IDENTIFYING RECURSIVE SELF-IMPROVEMENT: FIVE GATES FOR SYNTHETIC-DATA TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

A score gain over a trained baseline does not by itself identify recursive self-improvement: the comparator may have degraded, received different exposure, or been evaluated through a defective or contaminated pipeline. We formalize this distinction as a *claim-conditional identification contract*: every claimed effect must name its estimand, the contrast that estimates it, the factors held fixed, and the integrity checks that make that contrast interpretable. The contract yields five necessary gates—untouched-base comparison, exposure matching, verifier validation, controller activation, and decontamination—whose relevance depends on the claim being made. We apply it to DATAEVOLVE, a protocol-frozen, three-round generator–verifier–trainer study on a sealed 2,070-item logic benchmark. The full loop exceeds the strongest declared trained baseline by 10.26 percentage points, yet remains 4.12 points below its untouched starting model: the observed gain is real, but its self-improvement interpretation is not identified. A nominal 14.55-point verifier contrast is likewise confounded; under corrected and matched conditions, verified and unverified arms score 0.6680 and 0.6681. Rebuilding a positive control with instance-disjoint data reduces it from 0.6834 to 0.6678, near the 0.6671 base. An audit-derived mixture pattern is directionally consistent across exploratory seeds, but a pre-specified fresh-holdout replication gains only 0.7 points, below its two-point minimal effect. The five-gate contract turns performance differences into auditable scientific claims and provides a concrete reporting standard for recursive training.

1 INTRODUCTION

Recursive synthetic-data training promises a renewable source of supervision: a model generates problems or responses, a verifier or judge selects useful examples, and the accepted data train the next model (Wang et al., 2023; Chen et al., 2024; Yuan et al., 2024; Huang et al., 2026a; Zhao et al., 2026). The central scientific question, however, is not whether the final model beats *some* trained comparator. It is whether the evidence identifies improvement over the starting model, an effect of recursive data rather than exposure, or a contribution from the proposed verifier or controller. These are different claims with different estimands and different required controls.

This distinction matters because a correct score difference can support the wrong conclusion. A recursive arm may beat a trained baseline because the baseline forgot more, saw fewer tokens, used a different target format, or was trained on a contaminated corpus. A verifier ablation may change both filtering and optimization budget. An adaptive system may appear successful while its controller never changes state. Without a claim-specific identification contract, these asymmetries are easily compressed into a single statement that "the loop improves."

We introduce such a contract. For each scientific claim, it records the estimand, treatment and control, factors that must remain fixed, integrity checks, and evidence status. Applied to recursive training, it produces five necessary gates: compare against the untouched starting model; match training exposure; validate the verifier under the tested regime; demonstrate controller activation before attributing gains to adaptation; and remove evaluation overlap from every training arm. The gates are not claimed to be jointly sufficient for causal identification. Their role is sharper: each blocks a specific, common route from an observed gain to an unsupported mechanism claim.

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.

Table 1: Controls reported for closely related recursive-training systems. Symbols describe public evidence for the corresponding claim: ✓ adequate, ∼ partial, × audited but violated, ∘ not reported (unknown), and — not applicable. "Case v1" denotes the protocol-frozen experiment before post-hoc repairs. A missing or inapplicable gate is not a verdict on an entire system; gates are evaluated conditional on the estimand claimed.

Work	Rounds	Base	Exposure	Verifier	Controller	Disjoint
Case v1	3	✓	×	×	×	×
KITE	5	✓	✓	∘	∼	∼
Skill Self-Play	5	✓	∼	∘	✓	∘
R-Zero	3	✓	∼	∼	✓	∘
Absolute Zero	n.d.	✓	∘	∘	∼	∘
BigBang-v1	n.d.	✓	∘	∼	×	∘
CPMobius	n.d.	✓	∼	∘	✓	∼

We demonstrate the contract through DATAEVOLVE, a pre-specified three-round generator–verifier–trainer study on checker-attached logic reasoning. The protocol includes trained data controls, controller-policy controls, a no-verifier arm, three independent training seeds, and a sealed 2,070-item evaluation set. Its main result is an identification reversal. The full loop reaches 62.59%, 10.26 points above the strongest declared trained baseline, but 4.12 points below the untouched model. The same experiment also produces an apparent 14.55-point advantage over no verification, yet post-run inspection shows that checker validity, target format, controller activation, and exposure were not isolated.

The paper makes two primary contributions and one bounded secondary observation:

- **A claim-conditional identification contract.** We map four common recursive-training claims to explicit estimands and minimum controls, then instantiate five practical gates that distinguish absolute improvement, recursive-data value, verifier value, controller value, and clean generalization.
- **An attribution audit that changes the interpretation of the same result.** The protocol-frozen case study juxtaposes the trained-baseline gain with the untouched-base regression, traces why the original verifier contrast is non-identifying, and shows that matched repair and decontamination remove the apparent advantages in observed means.
- **An audit-derived boundary hypothesis.** Exploratory evidence suggests that verifier value depends on available error mass and that disjoint data mixtures may dilute family-level interference. A fresh, pre-specified holdout does not confirm a practically meaningful mixture effect, thereby fixing the boundary of this secondary claim.

The resulting contribution is methodological rather than a new state-of-the-art training loop. It provides authors and reviewers with a compact way to ask what a reported gain estimates, which alternatives remain open, and which additional comparison would close them. A desk audit of public recursive-training systems further shows that, for each major claim examined, at least one claim-relevant identifying control is absent or unclear in the reported evidence; unreported properties are treated as unknown, not as failures.

2 RECURSIVE TRAINING AND ATTRIBUTION

Recursive synthetic-data training. One-shot pipelines such as Self-Instruct bootstrap instruction-following data (Wang et al., 2023), while Evol-Instruct broadens and deepens seed instructions (Xu et al., 2024). Recent systems close the loop between a policy and its training distribution: SPIN trains against earlier generations (Chen et al., 2024); self-rewarding language models use model-generated judgments (Yuan et al., 2024); R-Zero, Absolute Zero, and CPMobius couple task proposal, solving, and verifiable feedback (Huang et al., 2026a; Zhao et al., 2026; Li et al., 2026); and Skill Self-Play, BigBang-v1, and KITE scale guided generation or failure-aware curation (Huang et al., 2026b; The BigBang Team, 2026; Luo et al., 2026). These works establish that recursive pipelines can improve downstream scores. Our question is orthogonal: which comparison identifies improvement over the starting model or isolates the mechanism named in the claim? A contemporaneous preprint similarly audits self-improvement against a pre-specified measured null (Xu et al., 2026).

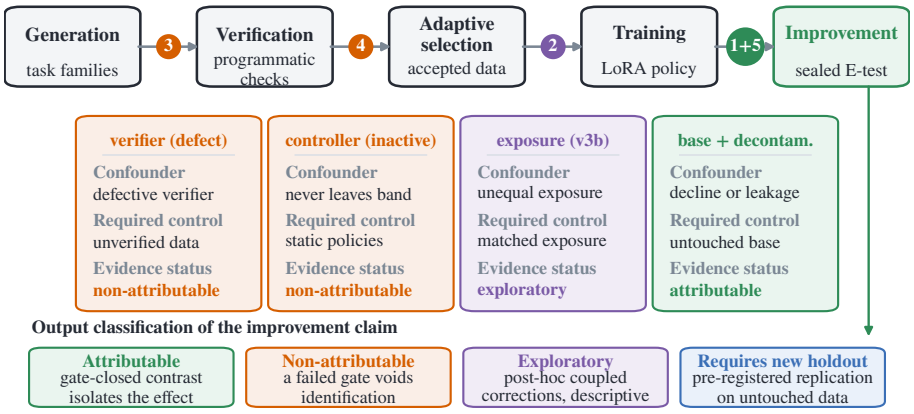

Figure 1: From reported gain to identified claim. Arrows show confounders, minimum controls, and as-run evidence status (v1 unless v3b); attributable means identified, not positive, gain.

Mechanisms and evaluation integrity. Programmatic verification can provide dense supervision (Wang et al., 2024), whereas learned judges introduce position, style, and verbosity biases (Zheng et al., 2023). Reusing generated data can erode distribution tails or alter scaling (Shumailov et al., 2024; Dohmatob et al., 2024); sequential fine-tuning can forget prior knowledge (McCloskey & Cohen, 1989), motivating regularization, replay, and rehearsal (Kirkpatrick et al., 2017; Rolnick et al., 2019; Huang et al., 2024a). Contamination detection ranges from benchmark-level measurement and guided-completion probes to exchangeability tests (Sainz et al., 2023; Golchin & Surdeanu, 2024; Oren et al., 2024), with encrypted test release proposed as mitigation (Jacovi et al., 2023). Format and option-position changes can also alter measured performance (Sclar et al., 2024; Zheng et al., 2024). These literatures motivate individual controls, but do not by themselves specify which combination is required for a particular recursive-training claim. Our contribution is the mapping from claim to estimand, minimum contrast, invariances, and integrity checks.

Prospective evidence. Because diagnostic repairs are naturally informed by observed failures, their evidential role differs from that of a frozen test. We retain the protocol-frozen endpoint, label every result obtained after inspecting it as exploratory, and use a new holdout only for a separately pre-specified replication, following prospective protocol specification and Registered Reports (Nosek et al., 2018; Chambers & Tzavella, 2022).

3 Identification Contract and Case Study

3.1 A claim-conditional identification contract

Let $Y(M, D_{\text{eval}})$ denote the outcome of model M on evaluation data D_{eval}, and let M_0 be the untouched starting model. We represent a scientific claim by a contract

$$\mathcal{C} = (\text{claim},\ \text{estimand},\ \text{contrast},\ \text{invariances},\ \text{integrity checks},\ \text{status}).$$

The contract prevents a result from silently changing meaning: an observed difference is evidence for a claim only when its contrast estimates the named quantity, the required factors are held fixed, and the relevant integrity checks pass. Evidence status records whether the contrast was pre-specified, post-hoc, or evaluated on a fresh holdout. This construction differs from a generic checklist because gates are activated by the claim. A study need not include an adaptive-controller test if it makes no controller claim; once it attributes a gain to adaptation, controller activation and a static-policy contrast become necessary.

Common recursive-training claims imply four primary estimands: the training effect $\tau_{\text{train}} = Y(M_{\text{loop}}) - Y(M_0)$; the recursive-data effect $\tau_{\text{data}} = Y(M_{\text{loop}}) - Y(M_{\text{static}})$ under matched exposure; and the verifier and controller marginals $\tau_{\text{ver}} = Y(M_{\text{verified}}) - Y(M_{\text{unverified}})$ and $\tau_{\text{ctrl}} =$

Table 2: Claim-conditional experiment contract. Each row specifies the minimum contrast and the principal alternative explanation it blocks.

Claim	Estimand	Required contrast	Fixed or checked	Open alternative
Training improves base	τ_{train}	trained model − untouched base	decoding, evaluation	comparator degradation
Recursion adds value	τ_{data}	recursive − matched static data	exposure, format	token/update budget
Verification adds value	τ_{ver}	verified − matched unverified data	candidate pool, exposure	treatment misdefinition
Adaptive control adds value	τ_{ctrl}	adaptive − static policy	exposure, data budget	inactive controller

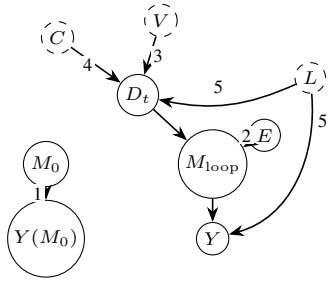

Figure 2: Identification graph for recursive-training claims. Numbered paths correspond to (1) untouched base, (2) matched exposure, (3) verifier validity, (4) controller activation, and (5) decontamination. Dashed inputs remain unobserved unless the corresponding gate makes them auditable.

$Y(M_{\text{adaptive}}) - Y(M_{\text{static-policy}})$, each holding the remaining factors fixed. Clean generalization additionally requires that D_{eval} be instance-disjoint from every training arm.

3.2 Five necessary gates

The contract yields five operational gates. **G1, untouched base:** evaluate M_0 with the same renderer and decoding; otherwise superiority over a trained baseline is compatible with absolute regression. **G2, matched exposure:** equalize examples, tokens, updates, target format, and optimization budget for data-source claims. **G3, verifier validity:** regression-test acceptance logic and compare verified with unverified data from the same candidate pool; otherwise the nominal treatment is undefined or coupled to exposure. **G4, controller activation:** log state and demonstrate that adaptive decisions differ from a fixed policy before attributing an effect to control. **G5, decontamination:** screen every training arm against evaluation instances and report the unit of overlap; otherwise the benchmark no longer estimates clean generalization. These gates are necessary relative to their estimands, not jointly sufficient for a unique causal mechanism.

3.3 Protocol-frozen case study

DataEvolve operates on checker-attached logic tasks instantiated from parameterized families. At each of three rounds, a generator samples task parameters under controller state Ω_t; the solver produces a response; a task-specific programmatic verifier decides whether the pair enters D_t; and a LoRA policy is trained on accepted pairs. The controller was intended to adjust family diversity, constraint complexity, and solution horizon using per-family development performance. Random-dimension and single-dimension policies test the adaptation claim; static, random-synthetic, and no-verifier arms test data and verification claims.

The trained policy is Qwen3-8B (Yang et al., 2025) with rank-16 LoRA adapters (Hu et al., 2021). Trained conditions use three independent seeds. The confirmatory endpoint is exact-match correctness on a sealed 2,070-item set, E_{test}, fixed before training with fingerprint `ab30f602829b11bf`; development data were used for pipeline checks. The pre-specified primary contrast compared final-

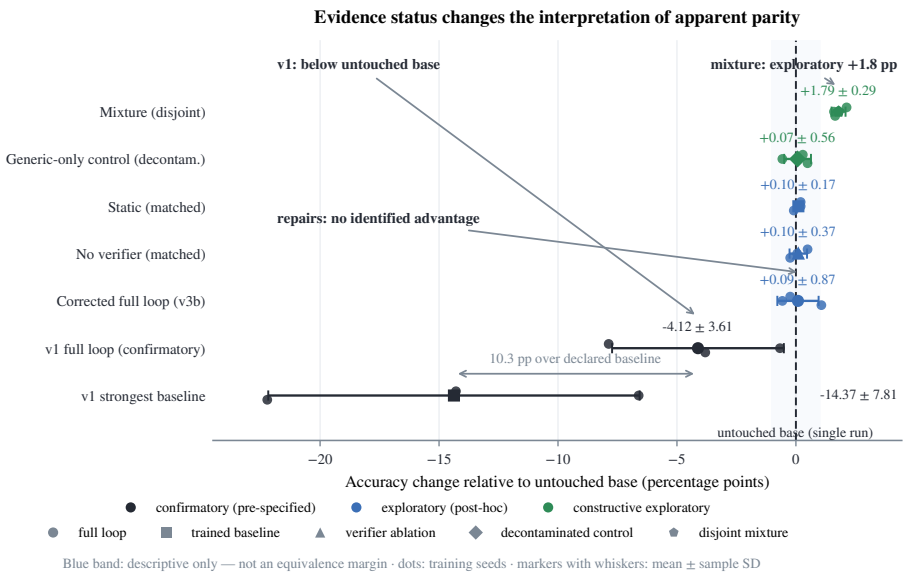

Figure 3: Evidence relative to the untouched base for seven decisive conditions. Black denotes protocol-frozen evidence, blue post-hoc diagnostics, and green constructive exploratory evidence. Dots are independent training seeds; whiskers show mean ± sample SD. The shaded ±1-point band is descriptive, not an equivalence margin.

round full-loop accuracy with the strongest declared trained baseline. Importantly, the later five-gate synthesis was produced by auditing this protocol-frozen experiment; it was not itself pre-registered before v1.

3.4 Audit and evidence status

After opening E_{test}, code and data inspection identified a case-normalization error in one checker, truncated training targets, negligible controller movement, unequal exposure, and evaluation-prompt overlap in the benchmark-derived positive control (Appendix Table 10). These findings invalidate specific interpretations of v1 without estimating additive per-defect effects. All repairs that reuse E_{test} are therefore labeled post-hoc exploratory. Corrected experiments use full closed traces, matched exposure, an audited checker, and instance-disjoint controls; a no-verifier arm differs only in correctness filtering. A separate 1,100-item untouched holdout with five fixed seeds and a two-point minimal meaningful effect provides the only confirmatory test of the audit-derived mixture hypothesis. This separation lets the original experiment falsify its intended interpretation, repaired analyses localize the open gates, and the fresh holdout test a new claim without retroactively upgrading post-hoc evidence.

4 Results: An Identification Reversal

The sealed evaluation contains 2,070 items: 700 in-distribution generator items, 570 held-out combinations, and 800 benchmark items. Accuracy is macro-averaged over items; reported ± values are sample standard deviations over independent training seeds, not confidence intervals. The untouched base is one deterministic greedy evaluation. Figure 3 summarizes the decisive conditions; Appendix Figure 8 reports the complete condition matrix and per-source results.

4.1 Trained-baseline superiority does not identify self-improvement

The full loop attains 0.6259 ± 0.0361, exceeding the strongest declared trained baseline (0.5234 ± 0.0781) by 10.26 percentage points. The same loop remains 4.12 points below the untouched base (0.6671), and each of its three seeds is below that starting checkpoint. It also does not outperform the

Table 3: The interpretation reversal and the controls that survive the audit. "Post-hoc" denotes analyses designed after inspecting the original sealed endpoint; close means are not equivalence claims.

Evidence	Contrast	Difference	Status and implication
v1 primary	loop − trained baseline	$+10.26$ pp	confirmatory; comparator-relative gain
v1 absolute	loop − untouched base	-4.12 pp	confirmatory; no self-improvement
v1 verifier	loop − no verifier	$+14.55$ pp	confounded as run
matched repair	verified − unverified	-0.01 pp	post-hoc; no detected marginal value here
fresh holdout	mixture − base	$+0.71$ pp	confirmatory; below 2 pp threshold

random-dimension (0.6211 ± 0.0285) or single-dimension (0.6385 ± 0.0123) controls. The score advantage over the trained baseline is real; the interpretation as self-improvement is not identified. Relative and absolute comparisons answer different questions, and the untouched-base gate changes the conclusion.

4.2 Four open gates explain the original contrast

The protocol-frozen comparison leaves multiple alternatives open. First, the Knights checker lowercased model output but not the expected label, so valid case variants could be rejected. Second, accepted responses were stored as short extracted answers, and many reasoning traces were truncated or unclosed; target format therefore changed with condition. Third, the adaptive controller remained in its initial band, making the observed controller contrast uninformative about adaptation. Fourth, the full-loop arm trained only on the final round's accepted set, whereas static and random-synthetic controls pooled data across rounds, so the nominal equal-budget design was not exposure matched. These defects are coupled: the apparent 14.55-point full-versus-no-verifier difference is an as-run association, not a causal verifier effect, and no decomposition of that gap is identified.

The strongest positive control supplied a fifth lesson. Its 0.6834 ± 0.0016 score came from a benchmark-derived generic corpus that reused evaluation prompts. Source-level gains concentrated on those families, a signature consistent with prompt-specific memorization. The observed score is valid for that trained model, but not for an instance-disjoint generalization estimand.

4.3 Corrected controls remove the identified advantages

Under full closed traces and matched exposure, v3b full, static, and random-synthetic arms score 0.6680 ± 0.0087, 0.6681 ± 0.0017, and 0.6662 ± 0.0025. Their observed means return to the neighborhood of the 0.6671 untouched base; because no equivalence margin was pre-specified, we do not claim equivalence. Full-dose retraining and a five-round extension likewise produce no stable advantage (Appendix Table 11); their role is diagnostic rather than confirmatory.

A matched no-verifier arm accepts complete traces regardless of correctness while holding target format, exposure, curriculum, and candidate generation fixed. It reaches 0.6681 versus 0.6680 for verification, a -0.01-point contrast with per-seed signs reversing. Only 3 of $5{,}843$ candidate traces (0.05%) fail the checker. Verification therefore has no detectable marginal value in this tested high-solve regime, where almost no error mass is available to filter. Exploratory low-solve settings show the complementary boundary: verifier filtering is positive when failures are induced by a shorter reasoning budget and on naturally lower-solve math tasks (Appendix A.17). This regime dependence is more informative than a global claim that verification works or does not work.

Rebuilding the generic corpus to be instance-disjoint yields 0.6678 ± 0.0056, down from 0.6834 ± 0.0016 with prompt overlap and only $+0.07$ points above the untouched base. The paired same-seed decrease is 1.56 points; the largest source drops occur on the previously overlapping benchmark families. Because reconstruction changes both instances and corpus size (1,139 to 1,031 traces), this is not a pure contamination-effect estimate. It is nevertheless decisive for interpretation: the positive control's advantage disappears after an instance-disjoint, near-matched rebuild.

Table 4: Claim-relevant gate status in our protocol-frozen case and three public pipelines audited in Appendix A.18. Symbols describe reported evidence: ✓ adequate, ∼ partial, × audited but violated, ∘ unreported (unknown), and — not applicable. Gates are interpreted per claimed estimand, not as a universal scorecard.

Gate	Estimand	Case v1	LMSI	Self-Rew.	SPIN
Untouched start	τ_{train}	✓	✓	∘	✓
Exposure match	τ_{data}	×	×	×	∼
Verifier validity	τ_{ver}	×	∼	∼	—
Controller active	τ_{ctrl}	×	—	—	∼
Instance-disjoint	τ_{gen}	×	∘	∘	∘

4.4 The audit generates a bounded, testable hypothesis

The instance-disjoint loop-plus-generic mixture reaches 0.6850 ± 0.0029 across three exploratory comparison seeds, $+1.8$ points over the untouched base and above both component-only arms. Two additional post-hoc hardening seeds preserve the direction. A dose sweep rises from both pure endpoints to a plateau at realized generic shares 0.20–0.347, and parameter-space assays are consistent with family-level interference dilution rather than simple retention (Appendix A.14). These observations define a falsifiable hypothesis: mixing disjoint generic traces with specialized loop data may rebalance cross-family interference.

The interpretation remains deliberately bounded. Interleaving generic data every round reaches 0.6862 ± 0.0032, close in observed mean to a fraction-matched single-shot mixture, but no equivalence margin was specified. The hypothesis was formulated after viewing E_{test} and all such analyses are exploratory. On a new 1,100-item holdout with five pre-registered training seeds, the mixture exceeds the base by 0.71 points at the seed level, with all seed differences positive, but falls below the frozen two-point minimal meaningful effect and has item-bootstrap intervals including zero. It therefore does not meet the pre-registered confirmation rule. This result does not establish no effect; it establishes that the exploratory $+1.8$-point pattern did not replicate at the magnitude claimed relevant.

5 Discussion

Recursive-training gains are properties of a comparison design, not of a score difference alone. The same protocol-frozen result supports "better than the strongest trained baseline" and rejects "better than the starting model." The five-gate contract makes that distinction explicit by tying every claim to the comparison that estimates it and to the alternatives that comparison must block.

From ablations to identification. Ablation tables enumerate variants; an identification contract states what a contrast is allowed to mean. For example, removing a verifier does not estimate τ_{ver} if it also changes accepted-token count, target completeness, or optimization steps. Comparing with an untouched checkpoint identifies absolute training change but not the source of that change. Reporting both comparisons is therefore not redundancy: the trained baseline asks whether one training procedure is less damaging than another, whereas the untouched base asks whether training improved the original model. This claim-to-estimand mapping is the transferable object of the paper.

5.1 An operational reporting standard

The contract can be used before, during, and after an experiment. **Before training**, authors declare the claim and its estimand, choose the minimum contrast, freeze the evaluation set and decision threshold, and specify which factors must be invariant. This makes the scientific target independent of the most favorable result eventually observed. **During training**, the pipeline records the quantities that determine whether the treatment occurred: accepted and rejected examples, token and update counts, target completeness, verifier regression tests, controller state transitions, and overlap-screen outputs. These are not implementation trivia. They establish whether labels such as "verified," "adaptive," and "matched" describe the realized experiment. **After evaluation**, every result receives one of three interpretations: identified support for the named estimand; identified evidence that the threshold was not met; or a non-identifying performance difference with the open gate stated explicitly. Repairs

Table 5: Admissible conclusions under the identification contract. The evidence label is part of the scientific result, not a footnote added after analysis.

Status	Required evidence	Admissible conclusion
Identified support	pre-specified contrast; relevant gates closed; threshold met	the named estimand is supported in the tested setting
Identified non-support	same design; threshold not met	the pre-specified claim is unsupported under its decision rule
Non-identifying contrast	one or more claim-relevant gates open	report the score difference and the unresolved alternative, not a mechanism effect
Post-hoc repair	corrected design after endpoint inspection	diagnostic or hypothesis-generating evidence; confirm on untouched data

selected after seeing the endpoint can close an engineering defect and motivate a new experiment, but they do not inherit the original test's confirmatory status.

This taxonomy avoids two common errors. First, it separates a negative result from a failed design: the untouched-base comparison in v1 is valid evidence against the pre-specified self-improvement claim, while the verifier contrast is non-identifying because multiple factors changed. Second, it separates engineering repair from evidential renewal: fixing a checker is necessary for future validity, but rerunning on an inspected benchmark cannot by itself create a new confirmatory result. The distinction preserves useful information from imperfect studies without granting every observed difference a causal interpretation.

Why trained baselines are a special risk. A trained comparator is simultaneously a method competitor and an outcome of its own data, formatting, and optimization choices. If it degrades below M_0, a recursive arm can win the relative comparison while also degrading. If it receives fewer tokens or shorter targets, the contrast mixes data provenance with exposure. Consequently, "best trained baseline" is not a substitute for the untouched checkpoint or an exposure-matched control: all three answer different questions. The compact reversal in Table 3 is therefore not a semantic caveat but an empirical demonstration that baseline choice changes the sign of the scientific conclusion.

Implications for experimental design. Future recursive-training studies can declare the contract before running the headline comparison. At minimum, the declaration should name the starting checkpoint, data and token budgets, target representation, verifier test suite, controller state trajectory, overlap unit, primary contrast, and decision threshold. If a gate is irrelevant, the associated claim should be absent; if it is relevant but open, the result remains a performance comparison rather than a mechanism estimate. This approach also clarifies efficient experiment design: one need not run every conceivable ablation, only the controls required by the claims the paper chooses to make.

What the case study establishes. The v1 audit rules out its intended interpretation without pretending to estimate the contribution of each defect. The absolute-base contrast shows that the loop did not improve its starting model. Controller logs show that this run cannot support a claim about adaptive control. Matched verified and unverified arms remove the nominal verifier gap in a regime where nearly every candidate already passes; exploratory low-solve settings reveal that this null is regime-specific. Finally, rebuilding the positive control instance-disjoint removes its apparent edge. Together, these results explain why the original comparison was non-identifying and show how targeted controls change the scientific conclusion.

Evidence beyond one pipeline. The public-system audit is deliberately claim-conditional. LMSI's reported ablations reveal sensitivity to target format and question source but do not isolate additive contributions. Self-rewarding models validate preference prediction, yet their headline evaluation remains entangled with response-length changes on a verbosity-sensitive benchmark (Dubois et al., 2024). SPIN reports its untouched starting checkpoint, while iteration and exposure are not fully separated. We do not infer that an unreported control failed. Rather, for each major claim examined, the public record leaves at least one required identifying condition absent, partial, or unclear. The contract thus supplies a common reporting language across heterogeneous recursive systems.

Constructive boundary. The audit also separates two design hypotheses from the headline claim. Verifier value appears conditional on error mass: it is near zero when 99.95% of candidates pass, and positive in the tested low-solve regimes. Disjoint mixtures show a recurring exploratory direction across model families and scales, but the fresh holdout does not clear the pre-specified two-point threshold. These findings motivate targeted future tests without being promoted to confirmed mechanisms.

A reviewer decision procedure. The contract reduces a broad recursive-training claim to five auditable questions. (1) What estimand does the headline sentence assert: absolute training gain, recursive-data value, verifier value, controller value, or clean generalization? (2) Which reported contrast estimates that quantity, rather than a neighboring one? (3) Which claim-relevant invariances and integrity checks are demonstrated, and which remain open or unreported? (4) Given those gates, is the admissible wording identified support, identified non-support, a non-identifying association, or a post-hoc diagnostic? (5) What single added control or untouched replication would change that status? This sequence turns review from an impressionistic reading of score tables into a reproducible decision about evidence. It also makes disagreement local: readers can contest the estimand, the required invariance, or the threshold without silently substituting a different claim. For authors, the same procedure exposes the decisive experiment before resources are spent on broader ablation grids.

Auditability is not causal decomposition. Successive repairs change several factors together, so they cannot assign shares of the 14.55-point v1 contrast to checker validity, target format, exposure, or optimization. The contract is a minimum standard against misattribution, not a guarantee of causal sufficiency. Its value is to expose exactly which statement is supported, which alternative remains open, and which next experiment would be decisive.

6 Conclusion

We introduced a claim-conditional identification contract for recursive synthetic-data training. Its five gates connect absolute improvement, recursive-data value, verifier value, controller value, and clean generalization to the minimum contrasts and integrity checks required to interpret them.

Applied to a protocol-frozen case study, the contract turns an apparent 10.26-point gain over a trained baseline into a different conclusion: the same loop is 4.12 points below its untouched starting model, the nominal verifier contrast is confounded, and corrected matched controls remove the identified advantages in observed means. The audit also yields testable regime hypotheses while a fresh holdout prevents an exploratory mixture pattern from being overstated.

Future recursive-training claims should therefore be evaluated against an explicit estimand and the controls required to close each relevant identification gate, rather than inferred from superiority over a trained baseline alone.

Limitations

Scope. The protocol-frozen study centers on one 8B model, three training seeds, three rounds, and checker-attached logic reasoning. Bounded exploratory tests cover additional model families, scales, and a math domain, but do not establish universal behavior. The five gates are necessary relative to stated estimands; they are not claimed to be jointly sufficient for every recursive system. **Statistical status.** Close means in corrected arms are descriptive because no equivalence margin was pre-specified. The audit-derived mixture is post-hoc, and its fresh 1,100-item holdout result did not meet the frozen two-point minimal meaningful effect; this is an unsupported confirmation outcome, not evidence of exact equivalence or zero effect. **Mechanism and provenance.** Coupled repairs can falsify the original attribution but cannot decompose per-defect effects. Interference dilution and low-solve verifier value remain bounded hypotheses, while legacy controller traces and some exploratory external-validity runs have less complete provenance than the primary artifact chain. These limits constrain mechanism and generalization claims without changing the identification reversal established by the protocol-frozen comparisons.

ETHICS STATEMENT

This work uses synthetic logic tasks and existing benchmark items, with no human participants or sensitive personal data. The main research-integrity risk is overstating a result from a defective pipeline; we address it by separating failed from corrected analyses and pre-specified from post-hoc evidence. Benchmark items come from public academic benchmarks under research-use licenses; any release respects the benchmark and base-model licenses and excludes credentials, private paths, and non-anonymous metadata.

REPRODUCIBILITY AND AVAILABILITY

Standard arms use LoRA rank 16, learning rate 5×10^{-5}, two epochs, and the shared chat template (exceptions in Appendix A.13). Reported results are linked to versioned artifacts; the supplement maps claims to results and scripts. The planned release includes the generator, controller, verifiers, training/evaluation scripts and run configurations, the sealed-test fingerprint, checker regression tests, the decontamination report, and figure-generation scripts. Large checkpoints and restricted credentials are excluded; a public release follows the anonymity period.

AI USE DISCLOSURE

Generative AI tools were used for research ideation; development and refinement of the identification framework and hypotheses; experimental-design discussion; synthetic-data generation; implementation, code drafting, and debugging; literature discovery; data formatting and analysis support; interpretation checks; and drafting, translating, and polishing portions of the manuscript. The in-loop solver was a locally hosted open-weight Qwen3-8B model whose generated traces were filtered by programmatic checkers. AI systems were not treated as authors and did not independently approve scientific claims. The human author reviewed the final protocol and interpretations, inspected source papers rather than relying on generated summaries, tested code and checker behavior, reconciled reported numbers against versioned artifacts, and reviewed all generated prose and tables. The human author takes final responsibility for the text, claims, code, data products, and submission.

REFERENCES

Christopher D Chambers and Loukia Tzavella. The past, present and future of registered reports. *Nature human behaviour*, 6(1):29–42, 2022.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*, 2024.

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

Elvis Dohmatob, Yunzhen Feng, Pu Yang, Francois Charton, and Julia Kempe. A tale of tails: Model collapse as a change of scaling laws. *arXiv preprint arXiv:2402.07043*, 2024.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*, 2024.

Shahriar Golchin and Mihai Surdeanu. Time travel in LLMs: Tracing data contamination in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=2Rwq6c3tvr.

Simeng Han, Hailey Schoelkopf, Yilun Zhao, Zhenting Qi, Martin Riddell, Wenfei Zhou, James Coady, David Peng, Yujie Qiao, Luke Benson, et al. Folio: Natural language reasoning with first-order logic. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 22017–22031, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Chengsong Huang, Wenhao Yu, Xiaoyang Wang, Hongming Zhang, Zongxia Li, Ruosen Li, Jiaxin Huang, Haitao Mi, and Dong Yu. R-Zero: Self-evolving reasoning LLM from zero data. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=96apU6YzSO.

Jianheng Huang, Leyang Cui, Ante Wang, Chengyi Yang, Xinting Liao, Linfeng Song, Junfeng Yao, and Jinsong Su. Mitigating catastrophic forgetting in large language models with self-synthesized rehearsal. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1416–1428, 2024a.

Jiaxin Huang, Shixiang Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. Large language models can self-improve. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 1051–1068, 2023.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=IkmD3fKBPQ.

Siyuan Huang, Pengyu Cheng, Haotian Liu, Tao Chen, Yihao Liu, Jingwei Ni, Shijie Zhou, Ziyi Yang, Gangwei Jiang, Mengyu Zhou, et al. Skill self-play: Pushing the frontier of llm capability with co-evolving skills. *arXiv preprint arXiv:2607.22529*, 2026b.

Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav Goldberg. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5075–5084, 2023.

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

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114 (13):3521–3526, 2017.

Ran Li, Zeyuan Liu, Yinghao Chen, Bingxiang He, Jiarui Yuan, Zixuan Fu, Weize Chen, Jinyi Hu, Chen Qian, Zhiyuan Liu, et al. Cpmobius: Iterative coach-player reasoning for data-free reinforcement learning. *arXiv preprint arXiv:2602.02979*, 2026.

Hanmeng Liu, Jian Liu, Leyang Cui, Zhiyang Teng, Nan Duan, Ming Zhou, and Yue Zhang. Logiqa 2.0—an improved dataset for logical reasoning in natural language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2947–2962, 2023.

Xiaonan Luo, Yue Huang, Kehan Guo, Ping He, Chuan Zou, Ting Hua, and Xiangliang Zhang. Learning from synthetic data without model collapse in iterative instruction tuning. *arXiv preprint arXiv:2607.17043*, 2026.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Brian A Nosek, Charles R Ebersole, Alexander C DeHaven, and David T Mellor. The preregistration revolution. *Proceedings of the National Academy of Sciences*, 115(11):2600–2606, 2018.

Yonatan Oren, Nicole Meister, Niladri Chatterji, Faisal Ladhak, and Tatsunori Hashimoto. Proving test set contamination in black-box language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=KS8mIvetg2.

Arjun Panickssery, Samuel R Bowman, and Shi Feng. Llm evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems*, 37:68772–68802, 2024.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. Nlp evaluation in trouble: On the need to measure llm data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10776–10787, 2023.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=RIu5lyNXjT.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, 2024.

Avi Singh, John D Co-Reyes, Rishabh Agarwal, Ankesh Anand, Piyush Patil, Xavier Garcia, Peter J Liu, James Harrison, Jaehoon Lee, Kelvin Xu, et al. Beyond human data: Scaling self-training for problem-solving with language models. *arXiv preprint arXiv:2312.06585*, 2023.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, 2023.

The BigBang Team. Bigbang: Pursuing open-ended intelligence through self-evolving synthesis of verifiable frontier tasks, 2026. URL https://endlessfrontier.tech/assets/paper.pdf. Technical report.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, 2024.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 13484–13508, 2023.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. WizardLM: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=CfXh93NDgH.

Cheng Xu, Nan Yan, Liming Chen, M Kechadi, et al. Phantom gains: Auditing self-improvement against a measured null. *arXiv preprint arXiv:2608.20290*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*, 2024.

Andrew Zhao, Yiran Wu, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. *Advances in Neural Information Processing Systems*, 38:105816–105879, 2026.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=shr9PXz7T0.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Table 6: Confirmatory v1 accuracy on the held-out 2,070-item evaluation set. Values for trained systems are mean ± sample SD over three training seeds. The unadapted base model has the highest overall accuracy; among trained systems, the single-dimension control is highest.

Condition	Overall	L1	L2
Unadapted base model	**0.6671**	0.6529	0.6745
Full DATAEVOLVE	0.6259 ± 0.0361	0.6567	0.6102
Static data	0.4839 ± 0.0818	0.5552	0.4474
Random synthetic	0.5234 ± 0.0781	0.5600	0.5046
Random dimensions	0.6211 ± 0.0285	0.6557	0.6034
Single dimension	**0.6385 ± 0.0123**	**0.6609**	0.6270
No verifier	0.4804 ± 0.0248	0.6076	0.4153

A DETAILED EXPERIMENTAL RESULTS

A.1 SEED-LEVEL CONFIRMATORY V1 RESULTS

Table 7: Overall accuracy by training seed for confirmatory v1.

Condition	20260912	41001	62002	Mean
Full DATAEVOLVE	0.6290	0.6604	0.5884	0.6259
Static data	0.4048	0.4787	0.5681	0.4839
Random synthetic	0.4449	0.6010	0.5242	0.5234
Random dimensions	0.6343	0.6406	0.5884	0.6211
Single dimension	0.6386	0.6507	0.6261	0.6385
No verifier	0.5063	0.4778	0.4570	0.4804

The full-minus-random-synthetic differences are 0.1841, 0.0594, and 0.0643, with a cross-seed mean of 0.1026. The reported item-bootstrap intervals were computed separately within each seed over the 2,070 evaluation items; they therefore quantify item-level variation within a training run rather than variation across training runs.

A.2 SEED-LEVEL V3B RESULTS

All v3 analyses and extensions reported below are exploratory because they were designed after observing the v1 evaluation results.

Table 8: Exploratory v3b results after correcting target handling and matching exposure. Values are mean ± sample SD over three seeds; the untouched base is a single greedy evaluation (no seed-level SD). No comparison is statistically supported, so no row is emphasized.

Condition	Overall	L1	L2
Full loop (corrected targets)	0.6680 ± 0.0087	0.6571	0.6735
Static, exposure matched	0.6681 ± 0.0017	0.6576	0.6735
Random synthetic, exposure matched	0.6662 ± 0.0025	0.6524	0.6732
Untouched base (reference)	0.6671	–	–

Per-seed full-minus-static differences are 0.0116, -0.0077, and -0.0044 (mean -0.0002). Full-minus-random-synthetic differences are 0.0097, -0.0057, and 0.0014 (mean 0.0018).

A.3 VERIFIER DEFECT

The v1 Knights checker compared parsed role labels without first normalizing the gold labels to lowercase. For affected records this produced an empty role map, allowing the universal consistency condition to pass vacuously. Inspection found that malformed responses were widely accepted in this high-volume family. The corrected checker normalizes labels and rejects malformed outputs; analyses performed after evaluating E_{test} are treated as exploratory.

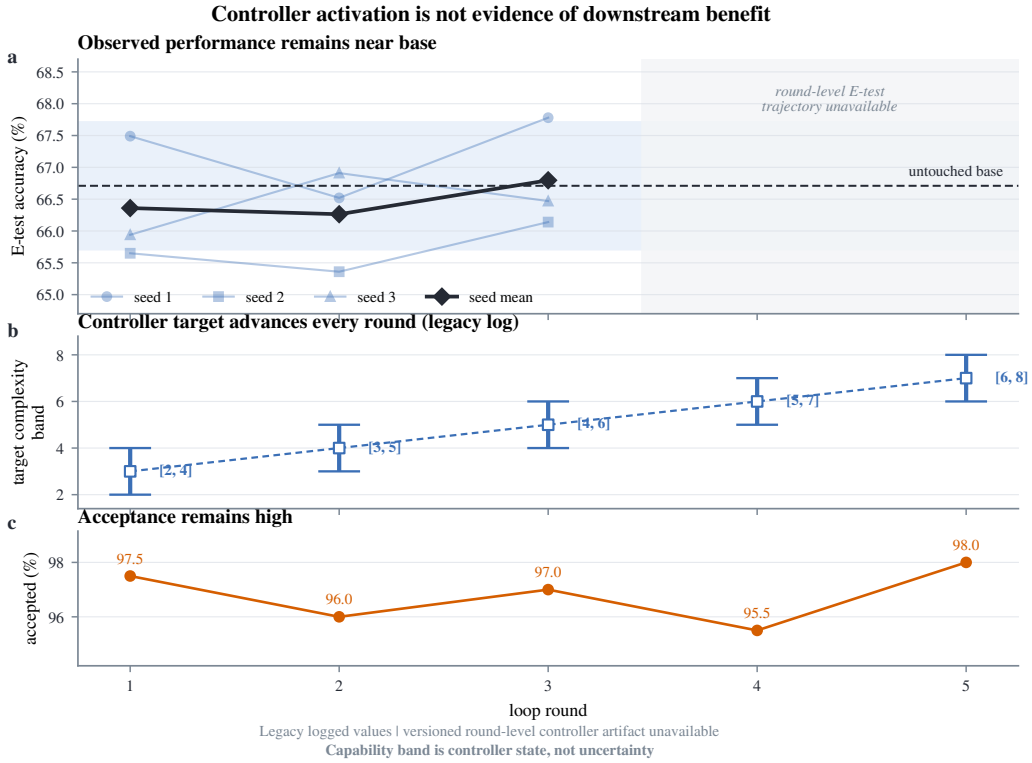

Figure 4: Controller activation and observed performance after correcting the v1 pipeline. (a) E-TEST accuracy for three independent training seeds and their mean over the three rounds with retained round-level evaluation data; the dashed line is the unadapted base model and the blue region denotes a descriptive ± 1 percentage point, not an equivalence margin. (b–c) Five-round target bands and acceptance rates from the original controller logs. The capability band represents controller state rather than uncertainty. Round-level controller traces and E-TEST values are unavailable for rounds 4–5, so these panels document only that the controller changed state in its logs; they establish neither a performance benefit nor a causal link between controller movement and performance, and the sequence is not presented as a sustained-improvement curve.

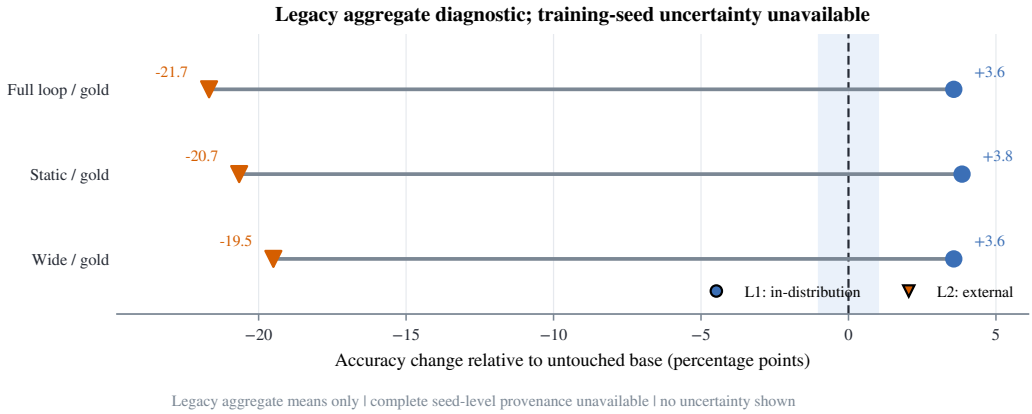

Figure 5: Legacy aggregate diagnostic; training-seed uncertainty unavailable. The v2 layer dissociation relative to the unadapted base model: short gold-answer targets produce a small L1 in-distribution lift and severe L2 external forgetting in every reported condition. The available results are legacy aggregate means with incomplete seed-level provenance, so no SD or confidence interval is shown. The base result is from a single greedy evaluation.

Table 9: Overall accuracy by seed for exploratory v3b.

Condition	Seed 1	Seed 2	Seed 3	Mean
Full loop (corrected targets)	0.6778	0.6614	0.6647	0.6680
Static, exposure matched	0.6662	0.6691	0.6691	0.6681
Random synthetic, exposure matched	0.6681	0.6671	0.6633	0.6662

Table 10: Implementation issues identified in v1 and their implications for interpretation.

Issue	Observation	Implication
Verifier defect	A missing case normalization made the Knights role-map check vacuous and admitted malformed responses.	The observed verifier contrast does not isolate the effect of correct programmatic verification.
Target integrity	Manual and programmatic inspection found widespread truncated or unclosed reasoning under the 256-token generation cap.	Training-target quality was substantially below the intended specification.
Inactive controller	The controller never left its initial capability band.	The experiment did not meaningfully exercise adaptive scheduling; policy comparisons cannot validate the intended mechanism.
Exposure mismatch	The full arm trained on one round at a time, whereas static and random-synthetic baselines trained on pooled data from all rounds.	Differences mix curriculum, exposure, and forgetting; "equal budget" was not achieved.

A.4 Target integrity and controller activation

The v1 solver used a 256-token generation cap and stored the full response as the training target. Inspection found widespread truncation or unclosed reasoning among accepted targets. The controller also remained in its initial capability band throughout the three rounds; consequently, v1 does not test whether adaptive movement through the intended data dimensions improves performance.

A.5 v3c dose-axis and T5 extension results

Table 11: Exploratory v3c dose study at fixed full-target format; means ± sample SD over three training seeds (seed-level values in Table 13). Matched-dose rows repeat the v3b arms from Table 8 (the loop's final round, ≈1,940 pairs; the baseline subsamples, 1,400 pairs); full-dose rows retrain from the cumulative loop pool or the untruncated baseline pools (≈5,800 pairs). The untouched base is a single greedy evaluation (Overall 0.6671, L1 0.6529, L2 0.6745) and has no seed spread.

Condition (training pairs)	Overall	L1	L2
Loop, last round only (≈1,940)	0.6680 ± 0.0087	0.6571 ± 0.0015	0.6735 ± 0.0137
Static, matched (1,400)	0.6681 ± 0.0017	0.6576 ± 0.0017	0.6735 ± 0.0017
Random synthetic, matched (1,400)	0.6662 ± 0.0025	0.6524 ± 0.0017	0.6732 ± 0.0034
Loop, cumulative (≈5,800)	0.6602 ± 0.0077	0.6586 ± 0.0038	0.6611 ± 0.0123
Static, full pool (≈5,800)	0.6638 ± 0.0039	0.6567 ± 0.0017	0.6674 ± 0.0053
Random synthetic, full pool (≈5,800)	0.6607 ± 0.0037	0.6572 ± 0.0025	0.6625 ± 0.0055

A.6 Per-source L2 accuracies

A.7 Wide-module analysis

With widened LoRA target modules (seven projections instead of two), two seeds yield 0.6140 and 0.6184 overall accuracy (external layer 0.594 and 0.599), well below both the unadapted base

Table 12: Exposure decomposition of the v3c dose axis (ranges over three seeds). Training pairs are the prompt–target rows given to the trainer; distinct prompts count unique prompt strings—the task generator's small combinatorial space makes the same task text recur within and across pools; tokens are Qwen3 tokenizer lengths of the training collator's chat-rendered sequences (truncation 2048), summed per pool; optimizer updates are two epochs at effective batch 16 (per-device batch 2, gradient accumulation 8; trainer rounding differs by at most two updates).

Condition	Training pairs	Distinct prompts	Tokens (M)	Updates
Loop, last round only	1,941–1,947	887–915	1.65–1.69	244
Static, matched	1,400	634–686	1.14–1.21	176
Random synthetic, matched	1,400	711–733	1.19–1.22	176
Loop, cumulative	5,785–5,811	1,946–2,018	4.96–5.04	722–726
Static, full pool	5,813–5,837	1,702–1,740	4.87–4.92	726–728
Random synthetic, full pool	5,787–5,799	2,038–2,065	4.97–5.04	722–724

Table 13: Seed-level values for the v3c dose study and the five-round extension (T5); seeds 20260912, 41001, and 62002 in order.

	Overall			L1			L2		
Condition	Seed 1	Seed 2	Seed 3	Seed 1	Seed 2	Seed 3	Seed 1	Seed 2	Seed 3
Loop, last round only	0.6778	0.6614	0.6647	0.6557	0.6571	0.6586	0.6891	0.6635	0.6679
Static, matched	0.6662	0.6691	0.6691	0.6557	0.6586	0.6586	0.6715	0.6745	0.6745
Random synthetic, matched	0.6681	0.6671	0.6633	0.6543	0.6514	0.6514	0.6752	0.6752	0.6693
Loop, cumulative	0.6599	0.6527	0.6681	0.6629	0.6571	0.6557	0.6584	0.6504	0.6745
Static, full pool	0.6652	0.6667	0.6594	0.6557	0.6586	0.6557	0.6701	0.6708	0.6613
Random synthetic, full pool	0.6599	0.6647	0.6575	0.6543	0.6586	0.6586	0.6628	0.6679	0.6569
Five-round extension (T5)	0.6671	0.6657	0.6739	0.6557	0.6629	0.6557	0.6730	0.6672	0.6832

model (0.6671) and the standard-module v3 loop (0.6680). Under this narrow-domain training signal, increasing the number of trainable parameters coincides with poorer transfer, consistent with target format and data breadth being more consequential than adapter capacity in this regime.

A.8 Seed-level paired statistics

All comparisons in Table 15 are exploratory; none is designated confirmatory. The prompt-overlapping result is not used as evidence for generalization. The v1 full-versus-baseline gaps have large point estimates ($+10.3$ and $+14.2$ points) but three-seed intervals that include zero; neither these comparisons nor the v3 base comparison can be estimated precisely from three seeds. All v1 full-loop seeds fall below the unadapted base model (differences -0.038, -0.007, -0.079). These tests remain underpowered at $n{=}3$.

A.9 No-verifier seed-level results

Across the three seeds, differences from the verified arm are $+1.3$, -0.6, and -0.7 points, yielding a mean difference of -0.01 points. This sample does not detect a verification effect under the corrected target format and was not designed as an equivalence test.

A.10 Replay and generic-only seed-level results

A.11 Decontamination of the replay corpus

The corrected generic corpus was assembled from held-out or disjoint material from the same external sources: the LogiQA2 dev split (900 candidates; the evaluation set uses the train split; rendered with the identical prompt template), the FOLIO validation split (204 items; the evaluation uses train), a KOR-logic pool verified to be item-disjoint from the evaluation's KOR family under normalized exact matching (250), and BBH logical-deduction variants absent from the evaluation set (500 candidates; 109 survive filtering). Every candidate was screened against all 2,070 evaluation

Table 14: External-layer (L2) accuracy by source for the unadapted base model, the confirmatory v1 full loop (seed 20260912, the seed closest to the v1 mean), and the exploratory v3 full loop (seed 20260912). The v1 degradation is concentrated in FOLIO and the tracking tasks; v3 lands close to or slightly above the base result on most sources.

Source	Base	v1 full loop	v3 full loop
BBH boolean expressions	1.000	0.945	1.000
BBH logical deduction (3 obj.)	0.886	0.891	0.968
BBH tracking shuffled (3 obj.)	0.977	0.650	0.936
FOLIO	0.391	0.087	0.391
KOR logic	0.272	0.260	0.296
LogiQA 2.0	0.591	0.557	0.613
L1 generated (reference)	0.653	0.654	0.656

Table 15: Seed-level paired differences (n=3 seeds), computed from retained per-seed summaries. Comparisons between trained conditions use paired differences; comparisons with the single untouched-base estimate use one-sample differences against a fixed constant. Intervals are 95% intervals from the t distribution with 2 df, reflecting variability across the three training seeds only. All comparisons are exploratory unless explicitly designated confirmatory; with three training seeds, intervals are highly uncertain. †Computed on the original prompt-overlapping corpus; the instance-disjoint comparison used for interpretation is reported below.

Contrast	Mean paired difference	SD of differences	Interval	Evidence status
v1 full vs. random syn.	$+0.1026$	0.0707	$[-0.073, +0.278]$	Exploratory
v1 full vs. static	$+0.1421$	0.1076	$[-0.125, +0.409]$	Exploratory
v1 full vs. untouched base	-0.0412	0.0361	$[-0.131, +0.049]$	Exploratory
v3b full vs. static	-0.0002	0.0103	$[-0.026, +0.025]$	Exploratory
v3b full vs. random syn.	$+0.0018$	0.0077	$[-0.017, +0.021]$	Exploratory
Generic-only vs. mix (original corpus)†	$+0.0083$	0.0010	$[+0.006, +0.011]$	Exploratory
Mix (disjoint) vs. base	$+0.0179$	0.0029	$[+0.011, +0.025]$	Exploratory
Mix (disjoint) vs. generic-only (disjoint)	$+0.0172$	0.0052	$[+0.004, +0.030]$	Exploratory
Interleave 0.35 vs. base	$+0.0191$	0.0032	$[+0.011, +0.027]$	Exploratory
Interleave 0.35 vs. single-shot 0.35	-0.0002	0.0038	$[-0.010, +0.009]$	Exploratory
Interleave 0.35 vs. loop-only	$+0.0182$	0.0055	$[+0.005, +0.032]$	Exploratory
Anneal 300 vs. base	$+0.0013$	0.0020	$[-0.004, +0.006]$	Exploratory

items using (i) normalized exact matching (case-folding, whitespace collapsing, and answer-suffix stripping; one LogiQA2 candidate removed) and (ii) scaffold-stripped embedding similarity with `bge-small-en-v1.5` at cosine 0.9. Removing shared instruction scaffolds before comparison prevents template similarity from being treated as instance duplication; this step removed 391 BBH and 3 LogiQA2 candidates. The final corpus contains 1,459 prompts (sha256 `4cd778dd...`); applying the same closed-trace format filter as for the original corpus yields 1,031 training traces (sha256 `48674be8...`), or 90.5% of the original corpus size. This difference lies within the dose range with a flat response in Section 4.3. The code release includes the construction script, per-subset report, source-split information, and dev-split sha256 `bbefb563...`.

A.12 JUDGE-ARM CALIBRATION

A reference-informed LLM judge was calibrated against construction-truth labels on 240 sampled responses, one third of which were deliberately corrupted negatives. Two prompt variants yielded Cohen's κ values of 0.161 and 0.164, below the pre-specified 0.6 agreement threshold, largely because the judge rejected many correct answers. We therefore omitted this arm and compare only programmatic verification with no verification.

Table 16: Exploratory no-verifier arm under the corrected target format: complete traces are accepted without correctness filtering (all three seeds).

Condition	Seed 1	Seed 2	Seed 3	Mean
No verifier (full traces, no check)	0.6647	0.6676	0.6720	0.6681
(reference: with verifier)	0.6778	0.6614	0.6647	0.6680

Table 17: Seed-level replay results for mixtures of loop data and generic self-distilled traces, together with generic-only training. The original generic corpus contains prompts that overlap verbatim with the evaluation set and therefore provides only an upper-bound comparison (Appendix A.11); results for the decontaminated corpus are reported separately. The nominal 50/50 slice truncates at the generic pool (1,139 traces; realized share 0.37).

Condition	Seed 1	Seed 2	Seed 3	Mean
Loop + generic (nominal 50/50)	0.6754	0.6744	0.6754	0.6751
Generic only (no loop data)	**0.6841**	**0.6816**	**0.6845**	**0.6834**
Decontaminated corpus (instance-disjoint, Appendix A.11):				
Generic only (no loop data)	0.6720	0.6614	0.6700	**0.6678**

A.13 Training configuration and mixing-fraction definition

All trained conditions use the LoRA configuration specified before the confirmatory evaluation: rank 16 ($\alpha = 32$, dropout 0.05) on the `q_proj`/`v_proj` projections—the wide-module analysis uses seven projections instead—at learning rate 5×10^{-5} for two epochs with micro-batch 8, rendered through the shared chat template, with per-round training starting from fresh adapters. The anneal arm resumes from the final v3 loop adapters at learning rate 1×10^{-5} for one epoch on 300 traces. The mixing fraction f in the dose response is operationalized per training round as $n_{\text{gen}} = \lfloor n_{\text{loop}}\, f/(1-f) \rfloor$, so f is exactly the generic share of the resulting per-round training set; the generic slice is drawn without replacement from the 1,031-trace instance-disjoint corpus of Appendix A.11 and re-sampled each round.

A.14 Per-source transfer matrix

Figure 7 (right) carries the main per-source visual: a heatmap of changes against the untouched base for five representative arms. The full raw matrix for all twelve arms, grouped by source family and marking the cells whose values survive only as three-seed aggregate means, is provided in the supplementary material. Values for trained arms are three-seed means; the base row is a single greedy evaluation.

The loop's accepted data is itself balanced across its three generator families (knights, arithmetic, and zebra at 34/33/33% across seeds and rounds), and the loop's family space contains no tracking-analog family: the tracking loss in every loop-only arm is consistent with passive interference rather than active overwrite by an over-weighted family. In this setup, the mixture results suggest that outside-loop data may help retain the affected capability. Two additional post-hoc hardening runs of the single-shot mixture at the pool-limited share (fixed loop source, new LoRA seeds; per-seed summaries are not packaged with the release, so this pooling is traceable to the aggregate) raise the five-seed estimate to 0.6840 ± 0.0044; all five seeds exceed the unadapted base model (minimum $+1.0$ points). In the per-round interleave variant, the dual-retention signature (LD-3 0.996, tracking 0.959) appears in every seed; sequential annealing raises tracking to 0.968 while the taught family falls to 0.873.

Gradient analyses. To evaluate the dilution account more directly, we analyzed the trained adapters and corpora without additional training. First, *update similarity* between the final LoRA deltas of the loop-only, generic-only, mixture, and dose arms: the two corpora push weights in nearly orthogonal directions ($\cos(\Delta_{\text{loop}}, \Delta_{\text{gen}}) = 0.18$ across three seeds), while two independent runs at the same realized generic share (0.347) agree at $\cos = 0.996$—indicating high reproducibility for this training recipe. The mixture update is not the weighted average of its parents: its cosine to the share-weighted

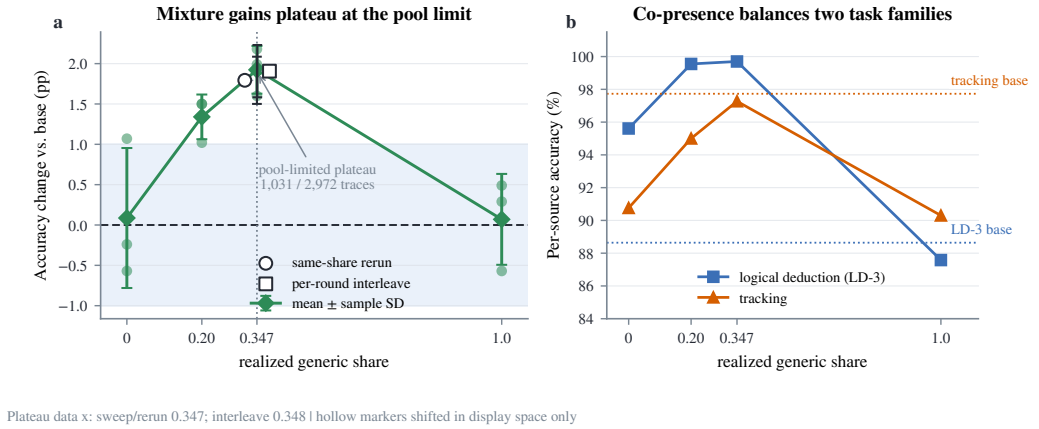

Figure 6: Exploratory mixing-share dose response, plotted at the realized generic share. Left: each filled point summarizes three independent training seeds with sample-SD whiskers. The nominal 0.35 and 0.5 mixtures both truncate to the full 1,031-trace pool (realized share approximately 0.347); the open circle is therefore a same-share rerun, not a higher dose, and the square is the per-round interleave variant at realized share 0.348. Right: BBH logical deduction and tracking three-seed aggregate means; source-level seed values are unavailable, so no uncertainty is shown.

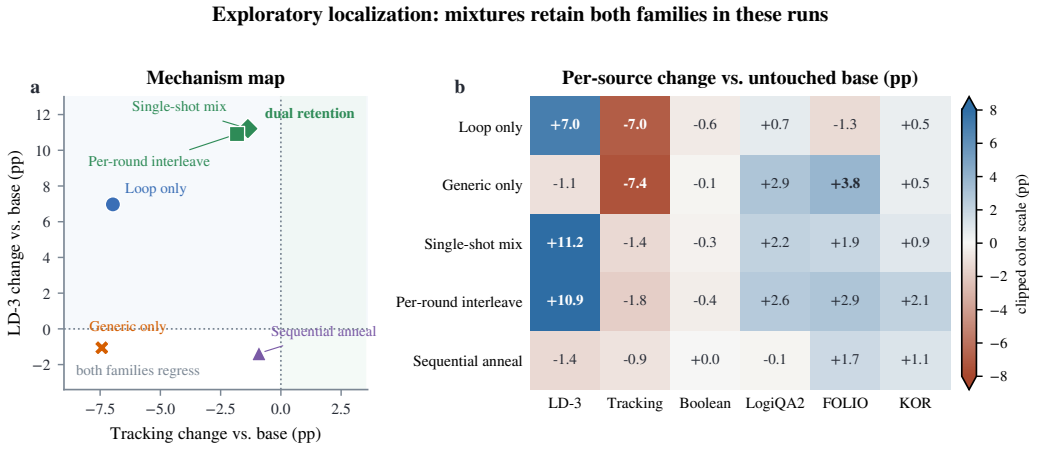

Figure 7: Exploratory mechanism localization using three-seed aggregate means. Left: family-level movement relative to the unadapted base model; loop-only arms gain LD-3 while losing tracking, mixtures retain both, and sequential annealing restores tracking while eroding LD-3 below base. Right: the complete 5×6 matrix of per-source changes in percentage points. Source-level seed values are unavailable, so no uncertainty is shown. The pattern is consistent with interference dilution but is not confirmatory evidence of that mechanism.

interpolation is 0.78 and its norm is $1.72\times$ larger, and the direction moves monotonically away from Δ_{loop} with realized share (cos drops from 0.85 at share 0.2 to 0.70 at 0.347). Second, *gradient cosine* between families in the LoRA subspace at four models, where g_L, g_G, and g_{liqa} denote average gradients over the loop corpus, the disjoint generic corpus, and its LogiQA2-dev subset (Table 18): at the unadapted base model, loop- and generic-family gradients are correlated ($\cos \approx 0.63$–0.68); after loop-only training the loop gradient collapses ($\|\nabla\|=0.026$) while the generic gradient is roughly seven times larger (0.179) and nearly orthogonal ($\cos \approx -0.01$), and the generic-only model shows the mirror asymmetry. In the mixture, both families' gradient norms are small and balanced (0.029 vs. 0.032) while cross-family cosines turn slightly negative (-0.06 to -0.10). Split-half within-family cosines mark the noise floor: they remain high (0.90–0.96) where gradients are large and drop where the norm collapses. Together, the single-corpus models sit at lopsided stationary points with

unresolved gradient pressure on the untrained families, and the mixture re-balances that pressure—a gradient-level signature consistent with interference dilution rather than replay-based retention. The gradient analysis uses one adapter seed, the LoRA subspace, up to 64 items per family, and 32 items per split half; it therefore provides exploratory rather than confirmatory evidence for the mechanism.

Table 18: Exploratory mechanism analyses on trained adapters and corpora. Gradient statistics use one adapter seed, the LoRA subspace, and 64 items per family. Left: per-family gradient statistics at each model, with split-half within-family cosine as a reliability reference. Right: update-direction similarities of final LoRA deltas (three seeds for the first row; the realized generic share is 0.347 for the mixture).

model	$\cos(g_L, g_G)$	$\cos(g_L, g_{\text{liqa}})$	$\|g_L\|$	$\|g_G\|$	within-G
base	0.68	0.63	0.045	0.048	0.90
loop-only	-0.01	0.00	0.026	0.179	0.96
generic-only	0.14	-0.03	0.045	0.023	0.23
mixture	-0.06	-0.10	0.029	0.032	0.37

comparison	cosine
$\cos(\Delta_{\text{loop}}, \Delta_{\text{gen}})$	0.18
$\cos(\Delta_{\text{mix}}, \Delta_{\text{loop}})$	0.70
$\cos(\Delta_{\text{mix}}, \Delta_{\text{gen}})$	0.50
$\cos(\Delta_{\text{mix}},\ \text{interp.})$	0.78
$\|\Delta_{\text{mix}}\|$ / $\|\text{interp.}\|$	1.72
$\cos(\Delta_{0.2}, \Delta_{\text{loop}})$ / $\cos(\Delta_{0.347}, \Delta_{\text{loop}})$	0.85 / 0.70
$\cos(\Delta_{\text{mix}}, \Delta_{0.347})$	1.00

A.15 Summary of experimental versions

Table 19 summarizes each experimental version, its design role, and its interpretation.

Table 19: Experimental versions and their interpretation.

Version	Design	Role in analysis
v1	Pre-specified primary experiment	Observed endpoint; implementation and design limitations constrain causal interpretation
v2	Exploratory, incomplete	Target-format diagnostic without a balanced main comparison
Original v3	Exploratory, incomplete	Excluded from accuracy analyses because the run did not complete
v3b	Exploratory, complete	Comparison after correcting the target format
v3c	Exploratory, complete	Dose-axis comparison
v3T5	Exploratory extension	Five-round extension of the corrected pipeline
Judge filter	Calibration only	Agreement was below the pre-specified threshold; no training arm was conducted
Replay and generic-only	Exploratory, complete	Data-breadth comparison

A.16 Complete evidence landscape

Figure 8 shows the complete evidence landscape with every condition, including the v2/v3 variants and prompt-overlapping probes omitted from Figure 3.

A.17 External-validity exploratory extensions

All findings above come from one model family, one scale, and one domain. A bounded exploratory program (24-hour budget, one H100 GPU; three training seeds per condition; same frozen hyperparameters and corpora as Section 4.2; post-hoc exploratory) probes these axes directly, holding the training corpora and the frozen evaluation fixed.

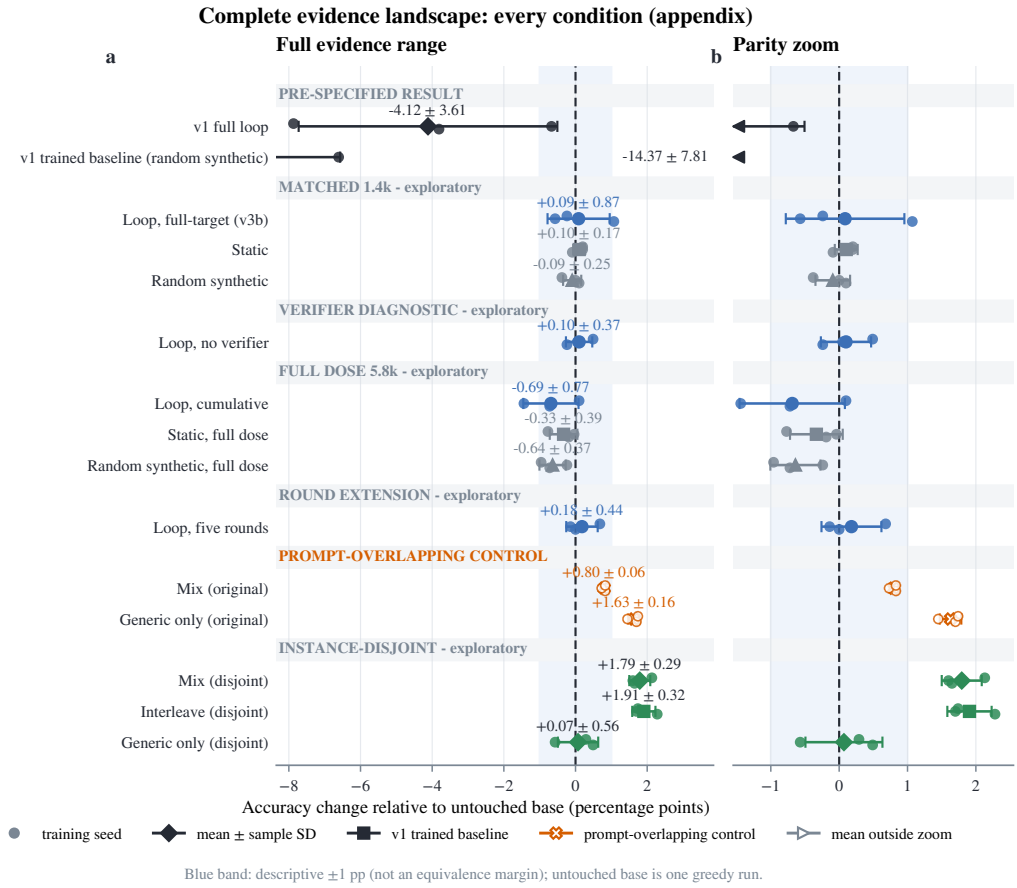

Figure 8: Complete evidence landscape with every condition, including the v2/v3 variants and prompt-overlapping probes omitted from Figure 3: full range (left) and parity zoom (right). Dots are independent training seeds, symbols with whiskers are means ± sample SD, and crossed markers denote prompt-overlapping probes. The blue ±1-percentage-point region is descriptive, not a pre-specified equivalence margin.

Second model family. Training Mistral-7B-Instruct-v0.3 (Jiang et al., 2023) on the same corpora does not reproduce the within-family pattern (Table 20): the loop corpus lifts overall accuracy by $+3.5$ points, concentrated entirely in the in-distribution layer ($+18.5$ points, all three seeds) with a -4.1-point external-layer loss; the generic-only control collapses L1 by -23.5 points; the mixture is again the best condition ($+7.8$ points). The L1-specific gain is consistent with answer-format alignment rather than capability transfer—the same loop corpus leaves Qwen3-8B at parity with its base.

Table 20: Cross-family exploratory replication (Mistral-7B-Instruct-v0.3; mean ± sample SD over three seeds; the untouched base is a single greedy evaluation). Training corpora, hyperparameters, and the evaluation endpoint are identical to the main experiments; only the base model changes.

Condition	Overall	L1	L2
Untouched base (Mistral-7B)	0.4449	0.4871	0.4234
Loop final-round corpus	0.4800 ± 0.0125	0.6719 ± 0.0044	0.3820 ± 0.0167
Mixture (realized share 0.347)	0.5232 ± 0.0119	0.6481 ± 0.0050	0.4594 ± 0.0168
Generic-only (decontaminated)	0.3610 ± 0.0120	0.2524 ± 0.0424	0.4166 ± 0.0088

Second scale. On Qwen3-4B (Yang et al., 2025) (untouched base 0.6541, only 1.3 points below the 8B base), loop training is mildly negative (-1.2 points), the mixture is again the best trained

condition ($+2.4$ points), and generic-only training collapses the external layer by -18.1 points with L1 unchanged (Table 21). The layer that the generic-only interference attacks therefore depends on the family, and its severity grows at smaller scale.

Table 21: Cross-scale exploratory replication (Qwen3-4B; mean $\pm$ sample SD over three seeds; the untouched base is a single greedy evaluation).

Condition	Overall	L1	L2
Untouched base (Qwen3-4B)	0.6541	0.6600	0.6511
Loop final-round corpus	0.6423 ± 0.0149	0.6547 ± 0.0041	0.6360 ± 0.0232
Mixture (realized share 0.347)	0.6776 ± 0.0042	0.6543 ± 0.0014	0.6895 ± 0.0056
Generic-only (decontaminated)	0.5343 ± 0.0037	0.6605 ± 0.0008	0.4698 ± 0.0059

Low-solve verifier regimes. In the main experiments the verifier has essentially nothing to filter (0.05% of the no-verifier pool fails the check). We swept the full generator complexity range: at bands $[6, 8]$, $[8, 9]$, and $[9, 9]$ the base solver verifies 96.5%, 95%, and 96.5% of generated tasks, so a low-solve regime is structurally unreachable within this generator family. We therefore induced it by budget and by domain. Under a $1{,}024$-token solver budget at band $[8, 9]$ (acceptance 69.5%), the verified arm matches the untouched base (0.6704 ± 0.0070 vs. 0.6671) while the no-verifier arm, trained on trajectories including roughly 30% failures, drops to 0.6377 ± 0.0049 (paired difference $+3.3 \pm 1.0$ points; three of three seeds positive; Table 22). In a second domain, GSM8K (Cobbe et al., 2021) is naturally low-solve: 66% of self-solved answers pass the numeric checker at full budget. There the same contrast yields $+1.7 \pm 0.5$ points on the frozen evaluation and $+3.9 \pm 1.7$ points on a disjoint 300-item math holdout, where the verified arm also beats the untouched base by $+5.9$ points in-domain (Table 23). Across the three regimes—high-solve logic (null), budget-induced low-solve logic, and natural low-solve math—the verifier's marginal value is positive in both tested low-solve regimes and absent in the tested high-solve regime, and its sign is protective against degradation on the frozen endpoint and constructive in-domain.

Table 22: Verification contrast across regimes (mean $\pm$ sample SD over three seeds where available). The high-solve row repeats the main corrected-pipeline contrast of Section 4.3.

Regime	Acceptance	Verified	No verifier	Paired diff.
High-solve logic, band $[2, 4]$	$\approx$100%	0.6680 ± 0.0087	0.6681	-0.01
Budget-limited logic, band $[8, 9]$	69.5%	0.6704 ± 0.0070	0.6377 ± 0.0049	$+3.3 \pm 1.0$
Natural low-solve math (GSM8K)	66%	0.6760 ± 0.0044	0.6594 ± 0.0029	$+1.7 \pm 0.5$

Table 23: Second-domain (GSM8K) arms on both endpoints; mean $\pm$ sample SD over three seeds. The math holdout contains 300 problems disjoint from the training pool; the base math accuracy is a single greedy evaluation.

Condition	Frozen eval (2,070 items)	Math holdout (300 items)
Untouched base	0.6671	0.6200
Verified arm	0.6760 ± 0.0044	0.6789 ± 0.0126
No-verifier arm	0.6594 ± 0.0029	0.6400 ± 0.0133

A.18 Claim-to-control analysis of public recursive-training systems

We compare the reported evidence for three public systems, one from each recursive-training family, against the criteria in Table 4: LMSI, self-generated instruction fine-tuning (Huang et al., 2023); self-rewarding language models, iterative DPO on self-judged preferences (Yuan et al., 2024); and SPIN, self-play fine-tuning (Chen et al., 2024). The analysis uses only controls explicitly reported in each paper. When a criterion is unmet or unreported, the corresponding gain cannot be uniquely attributed to the proposed mechanism.

LMSI (self-generated instruction fine-tuning). LMSI reports the unadapted PaLM-540B against its trained model on all six benchmarks (Table 3; criterion 1 ✓), but includes neither a human-labeled SFT baseline nor a matched-exposure control (criterion 2 ×) and does not use an adaptive controller (criterion 4 —). Its ablations show that removing CoT formats reduces CoT-prompted GSM8K accuracy from 73.5 to 61.6 (Table 5), while replacing benchmark training questions with self-generated questions reduces it to 66.2 (Table 6). The reported GSM8K result is sensitive to both target format and the source of training questions. Because the ablations are not orthogonal, they neither identify additive contributions nor isolate a unique residual attributable to self-generated correctness. The majority-vote filter is described as well calibrated (Figure 2) but is not evaluated against the available ground-truth labels (criterion 3 ∼), and no decontamination analysis is reported (criterion 5 ○).

Self-rewarding models (iterative self-judged DPO). The paper reports neither the unadapted base model nor the SFT baseline on AlpacaEval 2 (criterion 1 ○; Sections 3.2.1–3.2.2). Each iteration adds both data and DPO training without an equal-exposure control (criterion 2 ×; Section 3.1.3). The self-judge is evaluated on held-out human preferences (pairwise accuracy $65.1 \rightarrow 81.7$; Table 4) but not under adversarial perturbations; length bias remains uncorrected (Section 3.2.1), and reward hacking is left for future work (Section 6) (criterion 3 ∼). Because a fixed model generates prompts for the next iteration, prompt generation itself is not recursively updated (criterion 4 —). The paper also reports no overlap screen between self-generated training prompts and the AlpacaEval/MT-Bench prompts (criterion 5 ○). The headline AlpacaEval result is potentially confounded by response length: the paper reports increasing output lengths across iterations, while length-controlled AlpacaEval shows that the metric is generally sensitive to verbosity (Dubois et al., 2024). Model-based evaluators can also exhibit source-dependent self-preference (Panickssery et al., 2024), although that result does not establish the direction or magnitude of bias in this comparison. The most directly controlled reported comparison is the $+0.40$ MT-Bench gain over the SFT baseline.

SPIN (self-play fine-tuning). SPIN reports the untouched starting SFT checkpoint alongside all iterations and separately reports an additional-SFT control (criterion 1 ✓; Tables 4–6). Training exposure is not matched across checkpoints and controls: SPIN uses 50k examples at iteration 0 and 100k examples in later iterations for two epochs, whereas the reported additional-SFT control uses one epoch over the 200k corpus (criterion 2 ∼). Consequently, the iterative effect is not isolated under a common optimization budget. There is no data verifier (criterion 3 —), and the MT-Bench judge receives no position- or verbosity-bias calibration. The generated-data distribution shift across iterations is argued qualitatively rather than measured (criterion 4 ∼), and no contamination check covers the training corpus against GSM8K—the largest reported gain—or the other evaluation benchmarks (criterion 5 ○). Figures 4–5 suggest an effect of iterative synthetic data beyond plain SFT overtraining, although its magnitude remains confounded with exposure.

Related evidence. Related results show that attribution concerns arise in several self-training settings. ReSTEM reports that iterative self-training filtered by external binary correctness feedback can outperform fine-tuning on human-written solutions in mathematical reasoning and code generation (Singh et al., 2023); intrinsic self-correction without external feedback can fail or degrade (Huang et al., 2024b); and a contemporaneous preprint evaluates self-improvement against a pre-specified measured null, finding no discoveries under per-problem tests with false-discovery-rate control on held-out replicates (not peer-reviewed) (Xu et al., 2026). Together with Table 4, these cases motivate broader testing of the five criteria across iterative-training systems.

A.19 Pre-registered replication on an independent holdout

Before evaluating the independent holdout, we pre-specified the mixture share (nominal 0.35, realized 0.348), data construction, target format, all arms, five seeds, the primary contrast (mixture minus unadapted base model), a non-inferiority margin (1.0 point), an equivalence band (± 0.5 points), stopping and multiplicity rules, exclusion criteria, and the analysis code (SHA-256 pinned). The 1,100-item holdout was constructed before evaluation from sources not previously used for training or evaluation: newly generated instances beyond the exhausted original domain (arithmetic constraints, knights-and-knaves, and zebra puzzles) and unused splits (LogiQA 2.0 test (Liu et al., 2023), the FOLIO train remainder (Han et al., 2024), and BBH tracking with five or seven objects (Suzgun

et al., 2023)). It was decontaminated against the complete inventory of previously used examples. On this holdout, the unadapted base model scores 0.7145; loop-only models score 0.6991–0.7073 (mean 0.7018, -1.3 points); the pre-specified mixture arm scores 0.7155–0.7318 across five seeds (mean 0.7216, $+0.7$ points over base), with all five seed differences positive. The empirical 2.5th–97.5th percentile range of those five differences is $[0.1, 1.6]$ points; it is descriptive and not a confidence interval. The single-shot mixture (nominal 0.5 share, same realized 0.348) scores 0.7209–0.7264, and generic-only models score 0.7100–0.7273. The pre-registered analysis specified item-level paired bootstrap intervals but did not uniquely define their aggregation across training seeds. The analysis implementation used the conservative criterion that every per-seed lower bound exceed zero; all five lower bounds are nonpositive (four negative and one zero), so the criterion is not met. Independently, the $+0.7$-point estimate is below the pre-specified two-point minimal meaningful effect and therefore does not support a meaningful improvement over the base model. The pre-specified secondary comparisons meet the stated non-inferiority margin for mixture versus loop-only and the descriptive equivalence band for single-shot versus interleaved mixture, but they do not alter the primary conclusion.

A.20 Pre-specified design for a further replication

Because the mixture hypothesis remains unconfirmed, we record the design we commit to for any further confirmatory replication, to be frozen before new data are inspected. The minimum design has four arms: the untouched base model; exposure-matched static training; the repaired recursive loop (corrected checker, matched target format and exposure, active controller); and a frozen disjoint mixture (share and data partitions fixed in advance). If resources allow, we additionally run a matched no-verifier arm, a static-policy controller, and a conflict-matched mixture control. Each trained condition uses at least five—and preferably eight to ten—independent training seeds, fixed in advance, with no seeds added after results are seen. The three primary contrasts are mixture minus untouched base, recursive minus matched static, and verified minus matched no-verifier, analyzed under a pre-specified multiplicity strategy; all remaining comparisons are secondary or exploratory. Training seeds are the independent experimental unit: results are reported as paired per-seed differences with effect intervals rather than p-values, and each primary contrast carries a pre-defined equivalence or non-inferiority margin. Item-level bootstrap intervals characterize item uncertainty only and do not substitute for training-run uncertainty. The stopping rule—a fixed sample size, a group-sequential rule, or a confidence-width rule—is likewise frozen before any result is inspected.

B Reproducibility Details

The supplementary source reports (i) the test-set size and fingerprint, (ii) condition and seed names, (iii) seed-level and aggregate accuracies, (iv) the distinction between item-level and training-run uncertainty, (v) the treatment of unsuccessful runs, and (vi) the distinction between confirmatory v1 and subsequent exploratory analyses. The pre-registered replication on an independent holdout (Section A.19) follows the same reporting scheme and does not confirm the mixture gain.