TEACHING BY EXECUTION: VERIFIER-DISCOVERED LESSONS FOR AGENT-TO-AGENT TRANSFER

Anonymous authors
Paper under double-blind review

ABSTRACT

Language-agent systems routinely ask one agent to summarize experience for another, yet they evaluate the sender rather than what a frozen receiver can do next. We define *teaching utility*: the held-out behavioral improvement caused by a budgeted lesson. We introduce TEACHWORLD, a judge-free testbed of 144 latent policies, six domains, three rule structures, and three independently served model families. We then present LESSONSMITH, a training-free compiler that turns a teacher's rule and verifier feedback into explicit confirmation and correction receipts. Native agent lessons improve receivers by 44.6 percentage points over no lesson. Benchmark-supplied branch coverage adds 4.3 points (95% concept-clustered CI [2.8, 5.9]). More importantly, a preregistered adaptive verifier discovers all four action outcomes without branch labels in a median 9 executions (P95 15), retains four receipts, and improves the same native lessons by 2.5 points ([1.1, 3.8]) over 964 matched teacher–student–concept cells; all 9 model-pair estimates are positive. A strictly fresh replication excludes every earlier diagnostic and proposal state, matches adaptive accuracy within -1.3 points, and favors executed over predicted labels by 3.0 points. Matched controls show why: explicit receipts beat rewriting the rule from the same evidence by 2.7 points ([1.2, 4.2]), while teacher-chosen probes under-cover distinct outcomes. Execution thus becomes a compact teaching operation: discover behaviorally distinct cases, preserve their outcomes, and optimize the handoff for the next agent rather than the previous one.

1 INTRODUCTION

An agent finishes a long task and leaves a rule for the next agent: "authoritative records should be merged." The rule is plausible and even agrees with most of the agent's experience. It is also wrong on two boundary cases. A new agent that receives the rule confidently repeats both mistakes. This increasingly common handoff pattern appears in reflection, memory, multi-agent communication, and continual-agent systems (Shinn et al., 2023; Zhao et al., 2024; Sumers et al., 2024; Hu et al., 2026). These systems assume that successful experience can be converted into instruction. The assumption has rarely been tested directly.

The missing quantity is the receiver's gain from the sender's experience. A teacher succeeds when its message changes a student's future behavior on states neither agent has used to construct the message. A lesson can therefore be fluent but uninformative, locally correct but misleading, or compact and behaviorally decisive. This distinction connects language agents to machine teaching (Goldman & Kearns, 1995; Zhu et al., 2018), while introducing a distinctive setting: both teacher and student are general-purpose, frozen language models, and the teaching signal is natural language plus executable experience rather than a parameter update.

We operationalize this distinction with TEACHWORLD. Each concept is a latent policy over structured states and four legal actions. A teacher receives 12 verified decisions, writes a compact rule, selects four examples, and predicts four new diagnostic cases. A separate frozen student receives exactly one lesson and acts on six held-out states that are disjoint from both teacher history and diagnostics. The benchmark crosses 144 concepts, six semantic domains, three policy structures, three teacher families, and three student families. Because every outcome is executed by a deterministic environment, lesson content and downstream behavior can be evaluated without an LLM judge.

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.

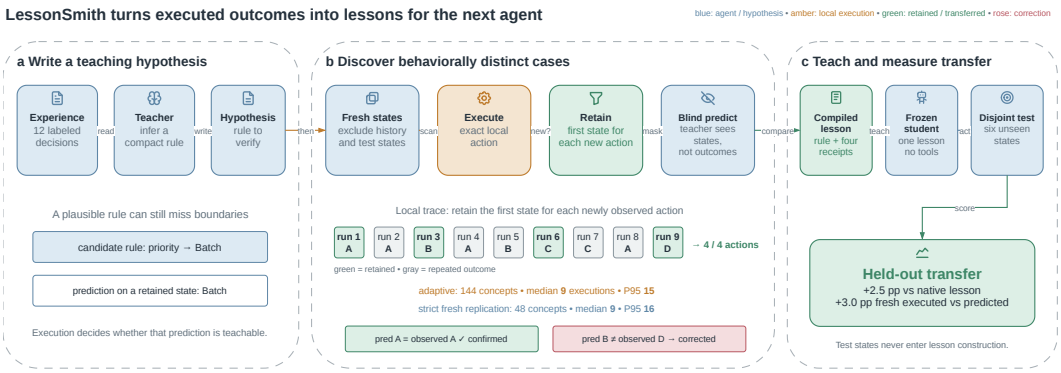

Figure 1: **LESSONSMITH discovers what is worth teaching, then preserves execution.** After a teacher writes a candidate rule, a local verifier scans fresh legal states and retains the first state producing each new action. The teacher predicts those states without outcomes; exact labels turn the predictions into confirmation or correction receipts before a frozen receiver acts on six disjoint states.

Our method, LESSONSMITH, changes the verifier's role from scoring or revising the originating agent to compiling execution feedback into a receiver-facing lesson. A controlled instantiation retains one diagnostic per latent branch. A deployable instantiation shuffles legal states, executes them locally, and retains the first state producing each unseen action—without reading branch labels. Both pair the same frozen candidate rule with explicit confirmations and corrections. Figure 1 shows the complete handoff.

The evidence gives a clear answer. Native lessons cause a 44.6 pp held-out gain. Controlled branch coverage adds 4.3 pp; branch-label-free adaptive coverage preserves a significant 2.5 pp gain, at a median of only 9 local executions. A matched no-execution control attributes 5.2 pp to corrected labels; a strictly fresh candidate-space replication preserves the direction at +3.0 pp. Explicit receipts beat an execution-informed rule rewrite by 2.7 pp. Our contributions are:

- We define **teaching utility** as a causal held-out behavioral effect on a frozen receiver, making the quality of an agent handoff directly measurable.
- We introduce **TEACHWORLD**, containing 144 fresh policy instances with 124 distinct family/structure signatures, 864 held-out states per condition, exact execution labels, and a full cross-model teacher–student protocol.
- We present **LESSONSMITH**: controlled covered lessons improve native teaching by 4.3 pp, while an adaptive verifier discovers coverage without branch labels and retains a 2.5 pp gain; a fresh-state replication independently recovers its accuracy and mechanism direction.
- We isolate two load-bearing principles: executed evidence should remain explicit, and the verifier should search for behaviorally distinct outcomes rather than rely on unguided teacher selection.

2 TEACHING UTILITY AS A RECEIVER-SIDE EFFECT

Let concept c be a policy $f_c : \mathcal{X}_c \to \mathcal{A}_c$. A teacher T observes verified experience E_c, then emits lesson $L_T(E_c)$. A frozen student S receives the lesson as context and predicts actions on query set Q_c, where $Q_c \cap E_c = \emptyset$. We define

$$U_{\text{teach}}(T, S, c) = \text{Acc}(S(Q_c \mid L_T(E_c)), f_c(Q_c)) - \text{Acc}(S(Q_c \mid \varnothing), f_c(Q_c)) . \quad (1)$$

The student model, decoding, legal actions, prompt format, and query states are matched; only the lesson changes. Equation 1 is therefore the intervention effect of the teaching artifact in this protocol.

We measure solving utility on a separate verifier set V_c: the teacher predicts $f_c(V_c)$ before seeing execution results. Solving and teaching can diverge. For an ordered policy over k binary predicates,

Table 1: **TEACHWORLD exposes scale at the policy-instance level.** Every instance has its own surface realization and disjoint history, diagnostic, candidate, and test states. The follow-up matrices evaluate three new conditions across every model pair.

Quantity	Count
Semantic domains / rule structures	6 / 3
Policy instances / distinct structural signatures	144 / 124
Teacher history records	1,728
Branch-covered diagnostic states / receipts	576 / 1,728
Fresh agent-selection candidates	1,152
Held-out states per condition	864
Teacher families / student families	3 / 3
Teacher–concept artifacts	432
Follow-up condition cells observed	2591 / 2592
Adaptive task cells observed	1293 / 1296
Strictly fresh matched cells observed	864 / 864
Legal actions per concept	4

two endpoint examples leave all $k!$ priority orders consistent. A contrastive chain that activates one predicate at a time identifies the order in $k + 1$ executions. Both lesson creators can solve their observed endpoints perfectly; only the latter communicates enough structure for reliable transfer. Appendix A gives the construction.

This formulation makes three choices deliberate. First, we score actions rather than lesson fluency. Second, the evaluation set is disjoint from all lesson construction. Third, students are frozen: improvement comes from communication at inference time, not from hidden fine-tuning.

3 TEACHWORLD: CONTROLLED AGENT TEACHING

3.1 CONCEPT CONSTRUCTION

TEACHWORLD instantiates policies in six domains that are recognizable to language agents but carry no external answer key: configuration precedence, research-result triage, document workflow recovery, record reconciliation, compute-job routing, and data-batch planning. Each domain contains 24 seeded policy instances. Action names, feature roles, predicate priorities, action permutations, and thresholds vary across instances, preventing a vocabulary-level shortcut. The 144 instances realize 124 distinct family/structure signatures; the largest repeated signature occurs 3 times, while every instance retains its own surface vocabulary and action map.

The concepts realize three structural families. *Ordered* policies return the action of the first active predicate. *Exception* policies combine conjunction, negation, and exclusive-or branches. *Threshold* policies combine numeric boundaries with Boolean gates. Every policy has four outcome branches and exact local execution.

For each concept, we sample 12 teacher-history states, three from each branch; four diagnostic states, one per branch and selected near decision boundaries; and six branch-stratified held-out states. The sets are state-disjoint. The teacher sees history labels and diagnostic inputs but not diagnostic outputs. The student sees a lesson but never the history/probe split metadata or held-out outputs.

For the agent-selection experiment, we additionally sample eight fresh unlabeled states per concept, disjoint from all three sets. These candidates are drawn without branch labels. A teacher predicts all eight and selects four for execution, allowing us to separate the value of branch-covered evidence from the ability to discover such evidence.

3.2 THE TEACHING PROTOCOL

For each concept, the teacher must (i) write a rule in at most 55 words, (ii) select exactly four experience identifiers to teach a new model, and (iii) predict the action on each diagnostic state. A

native teacher lesson contains the rule and selected labeled examples. Teacher solving accuracy is its diagnostic prediction accuracy.

The frozen student receives one independently delimited concept at a time and returns the six held-out action names. No agent receives tools at this stage, and scoring is exact after case normalization. Batched transport amortizes serving overhead, but each concept contains its own legal actions and lesson; uncertainty is computed at the concept level rather than treating nested decisions as independent.

4 LESSONSMITH: COMPILE, EXECUTE, TEACH

LESSONSMITH treats the teacher's output as a hypothesis to be made useful rather than a message to accept verbatim. Its three operations are intentionally lightweight.

Preserve the abstraction. The candidate rule carries structure that four examples alone often fail to express. LESSONSMITH keeps that text intact so that verification repairs evidence without silently rewriting the teacher's intended generalization.

Execute branch-covering diagnostics. The compiler executes four fresh cases selected before teacher inference: one case for each outcome branch, with boundary proximity used as a deterministic tie-breaker. The probe labels never come from the teacher or a language-model judge. In this controlled study, the environment uses its ground-truth branch partition to select those cases. LESSONSMITH therefore targets agent settings that expose an executable verifier and a branch-coverage mechanism; the experiment cleanly estimates the value of turning their outcomes into a teaching artifact.

Discover coverage from verifier outcomes. The adaptive instantiation removes the latent partition. It shuffles the legal state space with a frozen seed, excludes teacher history and evaluation states, executes up to 32 cases, and retains the first case yielding each previously unseen legal action. The teacher then predicts only those four retained states without seeing their outcomes; the compiler marks each prediction confirmed or corrected. Selection uses observable verifier behavior, not a branch label, model proposal, or student response. A fresh-state instantiation additionally excludes every earlier diagnostic, candidate, and retained receipt before repeating the search and matched label control.

Make prediction errors local and explicit. Each diagnostic record shows the observed action and either "prediction confirmed" or "teacher predicted a; execution corrected it." The student can integrate the broad rule with the exact boundary that refutes it. The procedure requires no gradient update, no student access, and four environment calls per concept.

The running example in Figure 1 illustrates the result. A teacher's rule incorrectly makes authority a global gate. Two diagnostic executions preserve its correct branches, while two corrections expose interactions with group completeness and key presence. The final student receives the compact hypothesis together with the minimum concrete evidence needed to revise it.

5 EXPERIMENTAL DESIGN

5.1 MODELS AND COVERAGE

We use DeepSeek V4 Flash, GLM 5.3 Flash, and GPT-5.6 Luna through one OpenCode harness, accessed September 16, 2026. All three act as teachers on all 144 concepts. DeepSeek and Luna act as students on the complete benchmark; GLM supplies a stratified 36-concept student replication containing six concepts from every semantic domain. This design gives complete teacher coverage, two full independent student families, and a third cross-family check while reserving premium models for review rather than data generation.

After freezing and reviewing the main study, we preregistered two discriminating mechanism conditions and the expanded solving analysis. For these conditions, all three students run all 144 concepts

against all three teachers: 2591 of 2592 planned teacher–student–concept cells are observed (99.96%). This full 3×3 follow-up matrix prevents the mechanism conclusions from depending on the original stratified GLM slice.

We then preregistered adaptive coverage with seed 92731, a 32-execution cap, and a $3.50 incremental soft cap. It observes 1293 of 1296 cells (99.77%) and covers every cell available for the frozen native and covered comparisons. The three missing transports are outside those baseline slices and are not imputed.

Finally, a preregistered replication on all 48 threshold concepts excludes history, held-out states, original diagnostics, the agent-selection pool, and every round-3 receipt; both matched conditions observe all 864 cells.

All calls use deterministic decoding, the same action-only output contract, six test decisions per concept, a 420-second timeout, and deterministic narrow JSON repair. Incomplete transport batches are subdivided to one concept when needed; no failed prediction is relabeled as correct. Prompts, event streams, parsed answers, versions, elapsed time, token counts, and costs are retained.

The primary comparison contains 963 of 972 planned teacher–student–concept observations (99.1%). We retain every complete concept answer from partially returned batches and omit only concepts for which transport produced no action list. The matched execution ablation contains 840 of 864 observations (97.2%).

5.2 CONDITIONS

The evaluation separates message size, structure, and verification:

- **No lesson**: legal actions and held-out states only.
- **Endpoints (2)** and **random examples (4)**: small familiar experience controls, evaluated on a six-concepts-per-domain slice.
- **Verified examples (4)**: the four executed diagnostic cases without a teacher rule; this isolates evidence from abstraction.
- **Raw history (12)**: the full unordered teacher experience.
- **Native teacher lesson**: the teacher rule plus its four selected history examples.
- **Agent-selected executed lesson**: the teacher predicts eight fresh unlabeled cases, chooses four to test and teach its candidate rule, and receives their execution receipts. No true branch label is exposed during selection.
- **Execution-revised rule**: after seeing the same four branch-covered outcomes used by LESSONSMITH, the teacher rewrites its rule in at most 55 words; the student receives that revised rule alone.
- **Unexecuted diagnostic lesson**: the same candidate rule and diagnostic inputs as LESSONSMITH, but paired with the teacher's predictions. This matched ablation isolates execution correction.
- **Covered LESSONSMITH**: candidate rule plus four benchmark-supplied, branch-covered execution receipts.
- **Adaptive LESSONSMITH**: candidate rule plus four distinct-action receipts discovered by local execution without branch labels.
- **Oracle rule**: exact environment policy, serving as a ceiling.

5.3 METRICS AND STATISTICAL PROTOCOL

Our primary metric is held-out action accuracy. The primary contrast is paired LESSONSMITH minus native teacher lesson; secondary contrasts use no lesson, raw history, verified examples, and the unexecuted control. We average repeated teacher–student observations within concept and bootstrap concepts 10,000 times for 95% intervals. This preserves the benchmark's independent unit and avoids pseudo-replication over six queries. We additionally report teacher diagnostic accuracy, exact four-probe pass rate, same- versus cross-model transfer, rule-family slices, model cost, intervals

Table 2: **Main result: verifier-compiled lessons improve held-out transfer.** Paired deltas and intervals cluster at the concept level. Covered LESSONSMITH uses one benchmark-supplied case per latent branch; adaptive LESSONSMITH discovers distinct outcomes from verifier results without branch labels.

Lesson	Rule	Receipts	Coverage source	Accuracy (%)	Paired effect (pp)
No lesson	–	0	–	28.1	–
Raw history	–	12	teacher history	71.3	–
Native teacher	yes	4	teacher history	72.2	+44.6 vs. none
Unexecuted diagnostic	yes	4	latent branch	71.2	covered +5.2
LESSONSMITH-covered	yes	4	latent branch	76.7	+4.3 vs. native
LESSONSMITH-adaptive	yes	4	verifier outcomes	74.9	+2.5 [1.1, 3.8]
Oracle rule	exact	0	–	99.6	–

for every model pair and domain, leave-one-domain-out estimates, a bootstrap clustered over the 124 latent signatures, and adversarial missing-record sensitivity. Lesson length uses a deterministic model-agnostic lexical tokenizer.

The follow-up protocols freeze all original prompts, models, held-out states, and metrics. Its preregistered contrasts are agent-selected execution versus native and branch-covered lessons, execution-revised rule versus native teaching, and LESSONSMITH versus execution-revised rule. We also combine the original four probes with eight new teacher predictions for a 12-item solving battery, estimate cross-form reliability, and compare within-concept rank inversions with a permutation null. These are reported as follow-up mechanism evidence rather than retroactively redefining the primary test. The adaptive primary contrast is paired against the native lesson on every available baseline cell; adaptive versus covered lessons quantifies the value of boundary-targeted probes after removing branch-label dependence. The fresh replication compares executed and predicted labels on otherwise identical states, teachers, students, and prompts.

6 RESULTS

6.1 AGENT LESSONS CAUSE LARGE HELD-OUT LEARNING GAINS

Native teacher lessons improve frozen-student accuracy by 44.6 pp over no lesson (95% CI [40.6, 48.5]). All six scored states are disjoint from the 12 history records and four diagnostics, establishing transfer beyond recognition of previously shown items. The result shows that compact, agent-generated lessons transfer latent policies to another model.

Controlled covered LESSONSMITH raises accuracy by a further 4.3 pp relative to the same teacher's native lesson (95% CI [2.8, 5.9]). It also exceeds the full 12-example raw history by 6.4 pp (95% CI [3.6, 9.3]). On the two student families with complete raw-history coverage, the gain is 6.7 points [3.9, 9.7]. The compiled lesson has a median 184 lexical tokens versus 354 for raw history, a 48.0% reduction. Thus four diagnostic executions plus a rule transfer better with substantially less context than twelve unordered examples.

The matched no-execution condition carries the identical candidate rule and diagnostic inputs, but pairs them with the teacher's predicted actions. Replacing those predictions with observed actions and explicit receipts raises held-out accuracy by 5.2 pp (95% CI [3.6, 6.9]). This isolates execution correction from diagnostic case choice and prompt budget. The primary result remains positive under two conservative checks: clustering the bootstrap by 124 structural signatures yields [2.8, 5.8] points, and assigning the theoretical worst effect (-100 points) to every one of nine missing primary pairs still leaves 3.4 points. Leave-one-domain-out estimates span 3.3–5.1 points.

Finding. A frozen language agent can learn a new decision policy from another agent's compact lesson, and executing the lesson before handoff yields a reliable additional gain.

Table 3: **Follow-up comparisons use exact paired subsets.** Both accuracies in a row are recomputed on the same teacher–student–concept cells; intervals bootstrap concepts.

Comparison	A acc.	B acc.	Pairs	A−B (pp)
Agent probes − native	70.2	72.4	963	-2.2 [-3.7, -0.7]
Revised rule − native	74.0	72.4	964	+1.6 [-0.4, 3.6]
LESSONSMITH − revised rule	76.8	74.1	971	+2.7 [1.2, 4.2]
LESSONSMITH − agent probes	76.8	70.2	970	+6.6 [4.8, 8.3]
Adaptive − native	74.9	72.4	964	+2.5 [1.1, 3.8]
Covered − adaptive	76.8	74.9	971	+1.9 [0.3, 3.5]
Fresh executed − predicted	70.2	67.2	432	+3.0 [-0.5, 6.6]

6.2 VERIFIER-DISCOVERED COVERAGE TURNS THE CONTROL INTO A PROCEDURE

The full-matrix follow-up separates two alternatives that the main condition ladder leaves open. The first ablation tests whether execution helps by improving only the teacher's rule. The execution-revised rule improves over the native lesson by 1.6 pp (95% CI [-0.4, 3.6]), but this interval crosses zero. Giving the student the branch-covered receipts explicitly is stronger: LESSONSMITH exceeds the rewrite by 2.7 pp ([1.2, 4.2]). Externalized evidence therefore survives transfer better than compressing the same observations back into prose.

Second, perhaps a teacher can find the useful checks itself. The eight-case candidate pool contains 2.65 of four branches on average. Uniform random four-of-eight selection covers 2.14 branches; teachers improve this to 2.40. Even when all four branches are present in the pool, their choices achieve full coverage in 12 of 54 teacher–concept opportunities (22.2%, versus 15.4% in expectation). The resulting lessons trail native lessons by -2.2 pp ([-3.7, -0.7]) and trail branch-covered LESSONSMITH by 6.6 pp ([4.8, 8.3]). Teachers select better-than-random coverage, yet the paired outcome implicates incomplete probe discovery as a bottleneck: local execution is effective once informative cases are available, while four choices from a random candidate pool do not reliably expose every decision branch. Conditional analysis localizes the failure: when teacher selection reaches all four branches, its transfer difference from covered LESSONSMITH is only +0.2 points, although the stratum is small (12 teacher–concept artifacts; 95% CI $[-8.1, 8.3]$). Coverage, rather than teacher identity or added prose, is the operative variable.

The preregistered adaptive verifier converts that diagnosis into a procedure. It scans a frozen random order of legal states, executes locally, and retains the first state producing each unseen legal action. It reads no latent branch ID and excludes both teacher history and student held-out states. Across all 144 concepts it reaches four outcomes in a median 9 executions (P95 15, maximum 32), then hands the student only four receipts. Adaptive LESSONSMITH improves matched native lessons from 72.4% to 74.9%: +2.5 pp [1.1, 3.8] over 964 cells, with all 9/9 model-pair estimates positive. The boundary-proximal controlled probes retain a 1.9 pp advantage [0.3, 3.5], quantifying the remaining value of targeted search rather than leaving deployability as an unspecified oracle.

The strictly fresh replication excludes every earlier diagnostic, proposal, and receipt, yet again finds all four outcomes (median 9 executions; P95 16) and completes all 432 cells. Executed receipts reach 70.2% versus 67.2% for matched predicted labels: +3.0 pp [-0.5, 6.6], with 8/9 positive model-pair estimates. Accuracy matches the full adaptive run within -1.3 pp [-4.7, 1.9].

6.3 TEACHING TRANSFERS ACROSS MODEL BOUNDARIES

The cross-agent matrix localizes the improvement to transferable teaching. Cross-model LESSONSMITH lessons reach 77.0% accuracy, compared with 75.7% for same-model pairs; their respective gains over native lessons are 4.7 and 3.6 points. Every teacher family improves at least two student families, eight of nine model-pair estimates are positive, and six model-pair intervals exclude zero. The strongest pairing is cross-model. This matters operationally: lesson production does not require knowing which model will consume the artifact.

All three rule structures benefit (Table 4). The largest gain appears for exception policies, where locally plausible verbal rules most often miss interacting conditions. Ordered policies begin near the oracle ceiling and still improve. Domain slices follow the same pattern: LESSONSMITH leads

Table 4: **Verification gains survive structural change.** Accuracy on paired teacher–student observations in the frozen complete-case matrix.

Rule structure	Native	LESSONSMITH	Gain
Ordered precedence	93.2	95.1	+1.9
Exception rules	54.9	62.4	+7.5
Numeric thresholds	69.2	72.8	+3.6

native lessons in five of six families in the complete-case analysis and produces its largest gains on reconciliation and workflow recovery.

6.4 SENDER ACCURACY PREDICTS TEACHING; RECEIPTS DETERMINE REPAIR

Teachers predict diagnostic actions with 73.1% mean accuracy, yet only 39.1% of teacher–concept rules pass all four probes. Across 432 teacher–concept artifacts, the original four-item solving score correlates with native teaching lift at $\rho = 0.53$. Expanding the score to 12 fresh predictions raises the estimate to $\rho = 0.56$ (Pearson $r = 0.54$); a reliability-adjusted estimate is $r = 0.81$ (bootstrap 95% CI [0.69, 0.94]). Solving competence is therefore a strong ingredient of teaching rather than a separate, unrelated trait.

Finite solver batteries are not reliable teacher rankings. The original four-item and fresh eight-item forms reverse 46 of 168 comparable within-concept orders (27.4%), nearly the same scale as their disagreement with teaching. We therefore use solver accuracy as a predictive diagnostic, not as evidence for a latent trait split. The causal student-side outcome remains the target: it directly tests whether the artifact transfers, and the matched compilation interventions identify which information makes that transfer improve.

Native lesson accuracy is 92.3% when all four predictions are correct, but falls sharply as probe errors accumulate. LESSONSMITH is self-limiting: with zero errors, it changes accuracy by 0.0 points [-0.8, 0.8]; with one and two errors it adds 4.8 [2.2, 7.4] and 8.7 [5.5, 12.0] points. The zero-error result is expected because no corrective label is introduced; the informative behavior is that confirmation carries no penalty while available corrections transfer reliably. The three- and four-error strata contain 33 and 9 artifacts and therefore have wider intervals (Figure 2). As expected from this design, probe accuracy correlates negatively with repair gain ($\rho = -0.28$): verification concentrates its signal where the teacher's hypothesis needs it.

Finding. Executable verification is a selective teaching operator: it preserves an already correct lesson and concentrates positive lift on artifacts with correctable prediction errors.

6.5 COST AND EFFICIENCY

Usable original records cost $6.92; mechanism, adaptive, and fresh follow-ups add $4.94, $3.29, and $1.67, while all persisted calls total $18.04. Generation, execution, scoring, and bootstrap run locally on CPU. Controlled LESSONSMITH adds four local executions; adaptive LESSONSMITH adds one teacher prediction call plus the unchanged student call, as does its matched control. Adaptive search takes a median 9 executions, exposes four receipts, and remains 48.0% shorter than raw history.

Reproducibility. The anonymous artifact includes all four seeds, prompts, raw events, exact model IDs, task rows, analysis/figure scripts, environment lock, checksums, and replay map; it rebuilds every reported number and quantitative figure. The review artifact is available at https://anonymous.4open.science/r/lessonsmith/.

7 RELATED WORK

Agents that learn from experience. Reflexion, ExpeL, Voyager, and memory-centric architectures turn trajectories into reusable verbal or procedural state (Shinn et al., 2023; Zhao et al., 2024; Wang et al., 2023; Sumers et al., 2024). MemoryAgentBench and AhaBench test whether stored or supplied

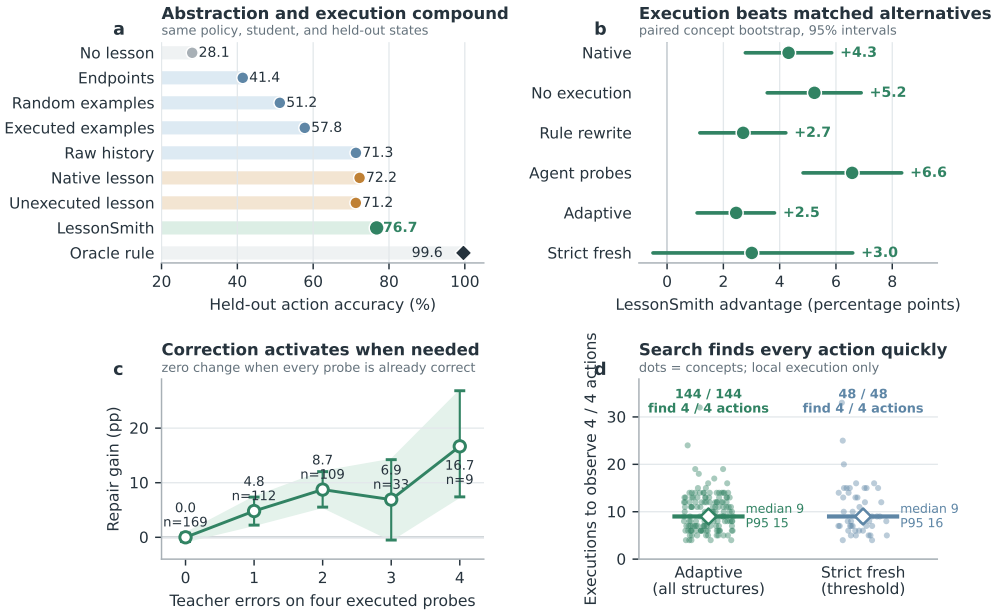

Figure 2: **The evidence chain from accuracy to mechanism. (a)** The controlled ladder separates examples, abstraction, and execution. **(b)** Six paired contrasts isolate coverage, explicit receipts, adaptive discovery, and the strict-fresh control; intervals bootstrap concepts. **(c)** Correction is self-limiting: it is neutral when predictions are confirmed and activates when execution corrects them. **(d)** Every adaptive and strict-fresh concept reaches all four observable outcomes with a compact local execution budget while exposing exactly four receipts.

experience helps later (Hu et al., 2026; Cheng et al., 2026). We study the upstream handoff: compiling a hypothesis with execution so another frozen model improves on unseen states, complementing environment preprocessing (Samuel et al., 2026).

Communication and multi-agent reasoning. Multi-agent work evaluates messages on the current problem (Rizvi-Martel et al., 2026; Liu et al., 2026; Gao et al., 2026); we evaluate future states. AgentFold compresses one trajectory (Ye et al., 2026), whereas LESSONSMITH transfers across models. DISCO selects evaluation items via disagreement (Rubinstein et al., 2026); our contrasts are selected to teach.

In-context learning and machine teaching. In-context examples induce behavior (Brown et al., 2020; Min et al., 2022; Wei et al., 2022). Curriculum and active learning choose learner data (Bengio et al., 2009; Settles, 2009); machine teaching optimizes it (Goldman & Kearns, 1995; Zhu et al., 2018). We make that objective executable for heterogeneous language agents.

8 DEMONSTRATED SCOPE AND EXTENSIONS

TEACHWORLD establishes causal, judge-free comparisons with exact executors, balanced branches, and localized states. Adaptive search removes branch membership, while the strict-fresh replication isolates transfer on entirely new diagnostics. The measured 1.9 pp gap identifies learned proposal, continuous-state abstraction, and repeated handoffs as the next extensions beyond the single bounded handoff studied here.

9 CONCLUSION

Execution turns agent experience into a transferable lesson. LESSONSMITH searches for behaviorally distinct cases, preserves their exact outcomes, and improves held-out receiver behavior across all nine

model pairings. The pattern survives a strictly fresh candidate space: execute before handoff, and teach the evidence rather than only prose.

REFERENCES

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *International Conference on Machine Learning*, 2009.

Tom B. Brown, Benjamin Mann, Nick Ryder, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.

Zerui Cheng, Jiawei Xu, Huacan Chai, Jiayang Sun, Pramod Viswanath, and Maxm Pan. AhaBench: Do agents learn from prior experience? a benchmark for long-horizon continual learning. *arXiv preprint arXiv:2609.05435*, 2026.

Jianxin Gao, Tianyi Yu, Linna Deng, Runze Li, and Zining Wang. Testing interchangeability in LLM agent teams. *arXiv preprint arXiv:2609.05279*, 2026.

Sally A. Goldman and Michael J. Kearns. On the complexity of teaching. *Journal of Computer and System Sciences*, 50(1):20–31, 1995.

Yuanzhe Hu, Yu Wang, and Julian McAuley. Evaluating memory in LLM agents via incremental multi-turn interactions. In *International Conference on Learning Representations*, 2026.

Jieyuan Liu, Mengzhou Hu, Jefferson Chen, JungHo Kong, Pratibha Jagannatha, Yiming Gao, Dexter Pratt, Hsin-Yuan Lee, Zhiting Hu, Trey Ideker, Wei Wang, Eric P. Xing, and Zhen Wang. HypoEvolve: Genetic algorithms enable multi-agent LLMs to discover scientific hypotheses. *arXiv preprint arXiv:2609.15938*, 2026.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Conference on Empirical Methods in Natural Language Processing*, 2022.

Michael Rizvi-Martel, Satwik Bhattamishra, Neil Rathi, Guillaume Rabusseau, and Michael Hahn. Benefits and limitations of communication in multi-agent reasoning. In *International Conference on Learning Representations*, 2026.

Alexander Rubinstein, Benjamin Raible, Martin Gubri, and Seong Joon Oh. DISCO: Diversifying sample condensation for efficient model evaluation. In *International Conference on Learning Representations*, 2026.

Vinay Samuel, Varun Ursekar, Vijay S. Kalmath, Apaar Shanker, Veronica Chatrath, and Yuan Xue. Studying without a syllabus: Task-agnostic environment preprocessing. *arXiv preprint arXiv:2609.10824*, 2026.

Burr Settles. Active learning literature survey. Technical report, University of Wisconsin–Madison, 2009.

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*, 2023.

Theodore R. Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas L. Griffiths. Cognitive architectures for language agents. *Transactions on Machine Learning Research*, 2024.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 2022.

Rui Ye, Zhongwang Zhang, Kuan Li, Yin Huifeng, et al. AgentFold: Long-horizon web agents with proactive context folding. In *International Conference on Learning Representations*, 2026.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: LLM agents are experiential learners. In *AAAI Conference on Artificial Intelligence*, 2024.

Xiaojin Zhu, Adish Singla, Sandra Zilles, and Anna N. Rafferty. An overview of machine teaching. *arXiv preprint arXiv:1801.05927*, 2018.

A Identification Example

Consider an ordered policy with k binary predicates, an unknown priority permutation, and $k + 1$ distinct rank actions. A lesson containing only all-false and all-true states leaves all $k!$ predicate orders consistent: the endpoints reveal the default and first-rank actions but no internal order. Starting from all-false and activating predicates one at a time from lowest to highest true priority yields a chain of $k + 1$ states. Each transition identifies exactly one predicate rank, so the policy is uniquely identified. This construction motivates branch-covering, boundary-proximal probes in TeachWorld; exception and threshold families use the analogous one-probe-per-outcome construction.

B Algorithm

```
LESSONSMITH(T, history E, state generator G, executor f)
  rule <- T.write_rule(E)
  discovered <- []
  for v in frozen_shuffle(G excluding E):
      observed <- f(v)
      if observed is a new action: discovered.append(v, observed)
      if four actions observed or 32 executions used: break
  predictions <- T.predict(rule, states(discovered))
  lesson <- [rule]
  for (v, observed) in discovered:
      predicted <- predictions[v]
      if predicted == observed: status <- "confirmed"
      else: status <- "execution corrected prediction"
      lesson.append(v, observed, status)
  return lesson
```

The adaptive executor is deterministic and local. It observes action values but no latent branch metadata. The student and its held-out query set are never inputs to lesson construction. The controlled condition replaces the discovery loop with one benchmark-supplied state per latent branch.

C Prompt Contracts

Teacher. For each independent concept, the teacher sees its legal actions, 12 verified history records, and four unlabeled verification cases. It returns one JSON object containing a rule of at most 55 words, exactly four history identifiers, and one action prediction per verification case.

Student. The student is told that each task is an independent environment with its own policy, receives exactly one condition-specific lesson, and returns a list of six legal action names. No prose or markdown is requested. Full rendered prompts and raw event streams are included in the artifact.

Follow-up teachers. For agent selection, the frozen teacher receives its original rule plus eight fresh unlabeled cases, predicts all eight, and selects four identifiers for execution. For rule revision, it receives its original rule plus the four branch-covered predicted and observed actions, then returns one self-contained rule of at most 55 words. The student contract and held-out queries remain unchanged. For adaptive coverage, a local verifier selects four states by first-seen action diversity; the teacher

Table 5: **Domain intervals for LESSONSMITH minus native teaching.** All six point estimates are positive; leave-one-domain-out gains remain 3.3–5.1 points.

Domain	Gain (pp)	95% CI
Precedence	0.6	[-1.1, 2.2]
Research triage	3.3	[0.4, 6.3]
Workflow	9.6	[5.2, 14.0]
Reconciliation	5.0	[1.4, 8.7]
Compute routing	3.2	[-1.0, 7.3]
Batch planning	4.2	[-0.0, 8.3]

sees those states but not their outcomes, predicts each action, and the compiler attaches the saved outcomes.

D REPRODUCIBILITY DETAILS

The main benchmark uses seed 91689, the mechanism follow-up uses 92227, adaptive search uses 92731, and the strict fresh-state replication uses 93017. Policy generation precedes every model call. The temperature is zero. The runner is OpenCode 1.18.7; raw event streams preserve provider model identifiers, session IDs, timestamps, token accounting, costs, and outputs. Student answer parsing accepts a JSON object and applies only two deterministic repairs: removing one unmatched closing bracket or adding one missing final object brace in the main study. The follow-up additionally normalizes semantically identical Q1–Q6 objects and one-task lists. Batches that still fail are subdivided into one-concept calls. Accuracy is computed from exact case-normalized action equality. The mechanism follow-up observes 2591/2592 task cells. The adaptive matrix observes 1293/1296 cells and matches all 964 native-baseline cells; the three remaining transports lie outside the frozen baseline slices and are recorded rather than imputed. The fresh-state matrix observes 864/864 cells. It uses only single-concept Q1–Q6 normalization for two complete responses whose JSON keys did not follow the requested concept-key wrapper; raw generations remain unchanged.

The main analysis first averages observations within concept for every contrast, then resamples concepts with replacement 10,000 times. The 2.5 and 97.5 percentiles form the reported interval. Derived CSV files contain task-level decisions, model-pair summaries, family and rule-class slices, mechanism groups, and a deduplicated cost audit.

E ROBUSTNESS DETAIL

The primary contrast covers 963 of 972 planned pairs. Its adversarial missing-record bound remains +3.4 points. A second bootstrap treats each of the 124 family/structure signatures as a cluster and returns [2.8, 5.8] points. The raw-history comparison is fully observed for the two full-coverage student families and remains +6.7 points [3.9, 9.7]. Tables 5– 7 give the complete generalization and coverage audit.

The follow-up task matrix is 99.96% complete. Paired concept bootstraps give +2.7 points [1.2, 4.2] for explicit receipts over rule revision and +6.6 points [4.8, 8.3] for branch-covered over agent-selected receipts. The adaptive experiment is 99.77% complete, matches every available native and covered baseline cell, and yields +2.5 points [1.1, 3.8] over native lessons. Every one of the 9 model-pair estimates is positive. The 12-item solving analysis uses 432 teacher–concept units. Its two component forms disagree on 27.4% of comparable within-concept rankings, so the paper does not use their residual rank difference as evidence of construct separation. Exact enumeration of all 70 four-of-eight subsets per candidate pool supplies the random coverage baseline. The reliability-adjusted solving–teaching correlation is 0.81 [0.69, 0.94]. The strict fresh-state executed-minus-predicted comparison is +3.0 points [-0.5, 6.6] over 432 pairs, and fresh executed accuracy matches the prior adaptive run within -1.3 points [-4.7, 1.9].

Table 6: **Model-pair generalization.** Gain is LESSONSMITH minus the matched native lesson on the same concepts; six of nine intervals exclude zero.

Student	Teacher	Concepts	Gain (pp)	95% CI
DeepSeek V4 Flash	DeepSeek V4 Flash	144	2.1	[-0.2, 4.6]
DeepSeek V4 Flash	GLM 5.3 Flash	138	6.6	[4.0, 9.3]
DeepSeek V4 Flash	GPT-5.6 Luna	144	3.5	[0.2, 6.6]
GLM 5.3 Flash	DeepSeek V4 Flash	36	6.0	[2.3, 10.2]
GLM 5.3 Flash	GLM 5.3 Flash	36	2.3	[-1.9, 6.9]
GLM 5.3 Flash	GPT-5.6 Luna	33	-1.0	[-8.6, 6.6]
GPT-5.6 Luna	DeepSeek V4 Flash	144	4.5	[2.0, 7.1]
GPT-5.6 Luna	GLM 5.3 Flash	144	5.3	[3.0, 7.8]
GPT-5.6 Luna	GPT-5.6 Luna	144	5.4	[2.5, 8.3]

Table 7: **Observed condition coverage.** Coverage counts complete concept-level student outputs; the headline paired analysis uses only matched records.

Condition	Planned	Observed	Coverage
No lesson	324	324	100.0%
Endpoints	108	103	95.4%
Random examples	108	97	89.8%
Raw history	324	300	92.6%
Verified examples	324	300	92.6%
Native teacher	972	964	99.2%
LESSONSMITH	972	971	99.9%
Unexecuted	864	840	97.2%
Oracle rule	324	324	100.0%

F AI USAGE STATEMENT

Language models are the objects of study and generate the teacher and student outputs reported in the experiments. AI coding assistants also supported literature triage, experiment implementation, manuscript drafting, and constructive review. The authors designed the study, executed and audited all experiments, verified every reported value against saved artifacts, checked citations, and take responsibility for the paper.