum of the two within-task single-execution variances, and τ^2 the within-domain variance of true task effects. With equal allocation across six domains, N tasks, and R independent trajectories per text and task, the planning variance is

$$\mathrm{Var}(\widehat{\Delta}) \approx \frac{\tau^2 + \nu/R}{N}. \tag{M.24}$$

We estimate τ^2 by the mean of the six unbiased sample variances of d_i, minus $\widehat{\nu}/4$, truncated at zero; $\widehat{\nu}$ uses unbiased variances of the four trajectories per text and task. Flash $A - S$ reaches the zero boundary; the other three estimates are positive. We use a noncentral-t planning approximation with $N - 6$ degrees of freedom, solve numerically for 80% power, and round future task counts upward to multiples of six. This retains task pairing and domain stratification rather than treating repeated trajectories as independent tasks. Monte Carlo checks reproduce the target power under this planning model.

Table 37: Planning standard errors and 80%-power MDEs for the current 24-task, four-execution design. The final columns give the design for a five-point difference, retaining four executions per text and task. Trajectory counts are $2 \times N \times 4$ for two texts under one configuration. Running all four comparisons on the largest shared task panel (432 tasks, three texts, two configurations) would use 10,368 trajectories.

Config.	Contrast	SE (pp)	MDE (pp)	Tasks for 5 pp	Two-text trajectories
Flash	$A - P$	6.31	23.1	432	3,456
Flash	$A - S$	4.81	17.6	252	2,016
Pro	$A - P$	3.45	12.7	132	1,056
Pro	$A - S$	4.85	17.8	258	2,064

These numbers assume that the estimated variance structure persists across additional tasks from the same six-domain mixture. Four tasks per domain provide limited precision for the variance components, so the projected counts are planning scales rather than collection guarantees. More repetitions reduce the ν/R term; more tasks also improve measurement of between-task effects. This calculation neither turns intervals spanning zero into evidence of no cost nor establishes a detection limit for other studies. It quantifies the resolution of this panel at stated error and power levels.

N Guide to Scores and Uncertainty

The studies use common comparison principles but retain their own task panels and score interfaces. This appendix locates those choices so that a numerical interval can be read against the estimand it describes. No normalization below requires another model call.

N.1 Score interfaces

The QA evaluator extracts an answer and applies HotpotQA token normalization for F1 and exact match. Tag cleanup removes residual answer delimiters. Polarity normalization reduces an eligible leading yes/no expression to the binary label when the reference has exactly that form. Appendix I gives the exact rule. These operations can change measured effects of answer-construction guidance, so the natural, repair, and section tables retain their alternative interfaces.

The six-configuration source panel reports original and polarity-normalized F1; the Executor A example in Section 2 uses the original interface. The independent DeepSeek source panel uses tag-cleaned F1, matching its source analysis. Prediction risks use squared errors of per-question changes on the 0–1 task-score scale; their values are not percentage-point improvements. The natural-repair repeated panel uses tag and polarity cleanup, and the generated-section panel uses polarity normalization. Automation reports strict task success and a diagnostic fraction of eligible assertions passed; these concern terminal world states, rather than answer-string similarity.

N.2 Intervals and their comparison families

Table 38: Uncertainty conventions. Different levels correspond to different, explicitly defined comparison families. A fixed-set execution bound has a different sampling target from a question-and-block interval.

Evidence	Interval or test	Resampled or conditioned units
Natural source panels	Descriptive 95%	Sources and stratified questions; all conditions share weights.
Proposal prediction	Exploratory 95%	Source blocks and the common questions; fixed predictor risks.
Deployment-gate replay	Individual descriptive 95%	Source blocks and stratified questions; evaluation outcomes reweighted with actions fixed on the original question panel.
Fully crossed prediction	Individual descriptive 95%	Replacement pairs within families, receivers, and stratified questions; 5,000 fixed-risk draws.
Budget decomposition	Individual descriptive 95%	Same fully crossed draws applied to fixed cell moments; untruncated intervals for pooled mean-error estimates.
Five-arm content controls	Individual descriptive 95%	Source lineages, events, receiver slots, and stratified questions; no six-contrast family adjustment.
Broad section interventions	97.5% or 98.75% family intervals	Generation blocks and stratified questions; source-usefulness and representation families are separate.
Direct configuration effects	99.5833%	Twelve configuration-difference comparisons, sharing texts and bootstrap weights.
Repeated-panel means	Descriptive 95%	Question means across executions, crossed with blocks where applicable; tasks within domains for Automation.
Repeated mean replacement	98.75% reference interval	Original replacement criterion retained before inspection of fresh QA effects; separate from the distribution-test family.
Repeated joint deployment	98.75%	Four-comparison families within a study and metric.
Historical diagnostic inputs	Individual 95% Hoeffding bounds	Fixed input membership; independent new executions, with shared response aliases merged.
Answer/requirement distributions	Paired permutation, Holm	Conditional text-label exchange; two configurations per study. Automation exchanges whole trajectories.

The broad natural-repair panel also reports its original twelve-comparison 99.5833% family and exploratory secondary contrasts (Appendix Q). The broad Automation panel's three primary contrasts use 98.3333% intervals; its full descriptive table uses 95% intervals (Appendix S). Repeated-panel combination comparisons use their own four-comparison families. These labels identify each interval's estimand, comparison family, and sampling design.

N.3 Shared inputs and fixed diagnostic membership

The twelve section blocks share a starting prompt and evaluation questions. A shared starting response is one observation with multiple uses, not twelve independent observations. Source clusters, generated blocks, questions, and execution repetitions are likewise different levels of support. Their roles are specified in the individual protocols. Fixed historical gain/loss sets select informative questions from earlier observations; new source and successor executions compare the two deployments on those same inputs. Their execution-only bounds assess uncertainty for that fixed set. Resampling questions and blocks asks a broader question and produces the separate intervals reported alongside them.

O Response Changes and a Concrete Section Use

The study combines a broad single-execution panel with repeated execution on an input-selected panel and fixed diagnostic sets. This section gives the broad panel and its text constructions; Appendix L specifies repeated sampling and estimation.

O.1 Agreement and reference-match accounting

The analysis uses the GLM responses for the same twelve blocks and 512 questions. A cell is a block–question pair. The common root response is shared across blocks, and each unique text–question input has one recorded response. Counts therefore describe 6,144 recorded comparisons with shared questions, rather than 6,144 independent trials or repeated estimates of an answer probability.

Let a_X be the normalized extracted answer for condition $X \in \{G, F, S\}$, and let m_X indicate an exact reference match under the same interface. Overall agreement is the fraction with $a_F = a_S$. Change-conditioned agreement restricts the denominator to $a_F \neq a_G$. A change in normalized answer can leave EM unchanged, so agreement and match transitions answer different questions.

Table 39: Complete agreement and EM transition accounting under the two answer interfaces. The polarity interface changes some root comparisons but leaves the section-versus-full net match difference unchanged.

Recorded quantity	Original	Polarity
All block–question cells	6,144	6,144
$a_F = a_S$	5,785	5,785
$a_F \neq a_G$	609	561
$a_F = a_S$ within $a_F \neq a_G$	359	311
Overall agreement (%)	94.16	94.16
Agreement where F changes G (%)	58.95	55.44
Full-skill gains: $m_G = 0, m_F = 1$	191	143
Of these, retained by S	147	99
Of these, lost by S	44	44
Full-skill losses: $m_G = 1, m_F = 0$	120	120
Of these, restored by S	41	41
New S matches with $m_G = m_F = 0$	45	45
Lost S matches with $m_G = m_F = 1$	34	34
Net match change $F - G$	+71	+23
Net match change $S - G$	+79	+31
Net match change $S - F$	+8	+8

Under polarity normalization, S obtains $41 + 45 = 86$ matches absent from F and loses $44 + 34 = 78$ matches obtained by F. Together with the shared match states, this accounts for every cell's EM relation. The section's eight additional matches do not imply an F1 advantage: F1 assigns partial credit to nonmatching answers, and its point estimate is slightly lower for S.

O.2 Question coverage and repeated execution

Table 40 gives the six nontrivial match transitions for both executors. The GLM transitions comprise 342 block–question cells on 63 different questions. Of these, 263 cells on 57 questions change match status between the starting prompt and full skill. A question can appear in different transition categories across generated blocks: eleven questions occur among both the retained and lost full-skill matches, and eleven among both recovered and still-missing starting matches. Counts across rows therefore describe overlapping question sets.

The starting prompt has 512 actual executions, each reused across twelve blocks. Each of the twelve full skills and twelve sections has one execution per question. Cross-block agreement thus compares different texts against a shared starting answer. It does not provide repeated executions of the same input. These initial transitions fix the source-gain and source-loss sets for the repeated execution. Appendix L executes the starting prompt, full skill, and section four times on those sets, comparing

Table 40: Reference-match transitions and their question coverage. Each executor has twelve generated skill blocks and 512 questions. The question sets can overlap across rows when different blocks produce different transitions on the same question.

	GLM		DeepSeek	
Recorded transition	Cells	Questions	Cells	Questions
Full-skill match retained	99	19	177	31
Full-skill match lost	44	20	55	22
Starting match recovered	41	21	68	40
Starting match still missing	79	19	188	42
New section match	45	19	74	27
New section loss	34	14	133	50

extraction with a freshly executed source. The bootstrap reported here remains a question-and-block resampling of the initial response table.

O.3 F1 CHANGES ON THE SAME OBSERVATIONS

Let $f = \mathrm{F1}(F) - \mathrm{F1}(G)$ and $s = \mathrm{F1}(S) - \mathrm{F1}(G)$ within each cell. The mean positive and negative full-skill increments are

$$M_+ = \mathbb{E}[\max(f, 0)], \qquad M_- = \mathbb{E}[\max(-f, 0)]. \tag{O.25}$$

For polarity F1, these are 1.960 and 1.396 points, leaving a net $F - G$ increment of 0.564. Define positive overlap as $O_+ = \mathbb{E}[\min(\max(f, 0), \max(s, 0))]$. It is 1.231 points, or 62.79% of M_+. This descriptive overlap is conditional on observed cell differences, not a population retention rate. The corresponding negative overlap is 0.873 points, or 62.54% of M_-.

Of the 6,144 cells, F has higher F1 than G in 247, lower F1 in 230, and equal F1 in 5,667. On these three subsets, the section's contributions to its overall $S - G$ difference are respectively $+1.298$, -0.794, and $+0.017$ points, summing to $+0.521$. Under original F1, $M_+ = 2.545$, $M_- = 1.396$, and $O_+ = 1.815$ points. Both interfaces show that the section's aggregate score combines overlap with the full skill's changes and other observed changes.

O.4 A PASSAGE THAT CHANGES THE DELIVERED ANSWER BOUNDARY

The illustrative case asks which role made Keira Knightley famous. The reference is *Elizabeth Swann*. It was selected retrospectively among nonbinary questions where S improves EM over G and $C + S$ improves EM over C. The intervention changes the complete section reproduced below; the proper-name clause helps interpret the response pattern but was not separately ablated.

The complete retained section.

```
<!-- skill-section:answer_format -->
```

Answer Format

Output the minimal exact answer that matches the gold label's lexical form. Include the full noun phrase when the gold uses one (e.g., 'documentary film' not 'documentary'). For entity questions, give the proper name only, without extra descriptors or parentheticals. For yes-no questions, respond with a single capitalized 'Yes' or 'No'. Never embed reasoning, alternatives, or meta-commentary inside the answer. If the question asks 'which monarchy', answer with the monarch's name if that is the gold entity, not the dynasty.

All eight responses have complete answer blocks. The longer answers contain the reference character and a film-series description; the shorter answers match the reference name. Their F1 scores are 0.4 and 1 respectively, and their EM scores are zero and one. Polarity normalization does not change these nonbinary answers. The example concerns the boundary of the answer supplied to the evaluator; it does not establish acquisition of previously absent factual knowledge.

Table 41: All eight extracted answers for the Keira Knightley example. These strings are the contents of complete answer blocks.

Object	Complete extracted answer	F1	EM
G	Elizabeth Swann in the "Pirates of the Caribbean" film series	0.4	0
C	Elizabeth Swann in the "Pirates of the Caribbean" film series	0.4	0
C^-	Elizabeth Swann in the "Pirates of the Caribbean" film series	0.4	0
R	Elizabeth Swann in the "Pirates of the Caribbean" film series	0.4	0
F	Elizabeth Swann	1.0	1
S	Elizabeth Swann	1.0	1
$C^- + S$	Elizabeth Swann	1.0	1
$C + S$	Elizabeth Swann	1.0	1

P Block Variation and Answer Differences

P.1 A fixed subset and shared uncertainty

The retrospective reanalysis uses existing answers without additional model executions. A binary-reference question has exactly one reference whose stripped lowercase form is `yes` or `no`. This excludes 114 questions, all from the verification stratum. The remaining 398 questions consist of 128 comparison, 128 shared-attribute, 128 multi-hop, and 14 verification questions. Means weight these remaining questions equally; the subset is not a four-stratum equally weighted population. Original and polarity-normalized F1 coincide in every cell on this subset, under both executors and all eleven logical text conditions.

The reanalysis retains every generation block and all six reported contrasts: $F - G$, $S - G$, $F - S$, $F - R$, and the two receiver append increments. It uses 20,000 paired bootstrap draws, resampling twelve blocks and questions within their remaining strata. Each draw uses the same block and question weights for both execution configurations and all contrasts. The new intervals are marginal 95% exploratory intervals. A positive block mean is a descriptive direction count on shared questions, not a separate hypothesis test or an equivalence conclusion for that block.

The nonbinary results locate part of the full-panel effect. For GLM's normalized $F - R$ difference of 0.7654 points, the nonbinary questions contribute $398/512 \times 0.8641 = 0.6717$ points. For the C^- append difference of 2.1457 points, they contribute $398/512 \times 1.5139 = 1.1768$ points, with the binary questions contributing the remaining 0.9689. This arithmetic separates reference subsets; it does not identify a causal answer-type effect because the questions themselves also differ.

P.2 Literal section identity and the unrun position control

All twelve feedback Answer Format sections have different literal texts; the extraction rule is shared. The generic sections are also distinct, and none is byte-identical to its paired feedback section. In the original five-section schema, Answer Format precedes Verification. Removing the generic section and appending that same text at the document's end therefore produces a different document from the original C. This is true in all twelve blocks after applying the execution renderer's whitespace handling, and none of those reconstructed texts matches the 99 executed text objects. The existing C versus C^- comparison measures removal from its native position, rather than the same-position append effect of the generic section.

The generic generation input includes a general QA proposer instruction, the minimal root, and the five-section schema, with no execution trajectories. It does not independently supply the complete executor prompt and scoring specification as a direct-task-contract baseline. A matched deployment comparison of feedback, generic, and directly specified sections would distinguish these content sources under the same insertion conditions.

P.3 Deterministic lexical relations among discordant answers

We compare the normalized answers for G, F, and S from the same response panel. For each full–section disagreement, we ask whether one normalized token sequence is a strict contiguous

Table 42: All six nonbinary-reference contrasts (398 shared questions, 12 generation blocks). Values are F1 percentage-point changes with paired, exploratory 95% percentile intervals; +/12 counts blocks with a strictly positive observed mean difference. The original and polarity scoring interfaces coincide on this subset. Blocks share test questions; the direction counts are descriptive.

	GLM-5.3-Flash		DeepSeek-v4-Flash	
Contrast	F1 change [95% CI]	+/12	F1 change [95% CI]	+/12
$F - G$	+0.27 [-0.87, +1.44]	8	-0.94 [-2.81, +0.87]	1
$S - G$	+0.11 [-1.12, +1.36]	6	-1.52 [-3.88, +0.71]	1
$F - S$	+0.16 [-0.53, +0.86]	8	+0.57 [-0.65, +1.87]	8
$F - R$	+0.86 [+0.16, +1.63]	12	-0.54 [-1.81, +0.75]	4
$(C^- + S) - C^-$	+1.51 [+0.44, +2.63]	12	-0.16 [-1.41, +1.10]	6
$(C + S) - C$	+1.12 [+0.11, +2.18]	10	+0.65 [-0.70, +2.00]	8

Table 43: All six direct execution-configuration differences on the nonbinary subset. Units are F1 points; shared bootstrap weights preserve the matched content, block, and question comparisons. These are retrospective marginal intervals.

Nonbinary contrast	Direct GLM-minus-DeepSeek difference [95% CI]
$F - G$	$+1.21\ [-0.78, 3.26]$
$S - G$	$+1.63\ [-0.75, 4.09]$
$F - S$	$-0.41\ [-1.88, 0.99]$
$F - R$	$+1.41\ [0.05, 2.74]$
$(C^- + S) - C^-$	$+1.67\ [0.15, 3.22]$
$(C + S) - C$	$+0.48\ [-0.95, 1.94]$

subspan of the other. Equal sequences do not enter this set. The two directional subspan classes are mutually exclusive; all other pairs are classed as non-nested. This is a literal relation between returned strings, not a semantic classification or an inferred model operation.

Table 44: Complete lexical-relation counts for polarity-normalized full–section answer disagreements. Counts are block–question comparisons; repeated questions and root responses are shared. "Inside" means strict contiguous token subspan.

Executor	Questions	Disagreements	S inside F	F inside S	Non-nested
GLM	All 512	359	84	49	226
GLM	Nonbinary 398	338	84	49	205
DeepSeek	All 512	661	103	108	450
DeepSeek	Nonbinary 398	603	103	108	392

The 359 GLM disagreements involve 94 unique questions and 244 distinct question/root/full/section triples; the nonbinary subset has 88 unique questions. The 661 DeepSeek disagreements involve 164 unique questions and 412 distinct triples, with 146 unique questions in the nonbinary subset. Differences between accent forms, synonyms, and equivalent word orders can be non-nested. Thus the category excludes literal trimming as a complete description of those recorded differences, while leaving their semantic relation to be interpreted from the question and evidence.

P.4 Purposefully selected response examples

The nonbinary restriction also retains the opposing GLM match transitions in the central analysis. Of the 44 lost full-skill gains, 36 are nonbinary; of the 41 restored root matches, 40 are nonbinary. The corresponding new section gains and losses where root and full share a match state are 33 and 34. Thus the section-versus-full net match difference is $40 + 33 - 36 - 34 = 3$ on this subset. These counts use the same cell definitions as Appendix O and the same fixed reference mask as the score reanalysis.

We directly inspected fifteen response triples together with their supplied evidence, chosen to examine different match transitions and lexical patterns. This was a purposefully selected reading set rather than a random sample or a blinded semantic-labeling study. It included answer-boundary changes, accent and grammatical variants, alternative supported occupations, entity choices, and relation-target changes. The inspection informs the examples and prevents treating all non-nested strings as factual disagreements. It supplies no estimate of category prevalence.

Selecting the Chinese rank. This GLM case concerns the title held by Sun Yat-sen. **Question:** Sun Yat-sen briefly took a title of what Chinese military ranking? **Reference:** dàyuánshuài

Supplied evidence excerpt: Taewŏnsu (literally grand marshal, usually translated as "generalissimo") is the highest possible military rank of North Korea and is intended to be an honorific title for Kim Il-sung and Kim Jong-il. The rank is senior to that of "wonsu" (marshal). The title also exists in Chinese military history as "dàyuánshuài" [...], and was briefly taken by Sun Yat-Sen.

> **Root** G**:** Taewonsu (grand marshal/generalissimo, "dàyuánshuài")
> **Full** F**:** Taewonsu
> **Section** S**:** Dàyuánshuài

Complete extracted section:

> <!– skill-section:answer_format –>
> ## Answer Format
> Match the gold answer's granularity and phrasing. Do not include nicknames, parentheticals, or 'a.k.a.' suffixes. For yes/no questions, answer with exactly 'Yes' or 'No' and no extra words. For choice questions, output only the selected option's canonical name as it appears in the gold, which may be shorter than the full descriptive phrase. Avoid articles unless the gold clearly includes them.

Returning the castle instead of its oldest part. This DeepSeek case concerns the oldest part of the castle where Robert Hay was born. **Question:** What is the oldest section from where Robert Hay was born? **Reference:** Norman Keep or Pele Tower

Supplied evidence excerpt: Robert Hay (6 January 1799 – 4 November 1863) was a Scottish traveller, antiquarian, and early Egyptologist. He was born in Duns Castle, Berwickshire. During his service in the Royal Navy he visited Alexandria, Egypt, in 1818. Duns Castle, Duns, Berwickshire is a historic house in Scotland, the oldest part of which, the massive Norman Keep or Pele Tower, dates from 1320.

> **Root** G**:** the massive Norman Keep or Pele Tower, dating from 1320
> **Full** F**:** Norman Keep or Pele Tower
> **Section** S**:** Duns Castle

Complete extracted section:

> <!– skill-section:answer_format –>
> ## Answer Format
> Format the answer to match the expected semantic type: a person's name for who-questions, a job title for occupation-questions, a magazine title for selection-questions, and a simple 'Yes' or 'No' for verification-questions. Use the full canonical name when known, as partial names may fail exact matching. For boolean questions, output the word 'Yes' or 'No' without additional phrasing. For list or common-attribute questions, provide the concise shared term, not a descriptive sentence. Ensure the answer stands alone as a direct response.

Restoring the requested discipline. This GLM case concerns Johannes Knoops's discipline. **Question:** In what discipline is Johannes Knoops a leading authority at New York's Fashion Institute of Technology? **Reference:** architecture

Supplied evidence excerpt: Johannes Knoops is an American architect, international architectural correspondent and professor of architecture in the department of interior design at the Fashion Institute of Technology in New York city.

Root G: Architecture
Full F: interior design
Section S: Architecture

Complete extracted section:

<!– skill-section:answer_format –>
Answer Format
Provide the answer solely as the required entity, value, or short phrase. For yes/no questions, respond with exactly 'yes' or 'no' without any explanation or preamble. For factual questions, give the name, number, or category directly. Do not include reasoning, full sentences, or extra details in the answer tag. If the question implies a choice between options, select the correct option name only. Ensure the answer matches the expected type (e.g., numeric values without commas or units unless specified) to maximize exact match scoring.

References to gold answers above are generated instruction wording; held-out references are not supplied to the executor. These cases were selected retrospectively to interpret different observed transitions. The Sun Yat-sen case is a new section match, the Robert Hay case loses a full-skill gain, and the architecture case restores a root match. They do not estimate the prevalence of semantic response categories.

Q THE FROZEN NATURAL REPAIR

The study combines a broad single-execution panel with repeated execution on an input-selected panel and fixed diagnostic sets. This section gives the broad panel and its text constructions; Appendix L specifies repeated sampling and estimation.

Q.1 SELECTION AND TEXT CONDITIONS

The repair study freezes an accepted natural update, separate from the 72-event natural panel in Appendix I. It was generated on the historical Executor A configuration using eight training questions and four development questions, and accepted as a new best candidate by that run's gated procedure. Of the proposed operations, one was skipped; the actual applied change is one replacement and two appends. The present intervention uses these applied texts. An earlier 64-question discovery analysis inspected three updates and motivated selecting this one before the 512-question confirmation panel. The intervention's estimates are conditional on that selection.

Let the parent be $P = R + H$, where H is the old Multi-Hop Reasoning body and R is the remainder, retaining the heading. The new body is B, and the two appended passages jointly form A. The full revision is $F = R + B + A$. Ten frozen objects are evaluated: no skill N; P; R; $P + A$; $R + A$; $R + B$; F; new text alone $S = B + A$ with its headings; inert replacement I; and $I + A$. The inert comment has the same character count as H, with the surrounding document unchanged. These equations describe document assembly, with each component inserted at its recorded position.

The Executor A and DeepSeek executors receive identical objects and 512 questions, stratified into four question families. Each has 5,120 terminal object–question records. All Executor A records are complete; DeepSeek has 5,115 complete responses and five incomplete outputs, scored as zero. Executor A uses a 16,384-token limit; DeepSeek uses the recorded Flash configuration with a 4,096-token limit. These are execution configurations, including their service settings. The two appended passages are tested together, so the comparisons do not separately assign their joint effect to answer-type checking or entity resolution.

Q.2 INFERENCE AND SCORE INTERFACES

The primary F1 removes residual answer tags and then applies the pre-existing leading yes/no transform. The bootstrap uses 20,000 draws, resampling questions within each of the four families. It does not resample updates. The original primary family contains six contrasts under two executors: $F - P$, $R - P$, $F - (R + A)$, the difference between the append effects on R and P, $S - N$, and $(F - P) - (S - N)$. The last contrast compares the full update's parent-relative increment with the

independently deployed new text's increment over no skill. Its twelve comparisons use 99.5833% intervals.

The three append increments on P, R, and I were specified as secondary comparisons in the original analysis plan. The table below gives their 95% intervals. The direct $F - (P + A)$ comparison also uses 95%. Its small point difference is a descriptive proximity of means; no equivalence margin was specified for that comparison. The append-effect difference between the removed and intact backgrounds is -1.86 $[-4.31, 0.60]$ on Executor A. Thus the three positive append increments do not by themselves establish a difference in their magnitudes across backgrounds.

Table 45: Mean primary F1 for the ten frozen repair objects. These are absolute scores, not increments.

Object	Executor A	DeepSeek
N	79.48	77.42
P	72.32	79.49
R	75.33	80.11
$P + A$	78.24	77.48
$R + A$	79.38	78.15
$R + B$	76.93	79.78
F	78.18	78.30
S	78.99	78.23
I	76.77	79.10
$I + A$	79.26	77.16

Table 46: Direct repair comparisons with 95% intervals. The family-adjusted interval for the main Executor A repair is [2.20, 9.68] at 99.5833%.

Contrast	Executor A	DeepSeek
$F - P$	+5.86 [+3.33, +8.48]	-1.18 [-3.40, +1.05]
$R - P$	+3.01 [+0.90, +5.15]	+0.62 [-1.38, +2.63]
$F - (R + A)$	-1.20 [-2.89, +0.43]	+0.16 [-2.06, +2.42]
$(P + A) - P$	+5.91 [+3.52, +8.38]	-2.00 [-4.34, +0.30]
$(R + A) - R$	+4.05 [+2.13, +6.04]	-1.96 [-4.12, +0.15]
$F - (P + A)$	-0.05 [-1.91, +1.79]	+0.82 [-1.60, +3.23]
$F - S$	-0.80 [-2.67, +1.02]	+0.07 [-1.86, +2.04]
$[(R + A) - R] - [(P + A) - P]$	-1.86 [-4.31, +0.60]	+0.04 [-3.03, +3.10]

Table 47: Repair and append increments under the three archived scoring interfaces. All intervals here are 95%. Primary adds the leading-polarity transform after residual-tag cleanup. The same responses are rescored in each row.

Contrast	Interface	Executor A	DeepSeek
$F - P$	Raw	+22.62 [+19.46, +25.92]	-1.18 [-3.40, +1.05]
	Tag cleanup	+6.72 [+4.09, +9.38]	-1.18 [-3.40, +1.05]
	Primary	+5.86 [+3.33, +8.48]	-1.18 [-3.40, +1.05]
$(P + A) - P$	Raw	+21.11 [+17.91, +24.35]	-2.00 [-4.34, +0.30]
	Tag cleanup	+6.77 [+4.30, +9.29]	-2.00 [-4.34, +0.30]
	Primary	+5.91 [+3.52, +8.38]	-2.00 [-4.34, +0.30]
$(R + A) - R$	Raw	+19.99 [+17.03, +22.98]	-1.96 [-4.12, +0.15]
	Tag cleanup	+4.73 [+2.73, +6.76]	-1.96 [-4.12, +0.15]
	Primary	+4.05 [+2.13, +6.04]	-1.96 [-4.12, +0.15]
$(I + A) - I$	Raw	+24.55 [+21.42, +27.76]	-1.95 [-4.14, +0.21]
	Tag cleanup	+3.47 [+1.47, +5.51]	-1.95 [-4.14, +0.21]
	Primary	+2.50 [+0.57, +4.44]	-1.95 [-4.14, +0.21]

Q.3 COMPLETE CHANGED PASSAGES

The following text reproduces the targeted body, its replacement, and both applied appends. References to gold answers are wording generated in the skill; held-out reference answers are not supplied to the executor.

Withdrawn passage H

> ## Multi-Hop Reasoning First identify the attribute requested for the target entity. If the question asks about a shared job between two people, determine the job of each from their respective documents, then find the intersection. Avoid adding extra words like 'both' or rephrasing the attribute. The final answer should be the exact string that appears in the context for the common attribute, not a descriptive paraphrase.

Replacement passage B

> ## Multi-Hop Reasoning First identify the question type and the exact information requested:
> – Shared attribute: determine the attribute for each entity from its document, then output the exact intersecting string as it appears in context.
> – Comparison / 'who would you' / 'which is more': identify the distinguishing fact for each entity and output only the named entity or value that answers the question, not a description.
> – Yes/No (e.g., 'Are X and Y both...?'): answer with 'Yes' or 'No' based on evidence; do not list attributes.
> – Event location or co-creator: trace the relation across documents and output the exact phrase from context that fills the requested slot. Avoid adding extra words like 'both', 'and', or rephrasing. The final answer should be the exact string from context that satisfies the asked relation.

Appended passage A1

> ## Output Discipline The final answer placed inside <answer>...</answer> must contain ONLY the answer text — no tags, no 'The answer is', no explanations. Match the gold answer type: a name, place, 'Yes'/'No', or short phrase exactly as it appears in the source. Always re-read the question to confirm the answer type before writing the tags.

Appended passage A2

> ## Entity Resolution When a question describes an entity by attributes (e.g., 'actress who began career in 2006', 'company that made CP-400'), first locate the entity in the context whose documented attributes match ALL described traits. Confirm the match across documents before using that entity in further reasoning. Do not answer based on the descriptive phrase alone; resolve it to a named entity from the context.

R RESPONSE MEANING AND COMBINED CORRECTIONS

This appendix analyzes answers and combination records from the initial 512-question panel. These annotations interpret the initial observations; Appendix L gives the repeated-execution design.

R.1 ANNOTATING THE EXISTING RESPONSES

The natural-repair study contains ten text conditions, two execution configurations, and 512 questions. We analyze the existing 10,240 condition–question records. Identical answers to the same question share an annotation, yielding 1,310 unique question–answer pairs. DeepSeek Flash receives anonymous answers and the supplied question and evidence, without text-condition identifiers, scores, or the interaction hypothesis. A separate task-level pass first identifies the requested answer type, the supporting evidence, and any ambiguity in the question. Answer labels distinguish correct, partial, incorrect, nonresponsive, abstaining, and uncertain responses. Partial answers remain a separate categorical label.

We use a common set of 429 questions for which the task assessment is supported without material uncertainty and all ten conditions under both configurations have determinate answer labels. Requiring

no task uncertainty at all leaves 367 questions. These restrictions concern the interpretation of responses: the original score comparisons continue to use all 512 questions. The other questions are neither imputed nor recoded as incorrect. Table 48 reports the full common-set distribution; its counts describe the annotations, rather than a human-gold semantic accuracy benchmark.

Table 48: Answer-status labels on the common 429 questions. Text symbols follow Appendix Q: B is the replacement body and A contains the two appended paragraphs.

Configuration	Text	Correct	Partial	Incorrect	Nonresponsive	Abstain
Executor A	N	415	2	12	0	0
	P	375	8	21	25	0
	R	385	6	25	13	0
	$P{+}A$	397	3	26	3	0
	$R{+}A$	403	3	20	3	0
	$R{+}B$	393	3	32	1	0
	F	396	1	30	2	0
	S	403	1	24	1	0
	I	387	11	23	8	0
	$I{+}A$	400	4	23	2	0
DeepSeek	N	408	6	13	1	1
	P	409	2	14	2	2
	R	412	6	9	2	0
	$P{+}A$	419	4	4	1	1
	$R{+}A$	410	3	13	2	1
	$R{+}B$	407	4	16	2	0
	F	408	4	15	2	0
	S	403	6	19	1	0
	I	411	6	10	2	0
	$I{+}A$	414	5	9	1	0

The Executor A parent-to-full comparison contains 29 transitions into the correct category and eight out of it. Seventeen of the former were initially labeled nonresponsive. This motivates a second, narrower annotation: an answer can communicate a yes/no commitment without using the requested yes/no form. The task-level pass identifies 114 binary-verification questions, of which 98 belong to the common set. A dedicated pass labels the context-supported polarity, the answer's commitment (yes, no, or undetermined), and whether the answer explicitly uses a yes/no form. It covers 366 unique answers. Agreement with the context polarity concerns that commitment; it does not certify every additional assertion in the answer.

Among the seventeen Executor A parent answers initially labeled nonresponsive, the dedicated pass finds fifteen commitments aligned with the context and two undetermined answers. An additional review using only the question and answer, without the supplied evidence, agrees on twelve aligned commitments. Thus the transition into a correct answer category does not uniformly represent a changed factual judgment. This distinction also explains why explicit forms and polarity agreement can move in different directions in Table 49.

R.2 Checks on the annotations

A separate review of 46 unique answers to sixteen questions agrees on the full status label for 35 answers and on the correct-versus-other distinction for 42. Reversing answer presentation order on 32 questions and 102 unique answers preserves 100 full status labels and 101 correct-versus-other labels. For the targeted commitment analysis, a further review of 35 answers to 26 questions agrees on 27 commitments. Reversing the order of 36 answers to twelve binary questions preserves all 36 commitment and form labels, as well as the twelve context-polarity labels. These selected checks expose both agreement and ambiguity; they are not estimates of population annotation accuracy. Order checks concern the judge, not repeated executions of the evaluated agent.

Table 49: Expression and logical commitment on the same 98 binary questions. Alignment is with the context-polarity annotation; explicit form and aligned commitment are different dimensions.

Configuration	Text	Explicit yes/no	Aligned	Opposed	Undetermined
Executor A	N	98	93	5	0
Executor A	P	68	91	3	4
Executor A	R	83	91	6	1
Executor A	$P + A$	97	91	7	0
Executor A	$R + A$	97	93	5	0
Executor A	$R + B$	98	88	10	0
Executor A	F	98	87	11	0
Executor A	S	98	91	7	0
Executor A	I	87	89	7	2
Executor A	$I + A$	97	91	7	0
DeepSeek	N	98	94	3	1
DeepSeek	P	95	92	5	1
DeepSeek	R	94	95	3	0
DeepSeek	$P + A$	97	96	2	0
DeepSeek	$R + A$	97	93	5	0
DeepSeek	$R + B$	98	93	5	0
DeepSeek	F	98	93	5	0
DeepSeek	S	98	92	6	0
DeepSeek	I	93	95	3	0
DeepSeek	$I + A$	93	94	3	1

R.3 Four text conditions and complete match patterns

The focal combination comparison uses R, $R + B$, $R + A$, and $F = R + B + A$. Its score interaction is

$$J = v(F) - v(R + B) - v(R + A) + v(R). \tag{R.26}$$

The Executor A F1 interaction is -2.80 points, with an exploratory paired 95% interval of $[-5.25, -0.46]$; the DeepSeek interaction is $+0.49$ points, $[-2.31, 3.28]$. Intervals use 20,000 task draws, paired across conditions and stratified by the original question categories. They summarize variation over these recorded task responses.

Table 50: All sixteen exact-match patterns, in the order remainder, replacement only, additions only, and both. Each model column sums to 512. The last column is the per-question interaction on the 0–1 EM scale.

Pattern	Executor A questions	DeepSeek questions	EM interaction
0000	151	137	+0
0001	3	2	+1
0010	5	4	-1
0011	6	1	+0
0100	4	4	-1
0101	4	5	+0
0110	1	1	-2
0111	18	5	-1
1000	3	8	+1
1001	0	0	+2
1010	4	8	+0
1011	10	6	+1
1100	2	3	+0
1101	2	13	+1
1110	4	8	-1
1111	295	307	+0

The eighteen Executor A questions with pattern 0111 have no exact match under the remainder, but match under both single additions and their combination. Their total contribution is -2.94

F1 points on the original 512-question denominator. The other 494 questions contribute $+0.14$, producing the overall -2.80. Twelve of these eighteen questions are binary and six are nonbinary. This decomposition locates the negative interaction primarily in overlapping match gains, rather than treating every negative contribution as evidence that combining text destroyed a success.

The distinction is concrete in the following responses. For Sun Yat-sen's military rank, the remainder returns "dàyuánshuài (grand marshal)", while both single additions and their combination return "dàyuánshuài". All four identify the rank, but the F1 scores are 50, 100, 100, and 100. For whether *Lesquerella* and *Thujopsis* are both conifer genera, the remainder answers "Yes" and the other three answer "No", a change in the requested judgment. Conversely, when asked who died younger, George Gershwin or Léo Delibes, the remainder and combination choose Delibes while each single addition chooses Gershwin. All four fail exact match against "George Jacob Gershwin", whereas their F1 scores are 0, 80, 80, and 0. The score pattern and the answer interpretation therefore provide complementary information.

S Frozen Instructions in Multi-step Automation

The study combines a broad single-execution panel with repeated execution on an input-selected panel and fixed diagnostic sets. This section gives the broad panel and its text constructions; Appendix L specifies repeated sampling and estimation.

S.1 Texts, tasks, and execution

We evaluate one historical AutomationBench revision in five frozen text conditions: no skill (N), the parent (P), the actual added text alone (S), a document-end append ($A = P + S$), and the historical full revision (F). These texts are independent of the QA studies and are not newly selected on the 72 evaluation tasks. The parent has 368 characters and the new text 1,594. The full revision has 1,965 literal characters, so the new text constitutes 81.2% of this document; it is not a short-section compression result.

The parent asks the agent to use real tool responses, preserve unrelated state, and report checks. The update adds three sections about locating authoritative sources, avoiding fabricated identifiers, and handling unavailable services or errors. The historical revision inserts the first two before the parent's Reporting section and the last after it. The append condition places all three at the parent's end. All parent rules remain present in both documents. Thus F and A compare organization and position of the same instruction content, including their separating whitespace.

The evaluation uses twelve tasks from each of six domains: sales, marketing, operations, support, finance, and human resources. Both configurations execute all five texts on all 72 tasks, yielding 720 trajectories. Every condition for a task starts from the same world state and assertion set. DeepSeek Flash uses the V4.1 configuration with its default thinking setting; DeepSeek V4 Pro uses high thinking. Each run permits 50 interaction steps, 16,384 output tokens per response, and 3,600 seconds per episode. The requested temperature is zero; inference behavior under the thinking setting is provider controlled. These are two execution configurations rather than a model-size-only comparison.

The benchmark's strict task-success signal and its implementation's partial-credit diagnostic evaluate the terminal world. Partial credit is the fraction of eligible assertions passed, computed separately for each trajectory and then averaged over tasks. Eligibility excludes, by default, requirements already satisfied initially and still satisfied at the end; an initially satisfied requirement that is broken counts as a failure. Explicit exclusion flags override this rule. An empty eligible set scores zero. Strict success is one when partial credit equals one. Fixed-assertion comparisons below instead retain every recorded requirement's passed flag. Each distinct text–task input is executed once. The paired task bootstrap uses 20,000 draws, stratified by domain, with shared weights across conditions and configurations.

The direct Pro-minus-Flash difference for the four-condition interaction is $+6.92$ points, $[-8.60, 22.71]$. For $F - A$ it is -2.75, $[-10.82, 5.20]$. Neither comparison establishes a configuration difference. Table 52 gives all six domain means under the study's original task mixture.

Table 51: All five deployment contrasts, in percentage points. These are marginal descriptive intervals. The original three primary comparisons use 98.3333% intervals, which also include zero.

Partial-credit contrast	Flash [95% CI]	Pro [95% CI]
$F - P$	$-2.57\ [-9.89, 4.68]$	$+0.72\ [-4.98, 6.52]$
$S - N$	$-1.17\ [-9.36, 6.85]$	$-2.06\ [-7.33, 2.68]$
$A - P$	$-4.76\ [-13.21, 3.90]$	$+1.28\ [-5.29, 7.74]$
$A - P - S + N$	$-3.59\ [-15.52, 8.07]$	$+3.33\ [-4.53, 11.74]$
$F - A$	$+2.20\ [-4.81, 9.21]$	$-0.56\ [-4.88, 3.40]$

Table 52: Complete domain means: official partial credit / strict success, in percent, with twelve tasks per domain. These small domain panels are descriptive.

Config.	Domain	N	P	S	A	F
Flash	Sales	76.16 / 50.00	63.38 / 33.33	57.41 / 25.00	66.44 / 50.00	56.71 / 25.00
Flash	Marketing	61.65 / 41.67	76.39 / 58.33	60.22 / 33.33	65.25 / 33.33	75.84 / 50.00
Flash	Operations	76.48 / 58.33	80.69 / 66.67	71.67 / 50.00	79.03 / 58.33	72.48 / 50.00
Flash	Support	59.24 / 16.67	60.04 / 25.00	62.43 / 16.67	58.27 / 16.67	60.88 / 16.67
Flash	Finance	58.49 / 25.00	66.82 / 41.67	79.51 / 58.33	77.93 / 41.67	73.96 / 41.67
Flash	HR	34.82 / 25.00	45.63 / 16.67	28.57 / 25.00	17.46 / 8.33	37.70 / 16.67
Pro	Sales	60.46 / 33.33	55.23 / 41.67	65.51 / 33.33	61.99 / 33.33	63.52 / 33.33
Pro	Marketing	79.40 / 50.00	72.37 / 25.00	75.44 / 41.67	71.22 / 25.00	71.93 / 25.00
Pro	Operations	70.28 / 50.00	64.05 / 25.00	70.86 / 41.67	71.81 / 41.67	67.18 / 33.33
Pro	Support	70.26 / 8.33	69.21 / 16.67	71.14 / 16.67	66.16 / 16.67	69.03 / 16.67
Pro	Finance	73.16 / 25.00	62.65 / 25.00	72.57 / 41.67	75.84 / 33.33	74.80 / 50.00
Pro	HR	21.83 / 16.67	23.39 / 0.00	7.54 / 0.00	7.54 / 0.00	4.76 / 0.00

S.2 Combined outcomes and initial conditions

Table 53 reports every strict-success pattern in the order N, P, S, A. Among tasks passed by both P and S separately, A fails six of 21 on Flash and two of fourteen on Pro. Four Flash cases are newly passed by both individual texts relative to N; the other two Flash cases and both Pro cases already pass under N. The reverse pattern, success only under the combination, occurs once on Flash and twice on Pro. These are observed matches across the four recorded executions, not estimates of the repeatable probability of an instruction conflict.

Table 53: All sixteen AutomationBench strict-success patterns in the order no skill, parent, new text, and parent plus new text. Each model column sums to 72.

Pattern	Flash tasks	Pro tasks	Strict interaction
0000	36	42	+0
0001	1	2	+1
0010	1	1	-1
0011	1	1	+0
0100	1	2	-1
0101	0	0	+0
0110	4	0	-2
0111	2	2	-1
1000	2	5	+1
1001	0	0	+2
1010	0	2	+0
1011	2	3	+1
1100	1	0	+0
1101	6	0	+1
1110	2	2	-1
1111	13	10	+0

In the illustrated blog-refresh task, the email gives traffic and staleness criteria. A Slack instruction additionally requires a priority score greater than 60, exempts evergreen-premium posts, and specifies

the reporting code `BLGRFSH-2026-Q1` in place of `BLOGR-519-Q1`. The parent and new-text deployments each flag three posts and use the designated code. Their concatenation also flags "Support Ticket Resolution Guide", whose priority score is 52, and reports the superseded code. The selected row's final Status is empty under each separate text and Refresh under their concatenation. The final-state assertions record four failures for the concatenation: the extra row, absence of the designated code, presence of the superseded code, and inclusion of the extra title in the channel report. This example is selected to interpret one of the six recorded common-pass losses; it does not estimate their mechanisms or frequency beyond the full pattern table.

All trajectories also contain the fixed assertion outcomes before and after execution. Across tasks there are 1,254 assertions, of which 548 are initially satisfied. Flash loses 20, 28, 26, 26, and 29 of those initially satisfied conditions under N, P, F, S, A, respectively; Pro loses 40, 48, 49, 36, and 45. Eight initially satisfied conditions on Flash and ten on Pro survive both P and S but not A. Assertions are clustered within tasks. A lost initial condition is not automatically an unnecessary side effect: interpreting it requires the task's intended changes.

For F versus A, Flash passes exactly 988 fixed assertions under each text, but 132 individual outcomes differ: 66 pass only under F and 66 only under A, across 25 tasks. Pro passes 985 and 989, with 41 versus 45 exclusive passes across 24 tasks. The official partial-credit metric uses task weighting and eligible-assertion denominators; it is not the fraction of these 1,254 fixed assertions. Eighteen tasks in each configuration have eligibility masks that differ across the five executions. Keeping fixed pass outcomes separate from eligibility prevents a changed scoring denominator from being mistaken for a changed assertion outcome.

S.3 The frozen instruction content

The following text gives the parent and its added sections. The insertion distinction described above determines F and A; S consists only of the three added sections.

Parent instructions

> Execute the requested task precisely using the available API tools.
> Identify the relevant records and discover endpoint syntax with api_search when needed.
> Perform only the requested mutations, preserve unrelated state, and do not invent identifiers or values.
> ## Before finalizing
> ## Reporting
> Finish with a concise factual summary after the tool work is complete.

Added instructions

> ### Locate authoritative data first
> - Before any mutation, find the records the task actually references. Read the task text for explicit IDs/names, and check the available context channels (inbox messages, Slack history, drafts) for procedures, codes, restrictions, or pointers to the source data.
> - Use ‘api_search‘ with targeted service + entity keywords (e.g. "docusign templates", "sheets values"), and stop discovering once the needed endpoint is confirmed. Repeated broad searches that return unrelated services are a signal to switch tactics, not to keep paging through results.
> ### Do not fabricate resources or identifiers
> - Never guess or brute-force record IDs, spreadsheet IDs, or names as a way to "probe" for data; every create call is a real side effect and every incorrect ID call wastes steps.
> - Do not create placeholder spreadsheets/files/drafts to substitute for source data you could not locate.
> - If the required source data or credentials genuinely cannot be found after reasonable, targeted attempts, stop mutating and finish with a report of the exact blocker and what was verified.
> ### Handling unavailable services/errors
> - When an API reports no connected account/credentials, a missing handler, or a not-found resource, treat it as actionable information: pivot to the services that are confirmed available rather than retrying the same failing call.
> - Keep tool calls to what the task needs; excessive probing is itself a failure mode. When a

blocker prevents completing the requested mutation, state clearly in the final summary what was and was not completed.

T Combining Guidance: Overlap, Loss, and Execution Variation

The same accounting applies when two passages are combined. A lower marginal gain can arise because the passages solve the same cases, because the combination loses successes, or because executions vary. These explanations require different comparisons.

Overlap can produce a negative interaction without destroying successes. For receiver R and additions A, B, define the four-condition interaction

$$J = V(R + B + A) - V(R + A) - V(R + B) + V(R). \tag{T.27}$$

For a recorded binary response table, replace V with v_z to obtain the corresponding recorded interaction. If an input fails under R but succeeds under either addition separately and their combination, its contribution to that interaction is negative: the separate effects count the same gain twice. Proposition J.3 separates gained, lost, and overlapping successes. If the combination retains every separate success and both separate additions retain every baseline success, the recorded binary interaction equals combination-only gains minus gains shared by the two additions. A negative value under these conditions measures overlap rather than destruction.

The natural repair supplies a concrete example. In the initial Executor A panel, eighteen questions lack an exact match under R but match under $R + A$, $R + B$, and their combination. Those cells contribute -2.94 F1 points to the aggregate -2.80 interaction; all other cells contribute $+0.14$. The combined text preserves the eighteen successes. On fresh executions of these fixed questions, its mean EM is 100%, compared with 95.83% and 91.67% for the separate additions. This diagnoses the historical pattern rather than establishing a population-wide interaction: on the repeated 128-question panel, $J = -2.13$ with a 95% interval of $[-5.42, 0.88]$ (Appendix M).

Changed trajectories need a same-text reference. Multistep tool use provides a harder setting in which to calibrate individual failures. We use 24 tasks from six domains in AutomationBench (Shepard & Salimans, 2026), with four fresh trajectories per condition. Two conditions concatenate or interleave the same parent and added passages. We compare disagreement on all recorded assertion passed flags, first averaging within each task and then across tasks. This fixed assertion set differs from the outcome-dependent eligibility used for partial-credit scoring (Appendix S). Permutations retain the whole trajectory; the statistic concerns marginal assertion outcomes rather than the full joint distribution of agent behavior.

Table 54: Automation placement changes relative to repeated execution. The same-text reference averages the two within-placement disagreement rates. Holm correction covers the two configurations. Neither observed excess is established. Excess is computed before rounding.

Executor	Cross-text	Same-text	Excess	Adjusted p
Flash	14.71%	14.29%	0.42 pp	0.662
Pro	9.32%	9.29%	0.02 pp	0.662

The apparent placement differences are close to the same-text reference (Table 54). Unlike the extracted QA sections, these placements are not distinguished by the repeated-execution test. This does not establish equivalence: the small task panel also gives imprecise utility estimates. Appending the additions to the parent changes strict success by $+4.17$ points for Flash and $+1.04$ for Pro, with 98.75% intervals of $[-9.38, 17.71]$ and $[-6.25, 8.33]$. The estimated 80%-power thresholds for the four combination contrasts range from 12.7 to 23.1 points (Appendix M.5). Individual changed failures consequently cannot establish a small average combination cost. The panel illustrates the practical boundary of the audit: repeated execution can reveal when an appealing interference story is not resolved by the available evidence.