When Agents Choose What to Observe: BEATS – Answer-Free Test-Time Selection for Verifiable-Reward Agents

Anonymous authors
Paper under double-blind review

Abstract

Verifiable-reward methods typically evaluate an output against a fixed problem instance or an externally supplied evidence record. In partially observable agent environments, however, the policy may also control which information is acquired. Verification is therefore conditional on a policy-constructed record: a certificate can be valid relative to that record while its decision is wrong under the complete record. We study this failure surface in LegalEvidenceGym, a synthetic executable testbed that separates latent facts, available sources, the agent's acquired information state, record-entailed decisions, and structured certificates. Across six registered cost-penalized RL runs, optimized reward, certificate validity, and joint decision–certificate success increase, while decisive-evidence recall (DER) and joint correctness on controlled Evidence-Twin pairs (ETC) decrease. We call this longitudinal signature *Evidence Collapse*; it is consistent with shortcut learning under the optimized objective but does not identify a unique cause or a verifier error. We introduce Best-of-N Evidence-Aware Trajectory Search (BEATS), an answer-free, training-free test-time procedure. BEATS diversifies candidate information states with bounded public prefixes, ranks them using only replayable public features, and serializes a certificate entailed by the selected record. On 150 held-out episodes (75 Evidence-Twin pairs) evaluated under six frozen Qwen checkpoint conditions, BEATS@4 achieves 92.33% joint success, compared with 86.89% for the strongest same-pool, same-recovery baseline, deepest-prefix Max-Probe@4 (+5.44 points; pair-clustered 95% CI [2.44, 9.00]), while reducing mean environment investigation cost by 23.3%. These results show that terminal verification should account for how an agent constructs its information state, not only whether its final output is internally valid.

1 Introduction

Language-model agents increasingly search, retrieve, and call tools before answering Yao et al. (2023b); Liu et al. (2024). Verifiers can make their final outputs machine-checkable, an approach that underlies outcome supervision and reinforcement learning from verifiable rewards Shao et al. (2024). In many standard RLVR settings, the problem instance and the information against which an output is verified are fixed independently of the policy. For an interactive agent, however, the policy may also determine which information enters that record by choosing what to inspect, retrieve, or ignore. In such interactive, partially observable environments, the verifier typically sees only the information that the policy chose to acquire. This couples two distinct problems: *constructing an adequate information state* and *producing an output justified by that state*.

The distinction creates a verification blind spot. A sound checker can establish that a structured certificate follows from the agent's acquired record, but not that the record contains all decision-changing information. For example, an agent may inspect several sources supporting a termination decision but stop before acquiring an accessible source that defeats a required element. Its certificate can then be perfectly valid relative to the acquired ledger, while the decision is wrong relative to the complete admissible record. A policy controlling both investigation and submission may stop early and produce an internally valid output from an insufficient record. The checker has not failed; its guarantee is simply conditional on a policy-controlled information state. When the training objec-

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.

tive also rewards low interaction cost, this boundary creates a shortcut: improving terminal reward need not improve the information-gathering process. This connects record construction to broader concerns about reward misspecification and goal misgeneralization Amodei et al. (2016); Langosco et al. (2022), while remaining distinct from verifier false positives.

To study this boundary, we need tasks in which an outcome depends not only on what evidence exists, but also on whether that evidence is accessible, admissible, reliable, and sufficient to meet an explicit burden. Structured procedural decision tasks provide this control: they make the relation between evidence, decision rules, and final justification executable rather than implicit. We instantiate this setting with LegalEvidenceGym, a synthetic testbed based on five Chinese employment-termination rule templates. The legal setting is an experimental substrate, not the paper's application target: it lets us represent burdens, thresholds, source reliability, admissibility, and provenance explicitly, rather than making claims about operative law. Throughout, an *evidence ledger* is the agent's acquired information state, and a *certificate* is its structured final justification; neither term denotes formal theorem proving. Controlled *Evidence-Twins* alter one source condition while holding latent case facts and the executable rule program fixed, allowing us to test whether a policy responds to decision-relevant changes rather than repeating a documentable label.

Our experiments reveal two complementary gaps. First, on 600 static Evidence-Record prompts, GPT-4o attains 92.83% decision accuracy but only 84.17% joint decision–certificate success; 52 outputs reach the correct decision with an invalid certificate (Supplementary Sec. S5). Second, across six registered cost-penalized GRPO runs, optimized reward and terminal validity improve while decisive-evidence recall and joint Evidence-Twin correctness decline relative to initialization. We call this observed longitudinal signature *Evidence Collapse*. It is consistent with shortcut learning under the optimized objective, with reward gaming as one possible explanation, but the experiments do not identify a unique causal mechanism: cost incentives, exploration, and initial policy bias may all contribute.

To mitigate this failure at inference time, we introduce Best-of-N Evidence-Aware Trajectory Search (BEATS; Figure 1). BEATS is an answer-free, training-free form of test-time compute Snell et al. (2024). It exposes one frozen policy to four bounded public investigation prefixes, deterministically replays the resulting trajectories, and selects using only observable state features. A lexicographic key prefers a recoverable, sufficiently resolved, low-cost record; ledger-bounded canonicalization then submits only what the selected record entails. The selector never accesses the target decision, evaluator reward, decisive-source annotations, latent facts, or twin identity.

Our contributions are:

1. **A controlled testbed for policy-constructed records.** LegalEvidenceGym separates latent truth, source availability, acquisition, record-relative entailment, decisions, and certificate validity, enabling controlled interventions on the agent's information-gathering process.
2. **An empirical process-level diagnostic.** Evidence Collapse identifies terminal-objective improvement accompanied by loss of decisive evidence or Evidence-Twin correctness. We measure within-run change across six registered RL runs without attributing it to verifier error or a unique causal mechanism.
3. **An answer-free test-time selection method.** On identical candidate pools with uniform recovery, BEATS@4 achieves 92.33% PVS, outperforming the strongest same-pool baseline, deepest-prefix Max-Probe@4, by +5.44 points (pair-clustered 95% CI [2.44, 9.00]) while reducing environment investigation cost by 23.3% and improving twin consistency.

2 Related Work

Legal reasoning and structured argument. LexGLUE, LegalBench, LawBench, and CUAD cover legal classification, reasoning, and document analysis Chalkidis et al. (2022); Guha et al. (2023); Fei et al. (2024); Hendrycks et al. (2021). LAR-ECHR evaluates continuation of legal argument chains Chlapanis et al. (2024), while tree-structured legal reasoning explicitly links facta probanda, evidence, and conclusions Shen et al. (2025). These benchmarks make reasoning structure visible, but the usual evaluated object is still a supplied case or generated argument. LegalEvi-

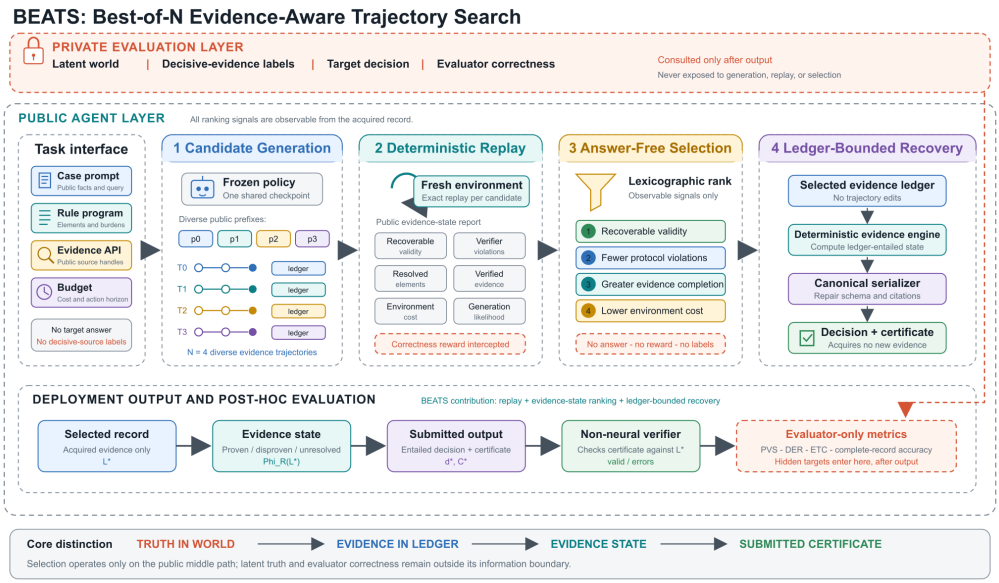


Figure 1: The verification boundary and BEATS pipeline. Candidate generation, replay, selection, and recovery use only public rules, actions, and acquired records; evaluator-only fields enter after selection and output are frozen.

denceGym instead evaluates how an agent constructs the evidence ledger on which an institutional decision rests. Its executable rule programs are closer to Catala Merigoux et al. (2021), but are synthetic experimental artifacts rather than statements of operative law.

Evidence sufficiency and verification-aware agents. Verification-aware planning can attach passing criteria to intermediate agent subtasks Xu et al. (2026), and VeRA generates benchmarks from executable specifications with deterministic labels Cheng et al. (2026b). Learned verifiers and verifiable process rewards likewise improve structured reasoning Cobbe et al. (2021); Pronesti et al. (2026). Our boundary is orthogonal: even a sound checker cannot determine whether a policy-controlled record omitted a decision-changing source. Evidence-Twins extend contrast sets Gardner et al. (2020) from output sensitivity to evidence-gathering behavior.

Recent work is closer at the evidence-control level. MAP-Law uses legal-element coverage and marginal gain to control retrieval and stopping Cheng et al. (2026a); GLEAN accumulates guideline-grounded evidence and triggers active verification Zhang et al. (2026); SeekBench tests whether search agents recognize evidentiary sufficiency Shao et al. (2025); and S2G-RAG predicts sufficiency and explicit retrieval gaps Li et al. (2026). BEATS differs by selecting among immutable same-policy candidate ledgers with an exact public evidence engine, while complete-record correctness remains hidden. This narrower setting permits deterministic measurement of both ledger validity and omitted decisive evidence.

Test-time selection and process-aware RL. Self-consistency, Tree of Thoughts, and inference scaling select among sampled solutions Wang et al. (2023); Yao et al. (2023a); Snell et al. (2024). BEATS instead ranks replayed evidence states without an answer key or learned critic. SPRO and iStar introduce process or implicit step rewards for reasoning and agentic RL Fei et al. (2025); Liu et al. (2026); IPS-GRPO analyzes a distinct form of outcome-level mode collapse Sinha et al. (2026). These approaches motivate process-sensitive learning, but they neither define our ledger-sufficiency diagnostic nor remove its main confound: our registered objectives explicitly penalize environment cost. Reward hacking remains a causal account of proxy exploitation Amodei et al. (2016); Langosco et al. (2022); Helff et al. (2026); Evidence Collapse is only the observable co-movement of optimized reward with declining DER or joint Evidence-Twin correctness.

3 METHODS

3.1 ACTIVE LEGAL EVIDENCE CONSTRUCTION

Let R be a public experimental rule program and W a private latent world. W determines evidence sources $E(W)$ with accessibility, authenticity, reliability, admissibility, relevance, provenance, and acquisition cost. After actions up to time t, the agent holds a ledger $L_t \subseteq E(W)$. A deterministic engine maps the ledger to legal-element evidence states

$$\Phi_R(L_t) = (s_1, \ldots, s_m). \tag{1}$$

Each s_k is PROVEN, DISPROVEN, or UNRESOLVED. The record-bounded institutional decision is $d_t = f_R(\Phi_R(L_t))$. A private evaluator separately computes $d^\star$ from the complete admissible record.

A certificate C contains a decision, a claim and citations for every element, burdens, unresolved elements, and the applied rule. The verifier $V_R(C, L_t)$ rejects unacquired citations, failed authenticity, inadmissible support, burden or threshold errors, inconsistent unresolved-element lists, and decisions not entailed by L_t. Canonical serialization can make a valid C_L for any ledger L, yet $f_R(\Phi_R(L)) \neq d^\star$ remains possible when L omits decisive evidence. Thus

$$V_R(C_L, L) = \text{valid} \;\not\Rightarrow\; f_R(\Phi_R(L)) = d^\star. \tag{2}$$

This is the structural reason terminal certificate reward cannot alone ensure adequate investigation.

For replay report r, observable evidence completion is

$$p_{\text{ev}}(r) = \frac{\#\{k : s_k \neq \text{UNRESOLVED}\}}{m} \in [0, 1]. \tag{3}$$

This is a selector feature, not the target metric: a burden-based decision can be entailed while some elements remain unresolved. Complete-record correctness is evaluated only after the output is frozen.

3.2 LEGALEVIDENCEGYM AND METRICS

LegalEvidenceGym is a synthetic, executable environment for active evidence adjudication. Its initial instantiation uses five stylized employment-termination templates—standard termination, redundancy, probation, misconduct, and mutual termination—because they make competing evidence, burdens of proof, admissibility, provenance, and ordered decisions explicit. These templates are experimental rule programs rather than models of operative law.

In each episode, an agent sequentially acquires and inspects evidence sources, updates element-level claims, and eventually submits a decision and structured certificate. Each rule program specifies two or three substantive elements, their dependencies, the evidence that supports or attacks them, party-specific burdens, decision thresholds, and decision order. The public observation excludes latent facts, the complete-record decision, reference certificate, and decisive-source annotations. We provide the complete action semantics, rule programs, and source-generation details in Supplementary Sec. S2.

To test whether an agent responds to decision-relevant information rather than merely producing a plausible label, we construct controlled *Evidence-Twin* pairs. The two episodes in a pair share the same latent case structure and rule program but differ in one source condition—availability, reliability, admissibility, or an irrelevant hidden field. A successful policy must change its decision when, and only when, the intervention changes the complete-record decision.

The interactive benchmark contains 1,500 deterministic episodes: 1,200 for training, 150 for validation (75 Evidence-Twin pairs), and 150 held-out test episodes (75 pairs), balanced across the five templates and three cost levels. Training-only oracle traces provide 9,594 supervised action examples.

Decision Accuracy (DA) compares the output with $d^\star$; Certificate Validity (CV) checks $V_R(C, L)$; and the primary endpoint, Prediction-and-Verification Success (PVS), is

$$\text{PVS} = \frac{1}{n} \sum_i \mathbf{1}[\hat{d}_i = d_i^\star \wedge V_R(C_i, L_i) = \text{valid}]. \tag{4}$$

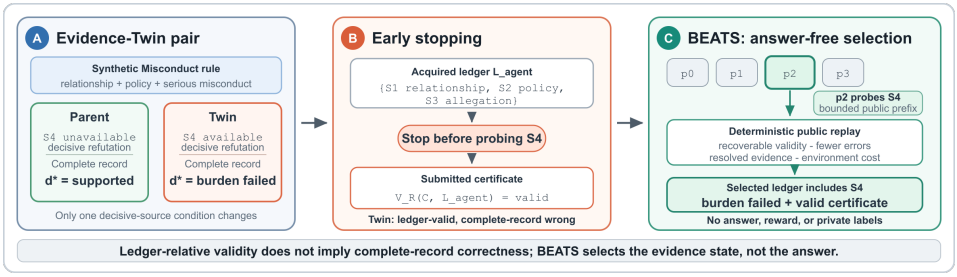


Figure 2: Schematic Evidence-Twin example in the synthetic Misconduct program. The parent and twin differ only in the availability of a decisive refutation. An agent that stops before probing this source can produce a certificate that is valid relative to its acquired ledger but wrong under the twin's complete record. BEATS generates bounded-prefix candidate ledgers and selects a better-resolved state using only public replay features; it never observes the target answer or decisive-source labels. This is an illustrative schematic rather than a logged episode.

Decisive Evidence Recall (DER) is the fraction of evaluator-private decisive sources acquired. Evidence-Twin Consistency (ETC) requires both members of a pair to receive their correct decisions; repeating one wrong label gets no credit. Environment investigation cost is reported separately from model inference.

We use *Evidence Collapse* as shorthand for the registered longitudinal pattern in which a terminal objective improves while DER or ETC declines across the registered runs. This signature is consistent with goal misgeneralization or reward gaming in agentic tool use Langosco et al. (2022); Helff et al. (2026). Because both registered objectives contain a direct cost penalty, however, it establishes an evidence–cost tradeoff under terminal reward rather than a causal demonstration of reward hacking.

Illustrative failure mode. Figure 2 concretizes the information boundary studied in this paper: a certificate can be valid relative to the acquired ledger while its decision is wrong under the complete record. BEATS instead selects among replayed public evidence states without observing the hidden target decision.

3.3 Best-of-N Evidence-Aware Trajectory Search

Diversify evidence states. BEATS generates $N = 4$ trajectories from one frozen checkpoint. Candidate $j \in \{0, 1, 2, 3\}$ receives the first j public probe bundles, each acquiring a public source and checking authenticity and admissibility. Within each episode, public handles are ordered deterministically by acquisition cost and then identifier; the order is not a learned or evaluator-dependent ranking. The model completes each prefixed trajectory at temperature 0.7. Prefixes vary the acquired information state, not merely final-answer wording.

Replay without answers. Every candidate is replayed in a fresh deterministic environment. DECIDE is intercepted before correctness reward. The report includes recoverable-validity indicator v_{rec}; counts of certificate, environment-action, and parse errors $(e_{\text{cert}}, e_{\text{act}}, e_{\text{parse}})$; evidence completion p_{ev}; environment cost c; verified-admissible evidence count n_{verified}; mean log likelihood ℓ; and candidate identifier id. It excludes correctness, PVS, DER, twin target, latent facts, and decisive-source membership.

Select sufficient evidence. For report r, the frozen lexicographic key is

$$K(r) = (v_{\text{rec}}, -e_{\text{cert}}, -e_{\text{act}}, -e_{\text{parse}}, p_{\text{ev}}, -c, n_{\text{verified}}, \ell, \text{id}). \tag{5}$$

It first prefers a recoverable terminal state; then fewer protocol failures, greater observable evidence completion, and lower cost at equal completion. Evidence count and likelihood are late tie-breakers. Putting evidence count first would favor accumulation irrespective of evidence completion; putting cost first would reward empty but internally consistent ledgers.

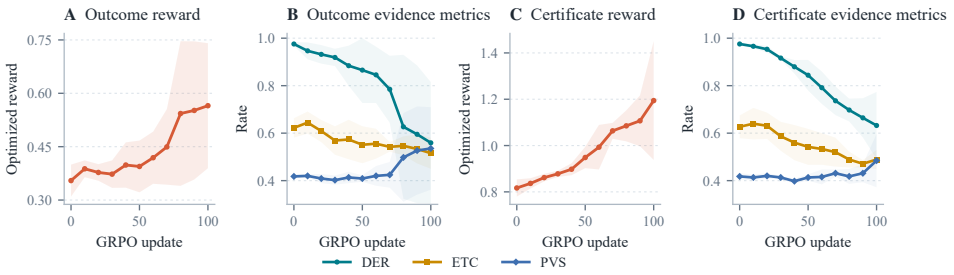


Figure 3: Registered training dynamics under cost-penalized terminal rewards. Optimized reward and PVS rise while decisive-evidence recall (DER) and joint Evidence-Twin correctness (ETC) fall. Curves show means over the three runs, and shaded regions denote ± 1 standard deviation. Reward scales are objective-specific. Because investigation cost is directly penalized, this figure documents co-movement rather than identifying a unique causal mechanism.

The registered $N = 4$ instance is a bounded depth scan, not open-ended planning. It assumes that the first three deterministic bundles induce useful state diversity. The nested $N \in \{1, 2, 4\}$ evaluation tests sensitivity to this coverage but does not guarantee that a decisive source always occurs within the first three bundles. Frozen evaluation uses temperature 0.7, at most 768 new tokens per action, a 12-unit environment budget, and a 20-action horizon; Supplementary Secs. S3–S4 provide the complete configuration.

Recover only from the selected ledger. If the raw certificate is invalid, the deterministic engine recomputes $\Phi_R(L)$ and serializes its canonical certificate. Recovery acquires no evidence, edits no trajectory, and reads no target answer. BEATS is therefore registered as one target-answer-free test-time procedure: structured state diversity, evidence-state selection, and ledger-bounded serialization. We will release candidate-level replay traces needed to audit this information boundary.

4 Experiments

4.1 Protocols and Baselines

The primary interactive model is Qwen2.5-7B-Instruct Yang et al. (2024). Three one-epoch LoRA SFT runs (distinct random seeds) Hu et al. (2022) initialize Outcome-GRPO and Certificate-GRPO Shao et al. (2024). Each RL run has 100 updates, four rollouts per group, a 20-step horizon, and a 30-unit training budget. The registered objectives are

$$R_{\text{outcome}} = \mathbf{1}[\hat{d} = d^\star] - 0.05c - p_{\text{invalid}}, \tag{6}$$

$$R_{\text{certificate}} = R_{\text{outcome}} + \mathbf{1}[V_R(C, L) = \text{valid}], \tag{7}$$

where c is environment cost and p_{invalid} aggregates invalid-action penalties. Neither objective observes DER, ETC, evidence completion, or decisive-source identity. Training dynamics use a 30-unit budget, whereas all BEATS comparisons use the separately frozen 12-unit budget; environment costs are not compared across these protocols.

BEATS is frozen on validation. The IID test evaluates three SFT and three update-100 Outcome-GRPO checkpoints on the same 150 unseen episodes: 900 correlated model-run-episode observations. Confidence intervals use 10,000 paired bootstrap resamples over 75 parent/twin identifiers; each sampled cluster contains both twins under all six checkpoint conditions. All Best-of-4 selectors reuse the same sampled pool. Baselines are Random, Self-Consistency, Certificate-only, BEATS without recovery, and deepest-prefix Max-Probe@4 (which deterministically selects the candidate generated from the longest exploration prefix) with *the same canonical recovery* as BEATS. Additional controls give the other selectors identical recovery; an evaluator-only pool oracle measures headroom. Exact hyperparameters, preregistration locks, and all per-run tables are in Supplementary Secs. S3–S7. Before held-out evaluation, the procedure and analysis were frozen after BEATS exceeded Max-Probe by 3.56 PVS points across the three Outcome-GRPO validation checkpoints (pair-clustered 95% CI [0.44, 7.78]).

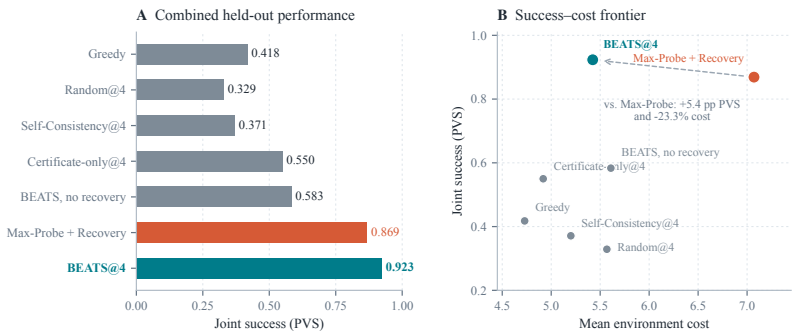


Figure 4: Held-out performance over six frozen Qwen checkpoints (150 unique episodes, 75 pair clusters, and 900 correlated checkpoint–episode observations). BEATS@4 improves PVS by 5.44 points over same-pool, same-recovery deepest-prefix Max-Probe@4 while using 23.3% less mean environment investigation cost.

Metric	Outcome-GRPO	Cert.-GRPO
ΔDA	-0.078	-0.113
ΔCV	$+0.280$	$+0.218$
ΔPVS	$+0.118$	$+0.067$
ΔDER	-0.416	-0.343
ΔETC	-0.107	-0.138
ΔCost	-5.498	-5.148

Table 1: Mean change from update 0 to 100 on registered validation checkpoints. Terminal validity improves while decision accuracy and evidence-gathering metrics decline.

4.2 Static Decision–Certificate Gap

Before interactive training, we test whether a strong model distinguishes a plausible decision from a record-supported certificate. One Azure deployment labeled GPT-4o produces structured outputs with 100% parse coverage on 300 Complete-Facts and 600 Evidence-Record prompts; the exact provider snapshot was not recorded. On Evidence-Record prompts, decision accuracy is 0.928 (557/600), whereas PVS is 0.842 (505/600). All 52 correct-decision/invalid-certificate outputs overclaim at least one engine-UNRESOLVED element as PROVEN. Conversely, 27 outputs cite sufficient evidence subsets that differ from the canonical reference and pass verification, showing that the checker evaluates semantic support rather than string identity. This single-deployment result establishes a decision–certificate gap within the testbed, not its prevalence across models (Supplementary Sec. S5).

4.3 Terminal Optimization and Evidence Collapse

Figure 3 follows all registered validation checkpoints. From update 0 to 100, both objectives improve CV and PVS while DA, DER, ETC, and cost decline (Table 1). DER declines from initialization in all three runs for both objectives. Across the three runs, the Spearman correlation between optimized reward and DER is -0.712 for Outcome-GRPO and -0.915 for Certificate-GRPO; the corresponding reward–ETC correlations are -0.415 and -0.857. These data show that the decline is not merely an artifact of a fixed initial offset. They do *not* identify reward hacking or any unique cause. Lower cost and early stopping are directly incentivized, the base policy already has biases, and the verifier remains sound relative to the ledger. Without a matched zero-cost or fixed-cost control, we therefore interpret "Evidence Collapse" only as a process-level evidence–cost tradeoff under the registered objective.

Method	**PVS↑**	CV↑	DER↑	ETC↑	Cost↓
Deployable answer-free methods					
Random@4 + Recovery	0.807	0.917	0.765	0.689	5.568
Self-Consistency@4 + Recovery	0.844	0.969	0.722	0.747	5.201
Certificate-only@4 + Recovery	0.802	0.912	0.702	0.704	4.918
Max-Probe@4 + Recovery	0.869	0.908	**0.942**	0.816	7.071
BEATS@4 (ours)	**0.923**	**0.969**	0.885	**0.876**	5.424
Evaluator-only upper bound (not deployable)					
Evaluator-only pool oracle	0.930	0.969	0.661	0.887	4.121

Table 2: Held-out same-pool comparison with uniform ledger-bounded recovery. PVS is the primary endpoint. The evaluator-only oracle accesses hidden PVS and is excluded from the deployable ranking. BEATS achieves the highest deployable PVS, improving over the strongest deployable baseline, Max-Probe@4, by 5.44 points while reducing mean environment investigation cost by 23.3%.

(a) Candidate budget

Setting	PVS↑	CV↑	DER↑	ETC↑	Cost↓
$N = 1$ + Recovery	0.784	0.899	0.707	0.698	4.693
BEATS $N = 2$	0.839	0.942	0.722	0.751	4.424
BEATS $N = 4$	**0.923**	**0.969**	**0.885**	**0.876**	5.424

(b) Cost strata: BEATS / Max-Probe

	PVS↑		Cost↓	
Stratum	BEATS	Max	BEATS	Max
Low	**1.000**	0.990	**2.732**	3.602
Medium	**0.997**	0.977	**5.837**	7.717
High	**0.773**	0.640	**7.703**	9.893

Table 3: Candidate-budget scaling and cost-stratified performance. Increasing N improves PVS, while BEATS achieves higher PVS than Max-Probe at lower environment cost in every cost stratum. Max denotes Max-Probe@4 with the same ledger-bounded recovery.

4.4 Held-Out Evidence-State Search

Figure 4 gives the main test. BEATS reaches 92.33% PVS (pair-clustered 95% CI [87.9, 96.1]), versus 41.78% for greedy, 55.00% for Certificate-only, 58.33% for the distinct raw-validity/no-recovery selector, and 86.89% for deepest-prefix Max-Probe@4+Recovery. Against the strongest deployable baseline, BEATS gains 5.44 points (CI [2.44, 9.00]), records 49 episode-condition wins and no losses, and reduces mean environment cost from 7.071 to 5.424 (23.3%). Max-Probe has higher DER (0.942 versus 0.885), but BEATS yields higher ETC (0.876 versus 0.816), CV (0.969 versus 0.908), and PVS. One plausible benchmark-specific mechanism is that deeper prefixes can add inadmissible or low-reliability sources and protocol opportunities without improving the entailed evidence state. The same-pool comparison supports selective sufficiency; it does not establish that mechanism for every episode.

What isolates the gain? With uniform canonical recovery, Random, Self-Consistency, and Certificate-only reach 0.807, 0.844, and 0.802 PVS; BEATS retains 11.67, 7.89, and 12.11 point advantages. The evaluator-only candidate-pool oracle reaches 0.930, 0.67 point above BEATS, and ties it on 894/900 outcomes. Nested budgets improve from 0.784 at $N = 1$ to 0.839 at $N = 2$ and 0.923 at $N = 4$. These controls isolate evidence-state selection from sampling and recovery. Final aggregate metrics give 872 verifier-valid outputs: 831 PVS outcomes and 41 valid-but-wrong outputs; 28 outputs remain invalid. Thus validity after the joint selection-and-recovery procedure still cannot correct an inadequate selected ledger. The distinct no-recovery selector changes both selection semantics and serialization, so its gap is not a pure recovery treatment effect.

Cost strata and checkpoint profiles. Table 3 shows that additional candidates improve PVS from 0.784 at $N = 1$ to 0.923 at $N = 4$. The gain is not indiscriminate evidence accumulation: BEATS selects a lower-cost trajectory than Max-Probe in every cost stratum, with the largest advantage in the high-cost stratum (+13.3 PVS points at 2.190 fewer cost units). Outcome-GRPO greedy has low DER (0.488); BEATS raises its PVS from 0.571 to 0.951 and ETC from 0.622 to 0.920. SFT greedy acquires much evidence (DER 0.961) but has CV 0.282; BEATS raises PVS to 0.896 and CV to 0.938. Thus the joint selection-and-recovery procedure improves both profiles through different mixtures of selected trajectories and serialization. BEATS also exceeds Max-Probe at all six seeds, by 4.0–6.7 PVS points.

Descriptive robustness. On the synthetic shift suite, BEATS obtains 0.912 aggregate PVS versus 0.852 for Max-Probe@4+Recovery. A matched Gemma-3-12B-IT replication also favors BEATS on IID and shifted episodes Gemma Team (2025); full results and scope limitations are reported in Supplementary Sec. S8.

Compute. $N = 4$ uses 2.073× reconstructed prompt tokens and 1.923× model calls relative to greedy. Environment-cost savings are therefore not total-compute savings; detailed accounting is in Supplementary Sec. S9.

5 Conclusion

Verification is usually treated as an end-of-trajectory property: given an answer and its supporting record, a checker decides whether the answer is justified. For an information-seeking agent, however, the record is not fixed. It is itself an output of the policy. This creates a gap between *record-relative validity* and *complete-record correctness*: an agent can submit a certificate that is fully supported by the evidence it acquired while missing an accessible source that would reverse the decision.

LegalEvidenceGym makes this gap observable. In the static setting, correct decisions frequently lack valid certificates. In the interactive setting, cost-penalized GRPO improves its optimized reward and terminal certificate validity while losing decisive evidence and becoming less reliable on Evidence-Twin pairs. We call this pattern *Evidence Collapse*. The point is not that the verifier is wrong: it remains sound for the ledger it is given. Nor do these experiments establish reward hacking as the unique cause. They show a more basic limitation: terminal reward and record-relative verification do not, by themselves, reward or certify adequate investigation.

BEATS uses this observation at test time. Rather than selecting a final answer from several samples, it selects among several acquired evidence states using only replayable public features, then serializes a certificate from the chosen ledger. On matched candidate pools with identical recovery, BEATS@4 raises PVS from 86.89% to 92.33% over deepest-prefix Max-Probe@4 and lowers environment investigation cost by 23.3%. This result indicates that the relevant object for selection is not only the final text, but also the information state from which that text is produced.

Beyond legal adjudication, this boundary exists in any agentic workflow where information acquisition is policy-controlled and interaction is costly—such as medical diagnosis, financial auditing, and tool-augmented web search. Our findings suggest that future RLVR paradigms must move beyond static terminal verification, explicitly penalizing premature stopping and ensuring evidence completeness during policy optimization.

Reproducibility statement

The supplementary material documents the formal definitions, benchmark construction, training and frozen-evaluation protocols, statistical analysis, implementation details, ablations, compute accounting, and artifact map. The source bundle also includes the plotted figure data, plotting scripts, and the synthetic rule-program files used by the paper.

REFERENCES

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Martin Katz, and Nikolaos Aletras. Lexglue: A benchmark dataset for legal language understanding in english. In *Annual Meeting of the Association for Computational Linguistics*, 2022.

Qinchuan Cheng, Jiaqi Liu, Ruixuan Xie, Xiaoya Yuan, and Yuxin Liu. MAP-Law: Coverage-driven retrieval control for multi-turn legal consultation. *arXiv preprint arXiv:2605.01486*, 2026a.

Zerui Cheng, Jiashuo Liu, Chunjie Wu, Jianzhu Yao, Pramod Viswanath, Ge Zhang, and Wenhao Huang. VeRA: Verified reasoning data augmentation at scale. *arXiv preprint arXiv:2602.13217*, 2026b.

Odysseas S. Chlapanis, Dimitrios Galanis, and Ion Androutsopoulos. LAR-ECHR: A new legal argument reasoning task and dataset for cases of the european court of human rights. In *Proceedings of the Natural Legal Language Processing Workshop 2024*, pp. 267–279. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.nllp-1.22.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Rei Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Wu Fei, Shuxian Liang, Yibo Yang, Yang Lin, Jing Tang, Lei Chen, Xiansheng Hua, and Hao Kong. Self-guided process reward optimization with redefined step-wise advantage for process reinforcement learning. *arXiv preprint arXiv:2507.01551*, 2025.

Zhiwei Fei et al. Lawbench: Benchmarking legal knowledge of large language models. In *Conference on Empirical Methods in Natural Language Processing*, 2024.

Matt Gardner et al. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP*, 2020.

Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Neel Guha et al. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023.

Lukas Helff, Quentin Delfosse, David Steinmann, Ruben Härle, Hikaru Shindo, Patrick Schramowski, Wolfgang Stammer, Kristian Kersting, and Felix Friedrich. Llms gaming verifiers: Rlvr can lead to reward hacking. *arXiv preprint arXiv:2604.15149*, 2026.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. Cuad: An expert-annotated nlp dataset for legal contract review. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021.

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. In *International Conference on Learning Representations*, 2022.

Lauro Langosco, Jack Koch, Lee Sharkey, Jacob Pfau, and David Krueger. Goal misgeneralization in deep reinforcement learning. In *International Conference on Machine Learning*, 2022.

Minghan Li, Junjie Zou, Xinxuan Lv, Chao Zhang, and Guodong Zhou. S2G-RAG: Structured sufficiency and gap judging for iterative retrieval-augmented QA. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 25846–25862. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1185.

Xiao Liu et al. Agentbench: Evaluating llms as agents. In *International Conference on Learning Representations*, 2024.

Xiaoqian Liu, Ke Wang, Yuchuan Wu, Fei Huang, Yongbin Li, Junge Zhang, and Jianbin Jiao. Agentic reinforcement learning with implicit step rewards. *arXiv preprint arXiv:2509.19199*, 2026.

Denis Merigoux, Nicolas Chataing, and Jonathan Protzenko. Catala: A programming language for the law. In *ACM SIGPLAN International Conference on Functional Programming*, 2021.

Massimiliano Pronesti, Anya Belz, and Yufang Hou. Beyond outcome verification: Verifiable process reward models for structured reasoning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 32187–32202, 2026. doi: 10.18653/v1/2026.findings-acl.1611.

Jiaqi Shao, Yuxiang Lin, Munish Prasad Lohani, Yufeng Miao, and Bing Luo. Do LLM agents know how to ground, recover, and assess? a benchmark for epistemic competence in information-seeking agents. *arXiv preprint arXiv:2509.22391*, 2025.

Zhihong Shao et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Jiaxin Shen, Jinan Xu, Huiqi Hu, Luyi Lin, Guoyang Ma, Fei Zheng, Fandong Meng, Jie Zhou, and Wenjuan Han. A law reasoning benchmark for LLM with tree-organized structures including factum probandum, evidence and experiences. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 17252–17274. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.887.

Abhijeet Sinha, Sundari Elango, and Dianbo Liu. Expected return causes outcome-level mode collapse in reinforcement learning and how to fix it with inverse probability scaling. *arXiv preprint arXiv:2601.21669*, 2026.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

Tianyang Xu, Dan Zhang, Kushan Mitra, and Estevam Hruschka. Verification-aware planning for multi-agent systems. *arXiv preprint arXiv:2510.17109*, 2026.

An Yang et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, 2023a.

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

Yichi Zhang, Nabeel Seedat, Yinpeng Dong, Peng Cui, Jun Zhu, and Mihaela van de Schaar. Guideline-grounded evidence accumulation for high-stakes agent verification. *arXiv preprint arXiv:2603.02798*, 2026.