

From Credit Assignment to Repair: A Leakage-Controlled Diagnostic Study of Tool-Using Agents

Shaobo Qiao

Euler AI Pty Ltd

Sydney, Australia

shaobo.qiao@eulerai.au

Abstract

End-of-trajectory rewards do not identify the decision associated with a tool-agent failure. We treat step-level credit assignment as failure localization and test whether it supports selective repair. Silver labels come only from completed training trajectories. Prefix-only classifiers use task-grouped model selection and remain frozen during evaluation. On the retail domain of τ^2 -bench, the selected classifier reaches 46.4% out-of-fold F1 and 61.1% F1 on 60 held-out trajectories comprising 333 scored decisions. The frozen gate selects 101 decisions, but the verifier changes actions in only seven trajectories. Their binary rewards match task-matched baselines, although independent rollouts prevent causal attribution. Airline and telecom produce no changed trajectory, so their differences estimate rollout noise rather than method effects. Every task-clustered reward interval includes zero. An exact telecom replay reproduces 60 baseline trajectories but triggers no intervention, validating replay fidelity rather than repair. The intervention funnel diagnoses where localization, verification, and treatment fail, but seven treated trajectories are insufficient to establish a general localization–repair gap.

1 Introduction

Language-model agents combine dialogue with external actions (Yao et al., 2023; Schick et al., 2023). One incorrect argument or premature write can alter every later observation. Benchmarks such as τ -bench and τ^2 -bench score final database states and conversation outcomes (Yao et al., 2025; Barres et al., 2025). Their rewards identify failed trajectories, not the decisions associated with failure.

This is a credit-assignment problem. Reinforcement learning propagates delayed return to earlier actions through value estimation or temporal-difference updates (Sutton and Barto,

2018). Agent-training systems and process reward models instead derive local signals from trajectories (Luo et al., 2025; Lightman et al., 2023; Chae et al., 2025). Tool agents add a leakage risk because benchmark evaluators may expose privileged reference actions. A test-time controller must not read them.

We ask three questions. Can training-only evidence localize failures on unseen tasks? Does a frozen localization model improve reward when it gates a conservative verifier? Can any observed difference be attributed to an executed intervention? Figure 1 summarizes the protocol.

The experiments separate localization from repair. The retail model identifies silver-labeled steps on held-out tasks, but reward differences have wide task-clustered intervals. Most selected actions remain unchanged. The seven treated retail trajectories have the same binary reward as their task-matched baselines, but they do not form replay-controlled pairs. Exact telecom replay removes pre-intervention variation, yet the frozen gate selects no step. A same-prefix ranking probe distinguishes some alternatives under grouped validation, but candidate generation is expensive and often produces no independent alternative.

Our contribution is a leakage-controlled intervention-funnel protocol for evaluating the path from localization to repair. It combines train-only silver labels, task-grouped credit-model selection, explicit treatment accounting, and exact candidate/user replay. Applied to three τ^2 -bench domains, the protocol exposes three distinct bottlenecks: threshold shift, verifier preservation, and scarcity of grounded alternatives. The credit classifier is an auditable component of this protocol rather than the primary methodological contribution. A train-only same-prefix ranking probe tests the last bottleneck without changing the main claim.

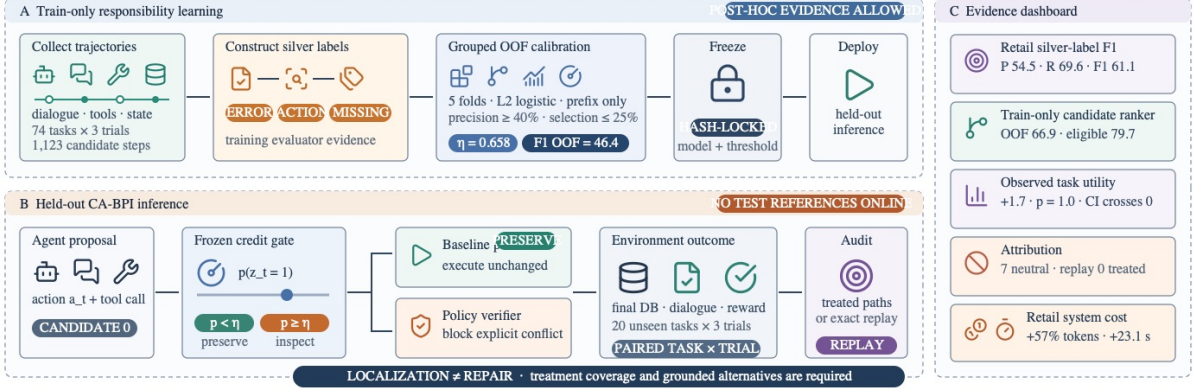


Figure 1: Training and evaluation protocol. Post-hoc evidence labels training trajectories only. Task-grouped out-of-fold (OOF) validation selects a prefix-only model and threshold. The frozen gate receives no test reference. Localization, task reward, treated outcomes, exact replay, and systems cost remain separate endpoints.

2 Related Work

Tool-agent evaluation. ReAct interleaves reasoning and action (Yao et al., 2023); Toolformer studies self-supervised tool use (Schick et al., 2023). τ -bench evaluates policy-following agents through simulated users, tools, and final-state checks (Yao et al., 2025). τ^2 -bench adds dual-control tasks, in which both the agent and user can change state (Barres et al., 2025). We use its official execution stack to study localization and intervention.

Process supervision and agent credit. Process supervision labels intermediate steps rather than complete outputs (Lightman et al., 2023). Web-Shepherd trains a process reward model from step-level preference pairs and applies it to web-agent search (Chae et al., 2025). Agent Lightning defines a trajectory interface and a credit module for agentic reinforcement learning (Luo et al., 2025). Our model is narrower: it predicts retrospective responsibility and does not update the policy.

Verification and self-correction. Reflexion uses verbal feedback across attempts (Shinn et al., 2023); Self-Refine generates and applies feedback iteratively (Madaan et al., 2023). Test-time correction can also alter correct behavior and add inference cost. We therefore record every executed change and preserve the baseline action unless the verifier identifies a concrete conflict.

3 Problem and Evaluation Targets

An episode contains histories and agent actions $\tau = (h_0, a_0, \dots, h_T, a_T)$ and ends with reward $R(\tau) \in \{0, 1\}$. For a candidate step t , a post-hoc

training procedure assigns a silver responsibility label $z_t \in \{0, 1\}$. The predictor

$$p_\theta(z_t = 1 \mid \phi(h_t, a_t)) \quad (1)$$

uses only features available at the current prefix. At test time, the gate selects an action when the score exceeds a frozen threshold η . The online controller cannot read z_t , evaluator output, or reference actions. Silver-label agreement measures reproduction of the labeling rule, not human-validated causal responsibility.

We evaluate four targets. *Localization* compares predictions with offline silver labels. *Observed task utility* compares independent baseline and method rollouts on matched task-trial identifiers. *Treated outcome* restricts this comparison to trajectories with an executed intervention. *Replay-controlled attribution* reuses candidate 0 and user responses until the first intervention. Replay removes pre-treatment model and simulator variation. A run without treatment validates replay fidelity but cannot estimate a treatment effect.

4 Method

4.1 Train-Only Responsibility Labels

We label a trajectory only after it ends. Successful trajectories supply negative examples. For a failed trajectory, an errored tool result labels the call that produced it. For each unmatched evaluator action, the last executed call with the same tool name is labeled as an argument error; if no such call exists, the final assistant step is labeled as an omission. A state-changing call absent from the evaluator’s expected actions is labeled as unexpected. When all

action checks pass but the database or language assertion fails, the procedure labels the final relevant tool or assistant step. If several candidates remain, the earliest localized message defines the root label. This deterministic rule uses official evaluator action checks during training. Test-time code cannot access them.

The rule yields silver labels, not causal annotations. It can mislabel valid alternatives because evaluator action checks use exact tool names and arguments. We group model-selection folds by task, so prefixes from one task never enter both fitting and validation folds. The retail training set contains 1,123 candidate steps from 74 tasks; 20.7% are positive.

4.2 Prefix-Only Credit Model

The primary model is L2-regularized logistic regression. Features encode action type, mutation status, prior tool errors, repeated calls, explicit confirmation, schema validity, argument grounding, policy-token overlap, and prefix position. Five-fold grouped OOF predictions select the regularization coefficient and threshold. The operating rule requires at least 40% OOF precision and at most 25% selection, then maximizes F1. The selected retail model uses L2 coefficient 0.01 and $\eta = 0.658$. Features, weights, and threshold remain frozen during final evaluation.

4.3 Baseline-Preserving Intervention

CA-BPI scores each proposed action. Scores below η preserve it. Selected actions enter a verifier that receives the policy, tool schemas, conversation prefix, and proposal. The verifier blocks only a concrete policy, schema, parameter, or confirmation conflict. Otherwise, it preserves the proposal. A block returns structured feedback to the same agent. The controller has no oracle replacement generator.

4.4 Same-Prefix Candidate Preferences

We also test a train-only route from localization to action selection. At each airline training prefix, structured prompts generate a baseline candidate, a conservative candidate, an alternative legal action, and a corrective candidate. A process verifier assigns one of three frozen safety labels: *correct*, *correctable*, or *irrecoverable*. Equal-safety candidates use workflow progress and argument grounding as tie-breakers. Unresolved sets abstain.

A Bradley–Terry model (Bradley and Terry, 1952) ranks candidates from prefix-available structural features. Task-grouped OOF evaluation prevents prefixes from the same task crossing folds. This study is secondary: it tests whether train-only supervision can rank alternatives, not whether the main credit classifier improves reward.

5 Experimental Setup

Benchmark and models. We use the retail, airline, and telecom domains in the official τ^2 -bench stack. It provides policies, user simulators, database tools, and final-state evaluators. Qwen Flash (dashscope/qwen-flash) serves as agent, user simulator, and language evaluator. Main CA-BPI runs use agent temperature zero. Each final task receives three trials because the hosted endpoint can vary at that setting. The exploratory ranking comparison uses temperature 0.7 for all methods.

Splits. Retail supplies 74 training tasks, a separate test-pilot subset, and 20 final tasks excluded from that pilot. The pilot informed controller design and diagnostic reporting; model fitting and threshold selection used only task-grouped OOF predictions on the training split. Airline and telecom supply 30 and 44 domain-specific training tasks. A 25% test pilot informed model-family and controller choices, and the remaining 15 and 30 tasks were used for domain extensions. However, v1 models were evaluated on those extension task IDs before the reported canonical v2 runs. The v2 models were still fit without test references, but this adaptive test reuse makes the airline and telecom results diagnostic rather than confirmatory. No online component receives an evaluator reference. The candidate-ranking study uses 20 airline training tasks. Its exploratory comparison uses 10 separate tasks and three trials.

The strict replay audit uses 20 unused telecom tasks and three trials. It caches each baseline candidate and user response. The method run replays both streams until an intervention changes the history. A zero-intervention trajectory must match its baseline message by message and reuse the baseline evaluation record.

Baselines and metrics. The baseline is the official LLM agent with the same model, policy, tools, simulator, and evaluator. Binary task reward is the primary endpoint. DB, natural-language (NL)

score, evaluator action-match rate, invalid tool-result rate, tool calls, tokens, cost, and duration are secondary or diagnostic endpoints. The task is the independent unit. We first average the three matched trial differences within each task, report task-level wins/ties/losses, apply a two-sided exact sign test to non-ties, and bootstrap tasks for 10,000 resamples. Trial-level counts are descriptive. We make no multiplicity-adjusted significance claims for secondary metrics.

6 Results

6.1 Finding 1: Held-Out Agreement Is Moderate

Table 1 reports agreement with the deterministic silver-label rule in all three domains. The frozen retail model obtains 61.1% F1 on held-out trajectories. Its 30.3% selection rate exceeds the 25% OOF constraint. We did not retune the threshold. Retail F1 rises by 14.7 points from OOF to held-out evaluation. The held-out positive rate is 23.7%, versus 20.7% in training, and the held-out tasks may also be easier. We therefore treat this increase as a task-set shift, not evidence that generalization improves after model selection.

Structural-only logistic regression reaches 46.5% OOF F1; grounding-only features reach 45.4%. Frozen semantic features, pairwise credit ranking, and nonlinear models reach 32.8%, 31.7%, and 32.7% selected F1. We retain the auditable linear model.

6.2 Finding 2: Reward Differences Are Inconclusive

Table 2 compares retail runs. Mean reward increases by 1.7 percentage points, while NL decreases by 1.7 points. At task level, the method records four wins, twelve ties, and four losses. The exact sign test gives $p = 1.0$; the 95% task-clustered interval is $[-11.7, 16.7]$ points. The experiment does not establish a reward improvement.

The controller averages 5.55 gate calls and 1.70 verifier calls per trajectory. Token use rises by 57.1%, and mean duration rises by 23.1 seconds. Task-clustered intervals for both differences exclude zero. The system-cost result is more precise than the reward result.

6.3 Finding 3: Localization Rarely Produces Treatment

Seven of 60 retail method trajectories contain an intervention; nine actions change. Each treated trajectory has the same binary reward as its task-matched baseline. Three treated trajectories change an action at a silver-labeled step. All seven trial-level wins and all six losses occur on untreated trajectories. The observed reward differences therefore cannot be attributed to the controller.

Table 3 gives the cross-domain reward results. Every interval includes zero. Airline reaches 50.5% silver-label F1 but changes no trajectory. Telecom reaches 15.4% F1 and also changes none. Because these two method runs execute the baseline policy throughout, their reward deltas are empirical run-to-run noise under independent model and simulator sampling, not method effects. Their common +2.2-point estimate and wide intervals place retail’s +1.7-point delta within the observed noise floor. The telecom duration reduction is likewise a correlation between independent rollouts.

Table 4 locates the break in the intervention pipeline. Retail selection is broad, but only seven trajectories receive treatment. Airline invokes the verifier 41 times without changing an action. Telecom selects only 13 of 206 scored decisions and also makes no change.

The funnel separates three mechanisms. Telecom exhibits threshold shift: its scores rarely cross the frozen threshold. Airline exhibits verifier preservation: selected proposals remain unchanged. Retail reaches treatment, but the changed actions do not alter binary reward. Localization quality alone does not identify which replacement improves return.

6.4 Finding 4: Replay Validates Fidelity, Not Repair

The telecom replay run reproduces all 60 baseline trajectories exactly. It replays 848 agent candidates and 204 user responses without a semantic mismatch. The method selects no action. Among 129 recorded gate decisions, the highest score is 0.314, below the frozen threshold of 0.385. Verifier calls, post-divergence model calls, and interventions are all zero. Trial W/T/L is 0/60/0.

The audit exposes two failures. First, the telecom score distribution lies below the training threshold. The training report already marks the

Domain	Split	Prec.	Rec.	F1	Select
Retail	Train OOF	42.8	50.6	46.4	24.6
Retail	Held-out	54.5	69.6	61.1	30.3
Airline	Train OOF	35.9	57.5	44.2	14.5
Airline	Reused test	61.0	43.1	50.5	16.9
Telecom	Train OOF	36.8	70.0	48.3	3.0
Telecom	Reused test	53.8	9.0	15.4	6.3

Table 1: Step-level silver-label agreement (percent). Retail held-out evaluation contains 333 scored decisions: 55 true positives, 46 false positives, and 24 false negatives. Airline and telecom canonical models were selected after earlier runs on the same test task IDs; their test rows are diagnostic.

Metric	Base	CA-BPI	Δ
Reward (%)	40.0	41.7	+1.7
DB score (%)	40.0	41.7	+1.7
NL (%)	88.3	86.7	-1.7
Action match (%)	58.8	63.1	+4.3
Invalid tool result (%)	8.3	9.7	+1.4
Tool calls	7.00	7.27	+0.27
Tokens	84,518	132,792	+48,274
Cost (USD)	.00459	.00749	+.00290
Duration (s)	23.7	46.8	+23.1

Table 2: Retail final results over 20 tasks and three trials. DB score equals binary reward whenever the evaluator returns a DB breakdown; missing DB breakdowns from prematurely terminated runs are scored zero. Lower is better for invalid tool-result rate and system costs.

required precision constraint as unsatisfied. Second, every baseline reward is zero: 49 trajectories terminate after too many tool errors, eight stop through the user, and three reach the step limit. The replay mechanism is faithful, but this run contains no treated pair and cannot estimate repair utility.

6.5 Mechanism Probe: Candidate Ranking

Structured generation produces 222 airline prefix sets. Of these, 104 contain no independent alternative. Among the remaining 118 eligible sets, 94 yield a preference: 79.7% eligible coverage and 42.3% gross coverage. The hybrid labels produce 293 pairs: 252 from frozen safety labels, 31 from progress, and 10 from grounding. The ranking model obtains 66.9% grouped OOF accuracy. Fold accuracy ranges from 60.3% to 72.6%.

The probe also quantifies alternative scarcity: 46.8% of prefix sets have no independent alternative. Exploratory deployment raises mean token use from 97,943 to as much as 802,440 and duration from 36.1 to 246.1 seconds. Conservative ranking replaces fewer candidates than aggressive ranking but generates more candidate sets (385

versus 327) and candidate attempts (3,276 versus 2,711) because its independent rollouts contain more decision points. Replacement rate therefore does not determine generation cost. These task-matched rollouts do not share candidates or simulator responses. Appendix C reports their outcomes as a prototype for future repair policies and evidence about alternative scarcity, not as causal evidence.

7 Discussion

Responsibility and action value differ. A responsibility label identifies a decision associated with failure. A repair policy must choose an available action that improves future return. This choice depends on valid parameters, workflow state, recovery feasibility, and later user behavior. The credit classifier models none of these quantities.

Treatment coverage is part of the method. Localization F1 has no operational effect when the controller preserves every selected action or the gate selects none. Selective controllers therefore require gate coverage, verifier acceptance, executed changes, and treated outcomes. Aggregate reward alone obscures this mechanism. In the present data, only seven trajectories reach treatment, which is insufficient to estimate a general localization-to-repair effect.

Candidate ranking offers a costly path forward. Same-prefix preference learning discriminates alternatives under grouped validation, but almost half of the sampled prefixes lack an independent alternative. Its cost is high, and independent rollouts prevent causal attribution. A stronger test would cache candidate 0, generate alternatives only at train-calibrated high-margin prefixes, and replay the user until the first replacement.

Domain	N	Base	CA-BPI	Δ	95% CI	Task W/T/L
Retail	20×3	40.0	41.7	+1.7	$[-11.7, 16.7]$	4/12/4
Airline	15×3	28.9	31.1	+2.2	$[-11.1, 15.6]$	3/9/3
Telecom	30×3	5.6	7.8	+2.2	$[-5.6, 10.0]$	4/24/2

Table 3: Observed binary reward by domain (percent). N is tasks $\times$ trials. Task-matched rollouts do not share candidate 0 or simulator responses. Airline and telecom execute no intervention, so those rows estimate rollout noise rather than treatment effects. Their canonical v2 models also follow earlier v1 evaluations on the same task IDs.

Pipeline stage	Retail	Airline	Telecom
Gate-selected decisions / scored decisions	101/333	41/242	13/206
Selected trajectories / method trajectories	47/60	27/45	7/90
Verifier calls	102	41	13
Changed actions	9	0	0
Treated trajectories / method trajectories	7/60	0/45	0/90
Reward-changed treated trajectories	0/7	–	–

Table 4: Failure decomposition for the primary runs. A selected trajectory contains at least one decision above the frozen gate threshold. Retail has 101 action-safety verifier calls for selected decisions plus one separate authorization check, giving 102 total verifier calls. A treated trajectory contains an executed action change. Dashes indicate that no treated trajectory exists.

Relation to RL. Our model estimates retrospective responsibility. It neither estimates return nor updates policy parameters, so it is an offline diagnostic rather than an RL algorithm. Responsibility labels could support preference learning or reward modeling, but that extension would require separate train-only selection and a frozen policy evaluation.

8 Conclusion

The retail credit model reproduces silver responsibility labels on held-out trajectories, but the gated verifier yields no supported reward gain. The intervention funnel identifies different failures across domains: threshold shift in telecom, verifier preservation in airline, and sparse, reward-neutral treatment in retail. Exact telecom replay validates the replay mechanism but produces no treatment. These runs establish the value of leakage-controlled funnel reporting; they do not establish a general localization–repair gap or a repair effect. Repair evaluation requires calibrated treatment coverage, grounded alternatives, and replay-controlled treated pairs.

Limitations

Construct validity. Silver labels inherit errors from exact action matching and evaluator logic. No independent human annotation validates them. Localization scores therefore measure agreement

with the implemented rule, not causal responsibility.

Internal validity. Primary cross-domain comparisons use task-matched but independent simulator rollouts. Exact replay is limited to telecom, where no intervention occurs and every baseline reward is zero. Neither design estimates a treatment effect in the present data. Airline and telecom canonical models were chosen after earlier v1 evaluations on the same test task IDs. Although no test reference entered fitting or online inference, this adaptive reuse prevents treating those extensions as untouched confirmatory tests.

External validity. The study uses one model family, one unversioned hosted endpoint, and three synthetic customer-service domains. Provider updates may affect reproducibility even at temperature zero. Results need not transfer to other models, interfaces, or domains.

Statistical conclusion validity. Final task sets are small, so reward intervals include meaningful gains and losses. The ranking study uses 20 training tasks and 10 exploratory final tasks. Its deployment comparison requires a larger replay-controlled evaluation. These limits support claims about localization and evaluation design, not repair effectiveness.

Ethical Considerations

The experiments use synthetic customer-service tasks and benchmark databases. They contain no real customer records. A controller that blocks state mutations can prevent unsafe writes, but it can also deny legitimate requests. Deployment would require domain-specific calibration, monitored abstention, and human escalation. We report null results, failed deployment gates, and inference cost to avoid presenting diagnostic accuracy as operational benefit.

References

- Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. 2025. [\$\tau^2\$ -bench: Evaluating conversational agents in a dual-control environment](#). *arXiv preprint arXiv:2506.07982*.
- Ralph Allan Bradley and Milton E. Terry. 1952. [Rank analysis of incomplete block designs: I. the method of paired comparisons](#). *Biometrika*, 39(3/4):324–345.
- Hyungjoo Chae, Seonghwan Kim, Junhee Cho, Seungone Kim, Seungjun Moon, Gyeom Hwangbo, Dongha Lim, Minjin Kim, Yeonjun Hwang, Minju Gwak, Dongwook Choi, Minseok Kang, Gwanhoon Im, ByeongUng Cho, Hyojun Kim, Jun Han, Taeyoon Kwon, Minju Kim, Beong-woo Kwak, and 2 others. 2025. [Web-shepherd: Advancing PRMs for reinforcing web agents](#). In *Advances in Neural Information Processing Systems*, volume 38.
- Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. [Let’s verify step by step](#). *arXiv preprint arXiv:2305.20050*.
- Xufang Luo, Yuge Zhang, Zhiyuan He, Zilong Wang, Siyun Zhao, Dongsheng Li, Luna K. Qiu, and Yuqing Yang. 2025. [Agent lightning: Train ANY AI agents with reinforcement learning](#). *arXiv preprint arXiv:2508.03680*.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. [Self-refine: Iterative refinement with self-feedback](#). In *Advances in Neural Information Processing Systems*, volume 36.
- Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. [Toolformer: Language models can teach themselves to use tools](#). *Advances in Neural Information Processing Systems*, 36.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. [Reflexion: Language agents with verbal reinforcement learning](#). In *Advances in Neural Information Processing Systems*, volume 36.

Richard S. Sutton and Andrew G. Barto. 2018. [Reinforcement Learning: An Introduction](#), 2nd edition. MIT Press.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. 2025. [\$\tau\$ -bench: A benchmark for tool-agent-user interaction in real-world domains](#). In *International Conference on Learning Representations*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. [ReAct: Synergizing reasoning and acting in language models](#). In *International Conference on Learning Representations*.

A Reproducibility Details

The final retail model uses five grouped folds, L2 coefficient 0.01, threshold 0.658338, and artifact prefix 53b1ac312be45f55. Retail, airline, and telecom runs use 20, 15, and 30 tasks with three trials and seed 300. Task IDs reside in the run configurations. Airline and telecom v1 models were evaluated on the same extension IDs before the canonical v2 models; the reported v2 models remained train-only but were selected after observing earlier aggregate test behavior. The strict telecom audit uses a disjoint 20-task manifest, exact candidate-0 and user replay, and the frozen model `credit_telecom_ca_telecom_v2_canonical`. Experiment configurations store hashes for the baseline results, task manifest, and credit model.

B Responsibility Ablations

Model	Prec.	Rec.	F1
Primary logistic	42.8	50.6	46.4
Structural only	42.7	51.1	46.5
Grounding only	41.7	49.8	45.4
Semantic pointwise	23.7	53.7	32.8
Pairwise credit ranking	–	–	31.7
Nonlinear OOF	–	–	32.7

Table 5: Task-grouped retail OOF ablations (percent). Dashes mark runs whose retained report contains only selected F1.

C Exploratory Ranking Deployment

Table 6 reports independent airline rollouts for the ranking prototype. Conservative ranking re-

places 3.1% of candidates and reaches 26.7% reward, compared with 10.0% for the baseline. The task-clustered interval is positive, but the exact sign test over four non-tied tasks gives $p = .125$. This large relative change under sparse replacement is compatible with rollout noise: the methods do not share candidate 0 or simulator responses. The comparison is descriptive and does not demonstrate a ranking effect.

Method	Reward	Replace	Tokens	Time
Baseline	10.0	0.0	97,943	36.1
Aggressive	13.3	29.6	642,114	226.0
Conservative	26.7	3.1	802,440	246.1

Table 6: Exploratory, non-causal airline ranking on 10 tasks with three trials. Reward and replacement are percentages; time is seconds. Conservative rollouts contain 385 candidate sets versus 327 for aggressive ranking, explaining their higher cost despite fewer replacements.

D Verifier Repair Pilot

A separate pilot on 20 training tasks made 30 repair-verifier calls. It accepted no candidate and executed no repaired action. We exclude this training-set result from final task comparisons. It provides another diagnostic of the replacement bottleneck.