QUALIFYING AI AGENTS UNDER COST-AWARE HUMAN OVERSIGHT

Anonymous authors
Paper under double-blind review

ABSTRACT

As AI agents move into real workflows, evaluation must answer more than whether an agent can complete a task. Deployment requires determining under what operating conditions a system can meet a specified reliability requirement at acceptable cost, and whether the available evidence supports that operating point. We introduce *Reliable Enterprise Agent Deployment* (READY), a framework that turns workflow-level evaluation evidence into a statistically supported deployment qualification. READY preserves each workflow's definition of successful execution while evaluating the complete agent–workflow–oversight configuration. Given a prespecified class of oversight policies, it evaluates their reliability and operating cost, tests them on held-out cases with finite-sample error control, and returns the lowest-cost member of the certified set. The resulting deployment profile characterizes reliability, autonomous coverage, human-oversight burden, and cost rather than reducing a system to a single benchmark score. We instantiate READY in the terminal accept-or-escalate setting on CliniCARE-Bench clinical audit (18 systems, 750 cases) and τ^2-bench retail (17 systems, 114 tasks). Across both workflows, autonomous accuracy shows no clear correlation with how well self-reported confidence ranks successes above failures (Pearson $r = -0.07$ and -0.13). Consequently, systems with similar autonomous performance can require substantially different oversight to meet the same reliability target. On clinical audit, Sonnet 5 and GPT-5.4 differ by only 0.3 points in development accuracy, yet their certified operating points require 12.0% versus 29.3% human review at the same target. Accounting for agent and review costs can further change which certified configuration is preferred. These results show that deployability is not an intrinsic property of an agent or its benchmark score, but of an agent-workflow-policy configuration under explicit assumptions, with routing quality, statistical uncertainty, and cost jointly shaping the reliability-oversight-cost tradeoff.

1 INTRODUCTION

AI agents increasingly perform professional work that requires reasoning over domain knowledge, using tools, and carrying out multi-step tasks. A growing class of benchmarks measures how well agents perform such work and the quality of the resulting work product (Sun et al., 2026; Patwardhan et al., 2026; Chen et al., 2025b). Deployment asks a different question: under what operating conditions can an agent meet the required reliability at acceptable cost, and what evidence supports that operating point? Answering this question requires specifying when the agent may operate autonomously, when oversight or intervention is required, and how reliability and cost trade off. The relevant object of evaluation is therefore not the agent alone, but the agent-workflow-policy configuration under which it operates. Two agents with similar benchmark performance may support very different deployment configurations if they differ in which cases can reliably be handled autonomously or how much intervention is required to meet the same reliability requirement.

We introduce *Reliable Enterprise Agent Deployment* (READY), a framework that turns workflow-level evaluation into deployment qualification. READY takes an agent, a workflow, representative cases, a prespecified class of candidate oversight policies, and declared deployment requirements. Each workflow retains its own definition of successful execution, which may incorporate outcome correctness, process, grounding, or policy adherence. READY evaluates each candidate policy for reliability and operating cost, tests whether it meets the deployment requirements on held-out cases

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.

with finite-sample error control, and selects the lowest-cost qualifying policy. The resulting *deployment profile* is a statement about a particular agent-workflow-policy configuration under stated assumptions, rather than an intrinsic score or universal certification of the agent. To avoid optimistic bias from selecting and evaluating policies on the same data, READY separates policy development from held-out qualification. Its qualification procedure builds on finite-sample risk control in the style of Learn-then-Test (Bates et al., 2021; Angelopoulos et al., 2025), while making the workflow-specific success criterion, oversight process, and operating cost explicit components of the deployment claim.

We instantiate the framework in the terminal accept-or-escalate setting, in which the agent completes its work before a policy decides whether to accept the result or route it to human review. We evaluate READY on two workflows: CliniCARE-Bench clinical audit (18 systems, 750 retrospective cases) (Chatrath et al., 2026) and the retail workflow of τ^2-bench (17 systems, 114 tasks) (Barres et al., 2026). Across both workflows, autonomous accuracy shows no clear correlation with how well self-reported confidence ranks successful executions above failures. As a result, systems with similar autonomous capability can require substantially different amounts of oversight to support the same reliability target. On clinical audit, Sonnet 5 and GPT-5.4 differ by only 0.3 points in development accuracy, yet their certified operating points require 12.0% versus 29.3% human review at the same reliability target of 75% and the assumed 90% review-success rate. The lowest-cost certified configuration can further change as agent execution costs, human-review quality, and review costs vary. Taken together, these results highlight the central contribution of READY: a reliability-oversight-cost qualification layer on top of workflow-level execution evidence. Section 2 positions READY relative to prior work. We make three contributions.

1. **A deployment-centered objective.** For a given agent-workflow configuration, we define the deployment objective as finding the lowest-cost oversight policy that satisfies a specified reliability requirement, with success determined by a workflow-specific predicate that can incorporate outcome, process, grounding, and policy requirements (Section 3).
2. **A held-out policy-qualification procedure.** We separate policy development from statistical qualification, testing a prespecified policy family on held-out evidence with finite-sample error control and reporting the lowest-cost member of the resulting certified set (Section 3.3).
3. **An end-to-end empirical instantiation.** Using workflows in clinical audit and retail, we show that autonomous capability alone does not determine the supported deployment configuration: routing quality, statistical uncertainty, human-review burden, and agent cost jointly shape the operating point. The clinical-audit study additionally evaluates sensitivity to the workflow-specific definition of successful execution (Sections 4 and 5).

2 Related Work

READY connects six lines of prior work into one account of reliability, oversight, and cost. We summarize the closest work here, with Table 15 in App. G giving the full comparison.

Capability, process, and work-product benchmarks. Evaluation has moved from closed-ended knowledge exams (Center for AI Safety et al., 2026) to executable professional work that scores whether an agent can *do* the task (Sun et al., 2026; Patwardhan et al., 2026; Chen et al., 2025b), alongside enterprise, web, and tool-use benchmarks and process- and evidence-oriented evaluations that grade intermediate behavior rather than only the final answer (Sun et al., 2025; Chen et al., 2025a). Closest to our motivation, Kapoor et al. (2025) show that accuracy-only benchmarks can reward needlessly costly agents and argue for reporting cost and accuracy on held-out data, and τ-bench's *pass*^{k} captures consistency across attempts (Yao et al., 2025). READY reuses this evidence, but asks which oversight policy meets a reliability target at minimum cost.

Selective prediction, deferral, and cost-aware routing. READY's accept-or-escalate setting is an instance of selective prediction (Chow, 1970; El-Yaniv & Wiener, 2010; Geifman & El-Yaniv, 2017), and its human-review path is closely connected to learning to defer (Madras et al., 2018; Mozannar & Sontag, 2020). The risk–coverage curve and AURC we use in Section 3.2 come directly from this literature, and our minimum-cost objective connects READY to model cascades

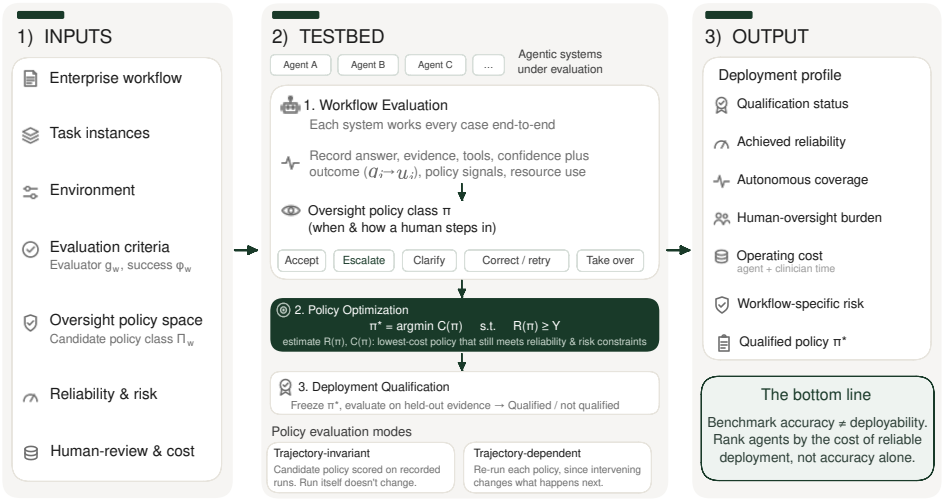


Figure 1: *Reliable Enterprise Agent Deployment* (READY) takes a workflow and evaluation specification, a candidate oversight-policy class, and deployment requirements. For each system, execution and evaluation generate instance-level evidence, candidate policies are evaluated for reliability and cost, and a prespecified family is tested on held-out cases. The output is the lowest-cost member of the resulting certified set together with its reliability, oversight burden, cost, scope, and assumptions.

and routing that hit a quality bar at lower cost by escalating between *models* (Chen et al., 2024b; Ong et al., 2025). Selective classification with guaranteed risk (Geifman & El-Yaniv, 2017) and distribution-free risk control (Bates et al., 2021; Angelopoulos et al., 2025; Laufer-Goldshtein et al., 2023) certify the risk of accepted predictions or a decision-rule family. More broadly, READY follows the classical decision-theoretic view that thresholds should reflect the costs of error and escalation, not accuracy alone. READY applies this style of guarantee to the reliability of the whole human–AI system, including the reviewed cases. Unlike cascades and routers, READY defers to a human reviewer rather than a stronger model, replaces a single average operating point with a statistically qualified reliability floor on held-out data, and uses a versioned workflow-specific success predicate as its quality bar, whose changes can move the supported operating point even when executions are fixed (Sec. 5.4).

Routing signals, human oversight, and assurance. Oversight often needs a signal that ranks executions by likelihood of success, including elicited and verbalized confidence, semantic uncertainty, and calibration (Kadavath et al., 2022; Kuhn et al., 2023; Lin et al., 2022; Guo et al., 2017). READY, however, is signal-agnostic. It asks only for the score that orders cases usefully, and folds any extra inference cost into $C_{m,w}(\pi)$. Human–AI teams often underperform the stronger component (Vaccaro et al., 2024), and mandated review can give false assurance (Green, 2022). READY takes this seriously. Rather than assume review helps, it treats the review path's success as a measured quantity or an explicit assumption and qualifies the system on held-out evidence, which complements assurance frameworks such as model cards (Mitchell et al., 2019). General evaluation harnesses such as Inspect (UK AI Security Institute, 2024), Harbor (Harbor Framework Team, 2026), and AgentBeats (Liu et al., 2026) supply the executable substrate beneath this qualification layer.

3 Method: Deployment Qualification

We formalize deployment qualification for a specified agent–workflow configuration. An *oversight policy*, π, determines when the agent may operate autonomously and when a declared human-review path intervenes. READY separates four logical stages. **Execution and workflow evaluation** produce instance-level evidence, the **policy specification** defines the allowed intervention rules, **policy optimization** evaluates candidate reliability and cost, and **deployment qualification** tests the candidates on held-out evidence and selects the lowest-cost member of the certified set. The output is a *deployment profile* for that complete configuration under its stated assumptions (Figure 1).

3.1 Workflow Specification and Deployment Objective

Three specifications are kept distinct. The *workflow specification* identifies the population, task instances, and execution environment. The *evaluation specification* defines the evaluator, success predicate, and observable routing signals. The *deployment specification* declares the policy class, reliability target, review model, and cost assumptions. This separation lets the same execution evidence be reconsidered under new deployment requirements when the candidate policies do not change the saved trajectory. The reference runtime and versioning details are given in Appendix B.

Specifically, a workflow, w, denotes a reusable type of enterprise work with execution environment, E_w, (tools, data interfaces, and systems available during execution) and a population, $\mathcal{D}_w$, over task instances, x. An evaluation set, $S_w = \{x_i\}_{i=1}^{N_w}$, contains independent task instances, with repeated trials grouped by task. Executing agent, m, on x in E_w under policy, π, produces a trajectory, $Z^\pi(x) = \mathrm{Execute}(m, x, E_w; \pi)$, that contains the final work product and the observable evidence needed to evaluate it (retrieved evidence, tool interactions, interventions, and resource use). Each workflow defines an evaluator, g_w, that maps a trajectory to measurements, $\mathbf{q}^\pi(x) = g_w(Z^\pi(x))$, covering outcome correctness together with relevant process, grounding, or policy aspects, and a pre-specified *success predicate*, ϕ_w, that defines the instance-level outcome, $u^\pi(x) = \phi_w(\mathbf{q}^\pi(x)) \in \{0, 1\}$. For an evaluation instance, x_i, we write $Z_i^\pi := Z^\pi(x_i)$, $\mathbf{q}_i^\pi := \mathbf{q}^\pi(x_i)$, and $u_i^\pi := u^\pi(x_i)$. Deployment reliability and expected operating cost of the human–AI system are then

$$R_{m,w}(\pi) = \mathbb{E}_{x\sim\mathcal{D}_w}[u^\pi(x)]\,, \qquad C_{m,w}(\pi) = \mathbb{E}_{x\sim\mathcal{D}_w}[K_w(Z^\pi(x))]\,, \tag{1}$$

where K_w is the workflow operating cost, including agent execution and human oversight. Given a target reliability Y, READY casts oversight-policy selection as a constrained optimization over a candidate policy class Π_w and selects the lowest-cost feasible policy,

$$\boxed{\pi^\star_{m,w}(Y) \in \arg\min_{\pi\in\Pi_w} C_{m,w}(\pi) \quad \text{s.t.} \quad R_{m,w}(\pi) \geq Y.} \tag{2}$$

If no policy in Π_w meets Y, agent m cannot be qualified on workflow w at that target. Equation 2 is the population objective. As $R_{m,w}$ is generally unknown in practice, READY replaces the unknown feasible set with a statistically certified set and returns its lowest-cost member (Sec. 3.3). The qualification therefore applies to the complete configuration (m, w, π, Y) and its declared assumptions, not to agent m independently of a workflow or policy.

3.2 Terminal Accept-or-Escalate Oversight

We instantiate the trajectory-invariant setting in which the agent completes its work and a policy then routes the result using an observable signal, s_i. Routing occurs after execution, so candidate policies can be evaluated by replaying the same saved runs. A threshold policy, π_τ, accepts autonomously when $s_i \geq \tau$ and escalates otherwise, giving autonomous coverage, $c(\tau) = \Pr(s \geq \tau)$, review rate, $1 - c(\tau)$, and accepted-case success, $a(\tau) = \Pr(u = 1 \mid s \geq \tau)$. With review-path success, $a_h(\tau)$, the reliability of the resulting human–AI system is

$$R(\tau) = c(\tau)\, a(\tau) + \big(1 - c(\tau)\big) a_h(\tau). \tag{3}$$

With per-case agent cost k_m and review cost k_h, the operating cost is $C(\tau) = k_m + k_h\big(1 - c(\tau)\big)$. The population objective is to minimize this cost over the candidate threshold family subject to $R(\tau) \geq Y$ (App. A).

Quality of the routing signal. A useful signal ranks higher-risk cases below lower-risk ones. We summarize this with the selective-risk curve $r(\tau) = \Pr(u = 0 \mid s \geq \tau)$, its area under the risk–coverage curve (**AURC**, lower is better), and the **AUROC** of s for predicting success. AURC captures the absolute risk–coverage tradeoff and depends on the base success rate $a_0 = \Pr(u = 1)$, whereas AUROC isolates ranking discrimination from prevalence. Neither metric is itself a qualification result, and threshold routing requires useful ordering, not numerical calibration of s.

3.3 Policy Certification on Held-Out Evidence

Choosing the cheapest threshold whose empirical reliability meets Y would reuse the same evidence for selection and evaluation. READY instead partitions the evaluation sample into a development

set used for routing diagnostics and any data-driven design of the candidate family, and a held-out qualification set used only for certification. When the review path is not executed and a constant review-success rate assumption a_h is declared, candidate τ has qualification-unit outcome

$$y_i(\tau; a_h) = d_i(\tau)u_i + \big(1 - d_i(\tau)\big)a_h, \qquad d_i(\tau) = \mathbf{1}\{s_i \geq \tau\}, \tag{4}$$

and bounded loss $L_i(\tau) = 1 - y_i(\tau; a_h)$. Thus both the observed autonomous outcomes and the fraction routed to review enter the reliability test, while the review-path performance is held fixed as an explicit assumption rather than estimated by that test.

For each candidate, READY tests $H_\tau : R(\tau) < Y$ using a finite-sample-valid p-value for bounded losses. Candidates are tested in a prespecified fixed sequence from review-everything toward accept-everything, stopping at the first non-rejection. Thus, no additional multiplicity correction is required within the candidate family. The rejected prefix is the certified set $\widehat{\mathcal{T}}_\alpha(Y)$, and the deployment profile reports its lowest-cost member,

$$\widehat{\tau}(Y) \in \arg\min_{\tau \in \widehat{\mathcal{T}}_\alpha(Y)} C(\tau). \tag{5}$$

With probability at least $1 - \alpha$, no member of this certified set has reliability below Y. The same test is inverted to report a one-sided lower confidence bound. App. A.5 gives the betting p-value, measured-review variant, and clustered sampling rule. The guarantee does not require reliability to be monotone in τ, but "lowest cost" means lowest within the certified prefix. It approaches the globally cheapest feasible threshold only under a nested-feasibility condition and with enough qualification data (App. E.1). Certificates are pointwise for one prespecified system and target.

4 Experimental Design

We instantiate the same terminal qualification procedure on two workflows that share policy family, certification procedure, and routing signal, but differ in task distribution, evaluator, success predicate, and sampling unit. Full execution and reporting protocols are in Appendices B, C, and D.1.

4.1 Workflow Instantiations

CliniCARE-Bench. For each retrospective clinical-audit case (Chatrath et al., 2026), an agent receives an audit question and patient-scoped access to longitudinal electronic health records, then returns an auditable adjudication with citations and stated confidence. The reference verdicts are

$$\mathcal{Y} = \{\text{Yes},\ \text{No},\ \text{Ind: Lack of Data},\ \text{Ind: Medically Ambiguous}\}. \tag{6}$$

The two indeterminate outcomes are valid task-level clinical judgments, distinct from the deployment-level decision to review a completed result. With reference adjudication $y_i^\star$ and agent adjudication $\widehat{y}_i$, the primary success predicate is

$$u_i = \mathbf{1}\{\widehat{y}_i = y_i^\star\}. \tag{7}$$

A process-aware robustness analysis additionally requires no disqualifying process defect,

$$u_{i,\mathrm{df}} = \mathbf{1}\{\widehat{y}_i = y_i^\star \ \wedge\ \text{no disqualifying process defect}\}. \tag{8}$$

The defect indicator is the benchmark's own per-run process flag, produced by CliniCARE-Bench's model-assisted process grader in a post-hoc pass over each transcript (Chatrath et al., 2026). We use it as released. All 18 systems run the 750 cases once, with reference labels following the natural cohort distribution and a fixed 375/375 split shared across systems. Each case comes from a distinct patient, making it an independent qualification unit. Confidence resolution on the qualification partition ranges from 4 to 33 distinct values, changing the set of reachable routing policies.

τ^2 retail. In the retail workflow of τ^2-bench (Barres et al., 2026), an agent resolves a customer's request through conversation and retailer tools under a written policy. All 17 systems run each of the 114 tasks four times (7,752 trials), with success given by the benchmark's all-or-nothing reward. Repeated trials share a task scenario, so the task is the sampling unit. The split holds 57 development and 57 qualification tasks, and each task-level loss averages its four trials. The simulated customer is an external user actor, not the assumed human reviewer. The simulator and the natural-language assertion judge are GPT-5.2, the leaderboard setting, which could favor GPT-family agents. App. D.1 gives the full protocol and harness-specific reporting safeguards.

4.2 Prespecified Qualification Protocol

Neither study executes the human-review path, so we set review success to the deployment assumption $a_h = 0.90$ and recompute profiles under other values in App. C.4. Every certificate is conditional on this value. Since the primary target $Y = 0.75$ lies below a_h, review-everything qualifies by assumption, so the informative comparison is the review burden and total cost needed to reach the target rather than binary system-level feasibility.

In both workflows the candidate family contains the threshold policies π_τ of Section 3.2 over the prespecified grid $\mathcal{G} = \{0, 0.01, \ldots, 1\}$ plus review-everything, scored with the case-level outcome of Equation 4 (a case as the qualification unit in CliniCARE, a task-averaged outcome in retail). Candidates are tested from review-everything toward accept-everything with the betting p-value for bounded means of Waudby-Smith & Ramdas (2024) at $\alpha = 0.05$. For each operating point we report its review burden, reliability estimate, and one-sided 95% lower confidence bound, with complete test construction and reproduction steps in Appendices A.5 and C.2.

Agent cost k_m comes from recorded token usage at list prices (App. C.6), and the review price k_h is swept. With k_m fixed per system, its lowest-review certified point is also its lowest-cost one.

5 Results

We show that capability does not determine routing quality, that routing quality, confidence resolution, and uncertainty set the certified review burden, that review and agent price jointly set cost, and that these findings are robust to workflow, partition, review assumption, and success definition.

5.1 Autonomous Capability Does Not Determine Routing Quality

Sweeping the threshold on development data traces each system's risk–coverage curve (App. C.1). Selective risk generally falls as coverage drops, as an informative signal should, but routing quality is not implied by autonomous accuracy. Across the 18 CliniCARE systems, base accuracy a_0 and AUROC show no clear correlation (Pearson $r = -0.07$, 95% CI $[-0.52, 0.41]$, Figure 2). Qwen-3.7-Plus pairs near-lowest accuracy with the best AUROC. The same holds in retail ($r = -0.13$) and under an idealized agreement signal built from repeated-trial rewards (App. D.4). The two signals rank systems almost independently, so routing quality is a property of a system paired with a signal rather than of the system alone. AURC tracks base performance more closely ($r = -0.80$ on CliniCARE and -0.60 on retail) so the two diagnostics answer different questions. AURC describes the absolute risk–coverage tradeoff, whereas AUROC describes ranking quality net of the base rate. Complete routing curves and per-system metrics are in Tables 2 and 11.

5.2 Certified Reliability–Oversight Frontiers

Routing quality becomes operational when translated into the oversight required to meet a target. For each target Y, we compute the fixed-sequence certified set on S_{qual} and report its lowest-review member. Figure 3 plots the resulting CliniCARE reliability–oversight frontier. At a fixed target, a system that requires less review supports a more oversight-efficient certified configuration under the same review assumption.

At $Y = 0.75$, all 18 systems certify an operating point that accepts some cases autonomously. Mean review is 37.5%, ranging from 12.0% to 85.6% (Table 1). Low-review certification can fail for two distinct reasons. First, a favorable point estimate may still lack sufficient evidence. GLM-5.2's 10.9%-review threshold estimates reliability at 0.762 but has a lower bound of 0.705, so its reported point requires 32.0% review. Second, coarse confidence produces few reachable policies. Gemini-3.1-Pro has only four distinct confidence values on the qualification partition and jumps from 33.6% review at $Y = 0.75$ to full review at $Y = 0.80$. Such jumps reflect the signal rather than the system. On retail, where its stated confidence likewise certifies only at full review, an idealized agreement signal built from repeated-trial rewards certifies the same model at 6.1% review (App. D.4). Thus the frontier depends on autonomous outcomes, routing resolution, and statistical uncertainty rather than on a point accuracy alone. The certificate itself is checked in App. A.6: over 1,000 redrawn splits, the certified threshold misses the target on the held-out half in 0.9% (CliniCARE) and 1.5% (retail) of draws, against 31% and 20% for a naive point-estimate pick.

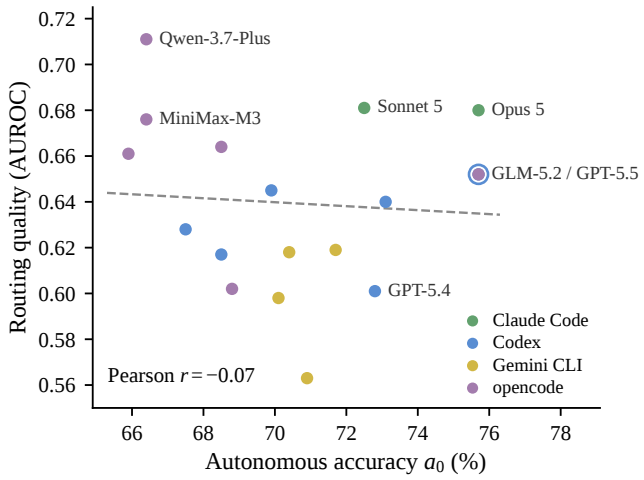


Figure 2: **Autonomous capability does not determine routing quality.** Four-way verdict accuracy a_0 versus routing-signal AUROC for all 18 CliniCARE-Bench systems on the development partition, colored by agent harness. The quantities show no clear correlation (Pearson $r = -0.07$, with the ordinary-least-squares fit dashed). Per-system values are reported in Table 2.

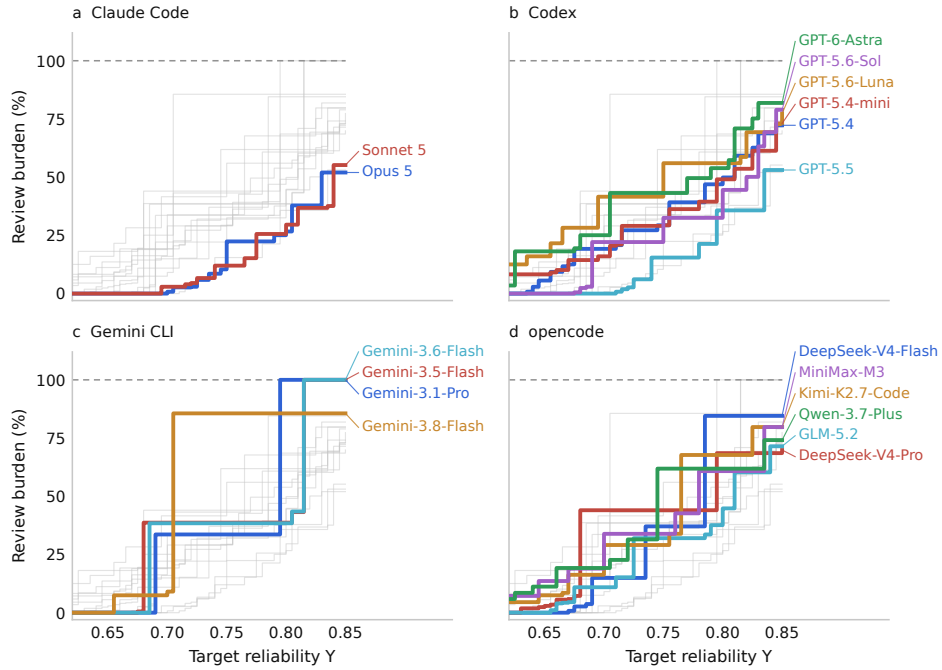


Figure 3: **Reliability–oversight frontier on the held-out CliniCARE qualification partition**, by agent harness, at $a_h = 0.90$. Each curve is the review burden of the lowest-review member of the fixed-sequence certified set as target Y increases. The curves are monotone step functions because the certified prefix shrinks with Y and confidence ties make the reachable thresholds discrete. At $Y = 0.85$, review burden ranges from 52.0% to 100%.

Matched-capability comparison. Across the 18 systems, certified review burden at $Y = 0.75$ has only moderate rank association with accuracy (Spearman $\rho = -0.45$) and less with development AUROC ($\rho = -0.26$). It tracks confidence resolution more closely ($\rho = -0.58$ with the number of distinct values). The paired comparison selected on S_{dev} is especially clear. GPT-5.4 ($a_0 = 72.8$) and Sonnet 5 ($a_0 = 72.5$) differ by 0.3 accuracy points, but Sonnet 5 has stronger routing discrimination (AUROC 0.681 versus 0.601). On held-out cases, Sonnet 5 certifies at 12.0% review

($\widehat{R} = 0.817$, LCB 0.760), whereas GPT-5.4 requires 29.3% ($\widehat{R} = 0.803$, LCB 0.750). The ordering survives a Bonferroni correction across all 18 systems ($\alpha/18$): both still certify, Sonnet 5 at 25.6% review (LCB 0.770) and GPT-5.4 at 39.2% (LCB 0.755). The pair is matched on development accuracy only. On the qualification half Sonnet 5 is also more accurate (about 75% versus 69%, Table 9), so part of this gap reflects accuracy as well as routing. Better AUROC does not guarantee less review at every target, because threshold granularity and uncertainty matter. The complete frontier is therefore the relevant comparison.

5.3 Cost of a Certified Deployment

Review burden is only one part of deployment cost. In the terminal setting, total operating cost per case is $C = k_m + k_h e$, the agent execution cost k_m plus the review price k_h times the certified review burden e. We record k_m from token usage at the evaluation gateway's list prices (App. C.6) and sweep k_h from \$0.50 to \$100 per reviewed case. Figure 4 shows that the preferred system is not determined by either accuracy or review burden alone.

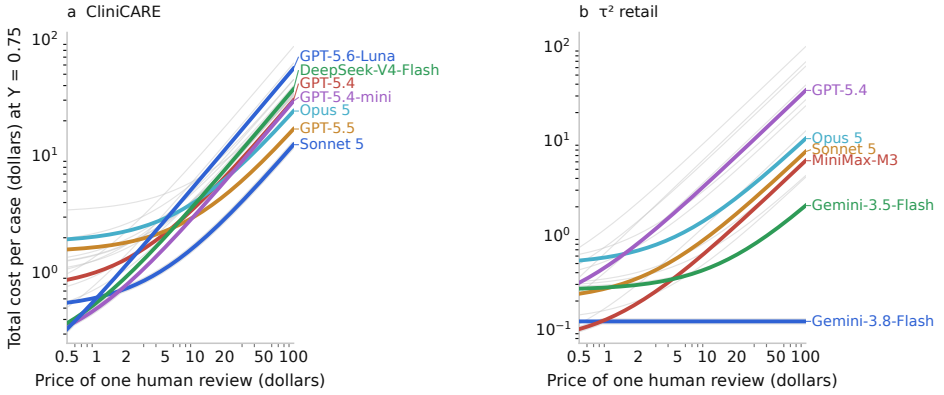


Figure 4: **Certified total cost versus review price** at $Y = 0.75$ and $a_h = 0.90$, where $C = k_m + k_h e$. On CliniCARE the lower envelope changes from GPT-5.6-Luna to GPT-5.4-mini to Sonnet 5. On retail, Gemini-3.8-Flash is cheapest above \$0.85.

On CliniCARE, GPT-5.6-Luna is cheapest at review prices of roughly \$0.50–\$0.57, GPT-5.4-mini at \$0.58–\$1.69, and Sonnet 5 above \$1.70. Sonnet 5 combines a mid-range agent cost with the lowest certified review burden, so cheaper agents remain preferable only while review itself is inexpensive. None of the three systems tied for highest clinical accuracy (Opus 5, GPT-5.5, and GLM-5.2) lies on the lowest-cost envelope of reported certified configurations. On retail, agent costs are smaller and more similar, so certified review burden dominates more of the ranking.

	Claude Code		Codex						Gemini CLI				opencode					
	Opus 5	**Sonnet 5**	**5.6-Sol**	**5.6-Luna**	**5.5**	**5.4**	**5.4-mini**	**6-Astra**	**3.8-Flash**	**3.6-Flash**	**3.5-Flash**	**3.1-Pro**	**DS-V4-Pro**	**DS-V4-Flash**	**GLM-5.2**	**Qwen-3.7**	**MiniMax-M3**	**Kimi-K2.7**
Review	22.4	12.0	32.5	56.0	15.5	29.3	29.3	43.2	85.6	38.4	38.7	33.6	44.0	37.1	32.0	61.9	33.9	29.1
$\widehat{R}$	83.9	81.7	84.2	85.3	83.0	80.3	80.3	81.8	90.6	85.0	84.7	83.8	83.9	83.2	82.9	87.4	80.9	80.6
$\mathbf{LCB_{95}}$	78.5	76.0	79.5	81.0	77.5	75.0	75.0	77.0	85.0	80.0	80.0	79.0	79.0	78.0	78.0	83.5	75.5	75.0

Table 1: **CliniCARE certified operating points at $Y = 0.75$.** Percentages, with $N_{\text{qual}} = 375$ and $a_h = 0.90$. Each column is the lowest-review certified point. `Codex` columns are GPT-x, `Gemini CLI` columns are Gemini-x, and DS-V4 is DeepSeek-V4.

5.4 Robustness and Scope

Human-review assumption and evaluation partition. The profiles are conditional on review-path effectiveness. At $Y = 0.75$, Sonnet 5 remains at 12.0% review for $a_h \in \{0.80, 0.90, 0.95\}$, whereas GPT-5.4 moves from 46.9% to 29.3% to 27.2%, so the comparison narrows but does not reverse. Across all systems, weakening a_h from 0.90 to 0.80 raises mean certified review over the sensitivity sweep by 13 percentage points. The direction of the Sonnet 5–GPT-5.4 comparison also persists when certification is run separately on the development half or on the full cohort, although the magnitude changes with partition and sample size. App. C.4 reports the full review-assumption analysis, and App. C.3 reports the partition and sample-size sensitivity.

Replication on τ^2 retail. Retail confirms that accuracy does not determine review burden. At $Y = 0.75$, 16 of 17 systems certify a policy that accepts some trials autonomously, with mean review 24.0%. Sonnet 5 certifies at 7.5% review ($\widehat{R} = 0.887$, LCB 0.790), whereas GPT-5.4 requires 34.2% ($\widehat{R} = 0.843$, LCB 0.750), even though GPT-5.4 has the higher development AUROC (0.784 versus 0.692). Routing quality alone does not explain this gap, since held-out accuracy, confidence granularity, and sampling uncertainty all shape it. Granularity matters elsewhere too. Gemini-3.5-Flash and Opus 5 tie on development accuracy, but at $Y = 0.80$ the former, whose stated confidence takes only a few distinct values, has no certified point below full review while the latter certifies at 15.8%. With only 57 independent qualification tasks, effect sizes are less precise than in CliniCARE. Complete tables and frontiers are in App. D.

Workflow success definition. Replacing verdict correctness with the process-aware predicate $u_{i,\mathrm{df}}$ leaves the ranking signal broadly similar (per-system AUROC changes by at most 0.032) but substantially changes the deployment profiles. Mean certified review across the target sweep rises from 26.7% to 49.6%, and at $Y = 0.75$ all three process-graded Gemini systems move from 33.6–38.7% review to no certified point below full review. This is the distinction the workflow-specific success predicate makes, since accuracy-only deferral would miss process defects (App. C.5).

Alternative policy families. READY certifies any prespecified family. A rule that always reviews indeterminate verdicts and thresholds confidence on the rest certifies at 31.5% mean review at $Y = 0.75$, against 37.5% for confidence alone, and needs less review for 10 of 18 systems. Over the full target sweep the two tie (28.9% vs. 29.0%). Cheap structural rules can therefore match self-reported confidence, but selecting between families is itself a qualification result.

6 Limitations

Our experiments validate the core procedure only for trajectory-invariant terminal routing with binary mean reliability. They do not empirically validate intervention during execution or the risk-sensitive extensions in the appendix. The human-review path is also not executed. Its success rate, latency, and price are deployment assumptions, and the constant a_h model ignores case-dependent reviewer difficulty. The retail qualification partition contains only 57 independent tasks, and both studies route on self-reported confidence whose resolution and stability may change across prompts, harnesses, and model versions. Certificates are pointwise for a prespecified system and target, not simultaneous over every system and target shown, so picking the cheapest system in Figure 4 would itself require splitting α across systems (App. C.8). Finally, each result applies only to the evaluated population and versioned configuration. Distribution shift, workflow changes, and system updates require monitoring and requalification. App. F discusses these limitations and future extensions.

7 Conclusion

Autonomous benchmark performance does not determine a deployed human–AI workflow's reliability or cost. *Reliable Enterprise Agent Deployment* (READY) qualifies the complete configuration by testing prespecified oversight policies on held-out evidence and reporting the lowest-cost certified option under declared assumptions. Across clinical audit and retail, similar accuracy requires different review, and agent cost can change the preferred deployment. The relevant comparison is which qualified configuration meets the required reliability at an acceptable cost.

AI USE STATEMENT

Generative AI tools were used in preparing this work. All AI-assisted output was reviewed and verified by the authors, who take full responsibility for the final content, including all text, claims, and artifacts. *Writing.* AI-based assistants were used to reorganize the manuscript, draft and revise prose from author-provided material, and improve LaTeX formatting. The authors are responsible for reviewing the resulting text and verifying all scientific claims and interpretations. *Code.* AI coding assistants helped implement and refactor the evaluation testbed, the oversight-policy optimization and statistical-qualification routines, and the analysis and plotting scripts. All such code was read, tested, and validated by the authors, and every reported number was reproduced from the underlying evaluation logs. *Literature.* AI tools were used to help surface related work, but every cited reference was read and verified by the authors against its primary source rather than taken from AI output. *Method and experiments.* Generative AI systems are the *object* of study rather than a means of producing our findings. The evaluated systems are AI agents, and, as described in Section 4, the CliniCARE-Bench workflow evaluator is in part model-assisted, the τ^2 retail customer is a model-based simulator and part of its evaluator is model-judged, and the routing signal is each agent's self-reported confidence. These uses are part of the experimental design and are documented in the main text and appendices. We did not use generative AI to fabricate, alter, or select data or results, and no AI system is listed as an author.

ETHICS STATEMENT

Human subjects and data. This paper introduces an evaluation *methodology* and collects no new human-subjects data. Our empirical study reuses CliniCARE-Bench (Chatrath et al., 2026), an existing, separately published benchmark of retrospective clinical-audit cases derived from de-identified electronic health records; this work introduces no protected health information, and all clinical examples shown are de-identified. The underlying records are drawn from MIMIC-IV (Johnson et al., 2023), accessed under the PhysioNet Credentialed Health Data Use Agreement. Model calls were routed through the evaluation gateway under enterprise terms that exclude training on prompts and routine human review of content. Data access and use follow the terms established by that benchmark's release, and because the analysis is retrospective and uses only de-identified records with no patient interaction, it required no additional IRB review.

Intended use and potential impact. READY is designed to make the deployment of AI agents in consequential settings *more* cautious, not less: it replaces autonomous benchmark accuracy with a statistically qualified, oversight-aware operating point and makes explicit how much human review a given reliability target requires. The clinical-audit case study is an illustrative research testbed; a *deployment profile* is a conditional statement under stated assumptions, not a clinical endorsement, a safety certification, or a recommendation to operate any system autonomously in patient care. As we note in Section 6, qualification claims are conditional on the human-review assumptions and the evaluated population and do not transfer under distribution shift.

Risks, fairness, and dual use. The framework is evaluative and its clinical instantiation is retrospective and de-identified, so we see limited potential for direct misuse. Requiring held-out statistical qualification and explicit human oversight is intended to reduce the risk of over-trusting capable-looking but unreliable agents. Reported reliability reflects the specific cohort and case mix evaluated and may not hold for other populations; subgroup and fairness analysis of deployment profiles is important future work.

Conflicts of interest. Some authors are affiliated with an organization that develops agent-evaluation tools. This affiliation did not influence the methodology, the systems selected for evaluation, or the reported results, and the framework and testbed are described in sufficient detail to permit independent reproduction.

REPRODUCIBILITY STATEMENT

We have aimed to make both the READY methodology and its empirical instantiation reproducible. The deployment-qualification objective, quantities, and estimators are defined in Section 3, with

the complete derivations, the assumed-review lower-confidence-bound estimator, and all stated assumptions given in Appendix A; routing-signal ordering and the formal, empirically uninstantiated tail-risk (CVaR) qualification procedure are discussed in Appendices E.1 and E.2. For the empirical study, Section 4 specifies both benchmark cohorts, the case- and task-level development/qualification splits, the routing signals, the candidate threshold family and its test order, the betting p-value and error level, and the human-review assumption a_h; certification uses only the qualification partition, while the development partition serves routing diagnostics and partition-sensitivity analysis (Appendix C.3). Appendix C.8 separately identifies its descriptive frontier comparison, and Appendix C.6 gives the recorded token usage and prices behind every agent cost. Appendix C gives a step-by-step reproduction of the certified operating point and its lower bound (Appendix C.2), sensitivity to a_h, robustness under the process-aware success definition, and complete per-system results; Appendix D gives the protocol and complete tables for τ^2 retail, whose τ^2-bench tasks, simulator, and evaluator are pinned to a public upstream commit. The empirical study uses the separately published CliniCARE-Bench benchmark (Chatrath et al., 2026) and τ^2-bench (Barres et al., 2026). To support independent reproduction, an anonymized version of the reference testbed together with the evaluation configurations, split seeds, and analysis scripts used to produce the reported tables and figures is provided as supplementary material.

REFERENCES

Amirhossein Abaskohi, Tianyi Chen, Miguel Muñoz-Mármol, Curtis Fox, Amrutha Varshini Ramesh, Étienne Marcotte, et al. DRBench: A realistic benchmark for enterprise deep research. In *The Fourteenth International Conference on Learning Representations (ICLR)*, pp. 6727–6795, 2026. URL https://openreview.net/forum?id=IGYQ4c92e2.

Pranjal Aggarwal, Aman Madaan, Ankit Anand, Srividya Pranavi Potharaju, Swaroop Mishra, Pei Zhou, Aditya Gupta, Dheeraj Rajagopal, Karthik Kappaganthu, Yiming Yang, Shyam Upadhyay, Manaal Faruqui, and Mausam. AutoMix: Automatically mixing language models. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, volume 37, pp. 131000–131034, 2024. doi: 10.52202/079017-4164. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/ecda225cb187b40ea8edc1f46b03ffda-Abstract-Conference.html.

Satyajith Amaran, Nikolaos V. Sahinidis, Bikram Sharda, and Scott J. Bury. Simulation optimization: a review of algorithms and applications. *Annals of Operations Research*, 240(1):351–380, 2016. doi: 10.1007/s10479-015-2019-x.

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

Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, J. Zico Kolter, Matt Fredrikson, Yarin Gal, and Xander Davies. AgentHarm: A benchmark for measuring harmfulness of LLM agents. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*, pp. 79185–79220. OpenReview.net, 2025. URL https://openreview.net/forum?id=AC5n7xHuR1.

Anastasios N. Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *The Annals of Applied Statistics*, 19(2):1641–1662, 2025. doi: 10.1214/24-AOAS1998.

Charles Audet and Warren Hare. *Derivative-Free and Blackbox Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, 2017. doi: 10.1007/978-3-319-68913-5.

Yavuz Faruk Bakman, Duygu Nur Yaldiz, Baturalp Buyukates, Chenyang Tao, Dimitrios Dimitriadis, and Salman Avestimehr. MARS: Meaning-aware response scoring for uncertainty estimation in generative LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7752–7767, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.419. URL https://aclanthology.org/2024.acl-long.419/.

Gagan Bansal, Tongshuang Wu, Joyce Zhou, Raymond Fok, Besmira Nushi, Ece Kamar, Marco Tulio Ribeiro, and Daniel S. Weld. Does the whole exceed its parts? The effect of AI explanations on complementary team performance. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI)*, pp. 1–16. ACM, 2021. doi: 10.1145/3411764.3445717.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-Bench: Evaluating conversational agents in a dual-control environment. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=OC2z7iSQKa.

Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael Jordan. Distribution-free, risk-controlling prediction sets. *Journal of the ACM*, 68(6):1–34, 2021. doi: 10.1145/3478535.

Jakub Binkowski, Denis Janiak, Albert Sawczyn, Bogdan Gabrys, and Tomasz Jan Kajdanowicz. Hallucination detection in LLMs using spectral features of attention maps. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 24354–24385, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1239. URL https://aclanthology.org/2025.emnlp-main.1239/.

Samuel R. Bowman, Jeeyoon Hyun, Ethan Perez, Edwin Chen, Craig Pettit, Scott Heiner, Kamilė Lukošiūtė, Amanda Askell, Andy Jones, Anna Chen, et al. Measuring progress on scalable oversight for large language models. *arXiv preprint arXiv:2211.03540*, 2022.

Mark Braverman, Xinyi Chen, Sham M. Kakade, Karthik Narasimhan, Cyril Zhang, and Yi Zhang. Calibration, entropy rates, and memory in language models. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1089–1099. PMLR, 2020. URL https://proceedings.mlr.press/v119/braverman20a.html.

Zana Buçinca, Maja Barbara Malaya, and Krzysztof Z. Gajos. To trust or to think: Cognitive forcing functions can reduce overreliance on AI in AI-assisted decision-making. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW1):1–21, 2021. doi: 10.1145/3449287.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, Ilya Sutskever, and Jeffrey Wu. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 4971–5012. PMLR, 2024. URL https://proceedings.mlr.press/v235/burns24b.html.

Margarida M. Campos, António Farinhas, Chrysoula Zerva, Mário A. T. Figueiredo, and André F. T. Martins. Conformal prediction for natural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 12:1497–1516, 2024. doi: 10.1162/tacl_a_00715. URL https://aclanthology.org/2024.tacl-1.82/.

Center for AI Safety, Scale AI, and HLE Contributors Consortium. A benchmark of expert-level academic questions to assess AI capabilities. *Nature*, 649(8099):1139–1146, 2026. doi: 10.1038/s41586-025-09962-4. URL https://www.nature.com/articles/s41586-025-09962-4. Humanity's Last Exam (HLE); preprint arXiv:2501.14249.

Mohammad-Amin Charusaie, Hussein Mozannar, David Sontag, and Samira Samadi. Sample efficient learning of predictors that complement humans. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2972–3005. PMLR, 2022. URL https://proceedings.mlr.press/v162/charusaie22a.html.

Veronica Chatrath, Bryan Zhu, George Pu, Jingxuan Fan, Apaar Shanker, Varun Ursekar, Anahita Sharma, Jason Qin, Keqi Han, Soham Dinesh Tiwari, Soham Dan, Vijay Kalmath, Yuan Li, Daniel Yue Zhang, Chenguang Wang, Zainab Doctor, Zhijun Yin, Nigam H. Shah, and Yuan Xue. CliniCARE-Bench: Clinical calibrated audit of medical reasoning in EHR. *arXiv preprint arXiv:2608.07796*, 2026. URL https://arxiv.org/abs/2608.07796.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. INSIDE: LLMs' internal states retain the power of hallucination detection. In *International Conference on Learning Representations (ICLR)*, pp. 3056–3076, 2024a. URL https://openreview.net/forum?id=Zj12nzlQbz.

Hui Chen, Miao Xiong, Yujie Lu, Wei Han, Ailin Deng, Yufei He, Jiaying Wu, Yibo Li, Yue Liu, and Bryan Hooi. MLR-Bench: Evaluating AI agents on open-ended machine learning research. In *Advances in Neural Information Processing Systems 38 (NeurIPS 2025), Datasets and Benchmarks Track*, pp. 131093–131141, 2025a. doi: 10.52202/085713-3945. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/ab8dd000d6f87f40061a73f8bca7fae4-Abstract-Datasets_and_Benchmarks_Track.html.

Kaiyuan Chen, Yixin Ren, Yang Liu, Xiaobo Hu, Haotong Tian, Tianbao Xie, et al. xbench: Tracking agents productivity scaling with profession-aligned real-world evaluations. arXiv preprint arXiv:2506.13651, 2025b. URL https://arxiv.org/abs/2506.13651.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024b. URL https://openreview.net/forum?id=cSimKw5p6R. Featured Certification.

Yangyi Chen, Lifan Yuan, Ganqu Cui, Zhiyuan Liu, and Heng Ji. A close look into the calibration of pre-trained language models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023*, pp. 1343–1367, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.75. URL https://aclanthology.org/2023.acl-long.75/.

C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. doi: 10.1109/TIT.1970.1054406.

Paul Christiano, Buck Shlegeris, and Dario Amodei. Supervising strong learners by amplifying weak experts. *arXiv preprint arXiv:1810.08575*, 2018.

Jasper Dekoninck, Maximilian Baader, and Martin Vechev. A unified approach to routing and cascading for LLMs. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 12987–13010. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/dekoninck25a.html.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Rühle, Laks V.S. Lakshmanan, and Ahmed Hassan Awadallah. Hybrid LLM: Cost-efficient and quality-aware query routing. In *International Conference on Learning Representations (ICLR)*, pp. 41348–41366, 2024. URL https://openreview.net/forum?id=02f3mUtqnM.

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H. Laradji, Manuel Del Verme, Tom Marty, David Vazquez, Nicolas Chapados, and Alexandre Lacoste. WorkArena: How capable are web agents at solving common knowledge work tasks? In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 11642–11662. PMLR, 2024. URL https://proceedings.mlr.press/v235/drouin24a.html.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Licheng Zhang, Xiaorui Wang, and Zhendong Mao. DeepResearch Bench: A comprehensive benchmark for deep research agents. In *The Fourteenth International Conference on Learning Representations (ICLR)*, pp. 42414–42448, 2026. URL https://openreview.net/forum?id=hQ0K2Hhq7H.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5050–5063, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.276. URL https://aclanthology.org/2024.acl-long.276/.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(53):1605–1641, 2010. URL https://jmlr.org/papers/v11/el-yaniv10a.html.

António Farinhas, Nuno M. Guerreiro, Sweta Agrawal, Ricardo Rei, and André F. T. Martins. Translate smart, not hard: Cascaded translation systems with quality-aware deferral. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 26742–26756, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1358. URL https://aclanthology.org/2025.emnlp-main.1358/.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024. doi: 10.1038/s41586-024-07421-0.

Vojtěch Franc, Daniel Průša, and Václav Voráček. Optimal strategies for reject option classifiers. *Journal of Machine Learning Research*, 24(11):1–49, 2023. URL https://jmlr.org/papers/v24/21-0048.html.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*, volume 30, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/4a8423d5e91fda00bb7e46540e2b0cf1-Abstract.html.

Gregor Geigle, Radu Timofte, and Goran Glavaš. Does object grounding really reduce hallucination of large vision-language models? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 2728–2742, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.159. URL https://aclanthology.org/2024.emnlp-main.159/.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6577–6595, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.366. URL https://aclanthology.org/2024.naacl-long.366/.

Ben Green. The flaws of policies requiring human oversight of government algorithms. *Computer Law & Security Review*, 45:105681, 2022. doi: 10.1016/j.clsr.2022.105681.

Yu Gui, Ying Jin, and Zhimei Ren. Conformal alignment: Knowing when to trust foundation models with guarantees. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, volume 37, pp. 73884–73919, 2024. doi: 10.52202/079017-2350. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/870ccde24673d3970a680bb48496ed63-Abstract-Conference.html.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330. PMLR, 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

Harbor Framework Team. Harbor: A framework for evaluating and optimizing agents and models in container environments, 2026. URL https://doi.org/10.5281/zenodo.20988961. Software, version v0.16.1.

Kilian Hendrickx, Lorenzo Perini, Dries Van der Plas, Wannes Meert, and Jesse Davis. Machine learning with a reject option: A survey. *Machine Learning*, 113(5):3073–3110, 2024. doi: 10.1007/s10994-024-06534-x.

Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. Decomposing uncertainty for large language models through input clarification ensembling. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 19023–19042. PMLR, 2024. URL https://proceedings.mlr.press/v235/hou24b.html.

Qitian Jason Hu, Jacob Bieker, Xiuyu Li, Nan Jiang, Benjamin Keigwin, Gaurav Ranganath, Kurt Keutzer, and Shriyash Kaustubh Upadhyay. RouterBench: A benchmark for multi-LLM routing system. arXiv preprint arXiv:2403.12031, 2024. URL https://arxiv.org/abs/2403.12031.

Yining Hua, Hongbin Na, Cyrus Ayubcha, and Levi Lian. Designing benchmarks for knowledge work. arXiv preprint arXiv:2605.23262, 2026. URL https://arxiv.org/abs/2605.23262.

Kung-Hsiang Huang, Akshara Prabhakar, Sidharth Dhawan, Yixin Mao, Huan Wang, Silvio Savarese, Caiming Xiong, Philippe Laban, and Chien-Sheng Wu. CRMArena: Understanding the capacity of LLM agents to perform professional CRM tasks in realistic environments. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2025*, pp. 3830–3850, Albuquerque, New Mexico, April 2025a. Association for Computational Linguistics. doi: 10.18653/v1/2025.naacl-long.194. URL https://aclanthology.org/2025.naacl-long.194/.

Kung-Hsiang Huang, Akshara Prabhakar, Onkar Thorat, Divyansh Agarwal, Prafulla Kumar Choubey, Yixin Mao, Silvio Savarese, Caiming Xiong, and Chien-Sheng Wu. CRMArena-Pro: Holistic assessment of LLM agents across diverse business scenarios and interactions. *Transactions on Machine Learning Research*, 2026. URL https://openreview.net/forum?id=EPlpe3Fx1x.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025b. doi: 10.1145/3703155.

Geoffrey Irving, Paul Christiano, and Dario Amodei. AI safety via debate. *arXiv preprint arXiv:1805.00899*, 2018.

Denis Janiak, Jakub Binkowski, Albert Sawczyn, Bogdan Gabrys, Ravid Shwartz-Ziv, and Tomasz Jan Kajdanowicz. The illusion of progress: Re-evaluating hallucination detection in LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 34728–34745, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1761. URL https://aclanthology.org/2025.emnlp-main.1761/.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023. doi: 10.1145/3571730.

Ziwei Ji, Lei Yu, Yeskendir Koishekenov, Yejin Bang, Anthony Hartshorn, Alan Schelten, Cheng Zhang, Pascale Fung, and Nicola Cancedda. Calibrating verbal uncertainty as a linear feature to reduce hallucinations. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3769–3793, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.187. URL https://aclanthology.org/2025.emnlp-main.187/.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? On the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021. doi: 10.1162/tacl_a_00407. URL https://aclanthology.org/2021.tacl-1.57/.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations (ICLR)*, pp. 54107–54157, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo A. Celi, and Roger G. Mark. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, 2023. doi: 10.1038/s41597-022-01899-x.

Jaehun Jung, Faeze Brahman, and Yejin Choi. Trust or escalate: LLM judges with provable guarantees for human agreement. In *International Conference on Learning Representations (ICLR)*, pp. 3101–3125, 2025. URL https://openreview.net/forum?id=UHPnqSTBPO.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know. arXiv preprint arXiv:2207.05221, 2022. URL https://arxiv.org/abs/2207.05221.

Sanyam Kapoor, Nate Gruver, Manley Roberts, Arka Pal, Samuel Dooley, Micah Goldblum, and Andrew Gordon Wilson. Calibration-tuning: Teaching large language models to know what they don't know. In Raúl Vázquez, Hande Celikkanat, Dennis Ulmer, Jörg Tiedemann, Swabha Swayamdipta, Wilker Aziz, Barbara Plank, Joris Baan, and Marie-Catherine de Marneffe (eds.), *Proceedings of the 1st Workshop on Uncertainty-Aware NLP (UncertaiNLP 2024)*, pp. 1–14, St Julians, Malta, March 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.uncertainlp-1.1. URL https://aclanthology.org/2024.uncertainlp-1.1/.

Sayash Kapoor, Benedikt Stroebl, Zachary S. Siegel, Nitya Nadgir, and Arvind Narayanan. AI agents that matter. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=Zy4uFzMviZ.

Tanzila Kehkashan, Muhammad Abdullah, Ahmad Sami Al-Shamayleh, Nikola Ivković, Nor Azman Ismail, Sharifah Sakinah Syed Ahmad, Abdul Rehman, and Adnan Akhunzada. From benchmarks to deployment: A comprehensive review of agentic AI evaluation. *Artificial Intelligence Review*, 59(8):167, 2026. doi: 10.1007/s10462-026-11571-0.

Polina Kirichenko, Mark Ibrahim, Kamalika Chaudhuri, and Samuel J. Bell. AbstentionBench: Reasoning LLMs fail on unanswerable questions. In Danielle Belgrave, Cheng Zhang, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, and Nancy Chen (eds.), *Advances in Neural Information Processing Systems 38 (NeurIPS 2025), Datasets and Benchmarks Track*, pp. 190800–190850, 2025. doi: 10.52202/085713-5729. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/fb122bfc3f0127a94ded048b5b03496f-Abstract-Datasets_and_Benchmarks_Track.html.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. VisualWebArena: Evaluating multimodal agents on realistic visual web tasks. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024*, pp. 881–905, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.50. URL https://aclanthology.org/2024.acl-long.50/.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

Bhawesh Kumar, Charlie Lu, Gauri Gupta, Anil Palepu, David Bellamy, Ramesh Raskar, and Andrew Beam. Conformal prediction with large language models for multi-choice question answering. arXiv preprint arXiv:2305.18404, 2023. URL https://arxiv.org/abs/2305.18404.

Vivian Lai, Chacha Chen, Q. Vera Liao, Alison Smith-Renner, and Chenhao Tan. Towards a science of human-AI decision making: A survey of empirical studies. *arXiv preprint arXiv:2112.11471*, 2021.

Hao Lang, Fei Huang, and Yongbin Li. Debate helps weak-to-strong generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 27410–27418, 2025. doi: 10.1609/aaai.v39i26.34952.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Measuring faithfulness in chain-of-thought reasoning. arXiv preprint arXiv:2307.13702, 2023. URL https://arxiv.org/abs/2307.13702.

Bracha Laufer-Goldshtein, Adam Fisch, Regina Barzilay, and Tommi Jaakkola. Efficiently controlling multiple risks with Pareto testing. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=cyg2YXn_BqF.

Fangyu Lei, Jixuan Chen, Yuxiao Ye, Ruisheng Cao, Dongchan Shin, Hongjin Su, Zhaoqing Suo, Hongcheng Gao, Wenjing Hu, Pengcheng Yin, Victor Zhong, Caiming Xiong, Ruoxi Sun, Qian Liu, Sida Wang, and Tao Yu. Spider 2.0: Evaluating language models on real-world enterprise text-to-SQL workflows. In *International Conference on Learning Representations (ICLR)*, pp. 28691–28735, 2025. URL https://openreview.net/forum?id=XmProj9cPs.

Ido Levy, Ben Wiesel, Sami Marreed, Alon Oved, Avi Yaeli, Nir Mashkif, and Segev Shlomov. ST-WebAgentBench: A benchmark for evaluating safety and trustworthiness in web agents. In *The Fourteenth International Conference on Learning Representations, ICLR 2026*, pp. 122213–122255. OpenReview.net, 2026. URL https://openreview.net/forum?id=MuCDzH0ctf.

Yukun Li, Sijia Wang, Lifu Huang, and Liping Liu. Graph-based confidence calibration for large language models. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=BDPvuD5FTg.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=8s8K2UZGTZ.

Gabrielle Kaili-May Liu, Gal Yona, Avi Caciularu, Idan Szpektor, Tim G. J. Rudner, and Arman Cohan. MetaFaith: Faithful natural language uncertainty expression in LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 29612–29656, Suzhou, China, November 2025a. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1505. URL https://aclanthology.org/2025.emnlp-main.1505/.

Peng Liu, Jiaxin Zhang, Shuaiqi Chen, and Shanguang Chen. Human-AI teaming in healthcare: $1 + 1 > 2$? *npj Artificial Intelligence*, 1(1):47, 2025b. doi: 10.1038/s44387-025-00052-4.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. AgentBench: Evaluating LLMs as agents. In *International Conference on Learning Representations (ICLR)*, pp. 52989–53046, 2024. URL https://openreview.net/forum?id=zAdUB0aCTQ.

Xiaoyuan Liu, Jianhong Tu, Yuqi Chen, Siyuan Xie, Sihan Ren, Tianneng Shi, Gal Gantar, Evan Sandoval, Donghyun Lee, Daniel Miao, et al. AgentBeats: Agentifying agent assessment for openness, standardization, and reproducibility. *arXiv preprint arXiv:2606.13608*, 2026. URL https://arxiv.org/abs/2606.13608.

Keming Lu, Hongyi Yuan, Runji Lin, Junyang Lin, Zheng Yuan, Chang Zhou, and Jingren Zhou. Routing to the expert: Efficient reward-guided ensemble of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1964–1974, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.109. URL https://aclanthology.org/2024.naacl-long.109/.

Mingyu Derek Ma, Yanna Ding, Zijie Huang, Jianxi Gao, Yizhou Sun, and Wei Wang. Inferring from logits: Exploring best practices for decoding-free generative candidate selection. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025*, pp. 33125–33144. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1589. URL https://aclanthology.org/2025.acl-long.1589/.

David Madras, Toniann Pitassi, and Richard Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/09d37c08f7b129e96277388757530c72-Abstract.html.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9004–9017, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.557. URL https://aclanthology.org/2023.emnlp-main.557/.

Mike A. Merrill, Alexander G. Shaw, Nicholas Carlini, Boxuan Li, Harsh Raj, Ivan Bercovich, Lin Shi, Jeong Yeon Shin, Thomas Walshe, E. Kelly Buchanan, et al. Terminal-Bench: Benchmarking agents on hard, realistic tasks in command line interfaces. In *The Fourteenth International Conference on Learning Representations (ICLR)*, pp. 40903–40986, 2026. URL https://openreview.net/forum?id=a7Qa4CcHak.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. GAIA: a benchmark for general AI assistants. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*, pp. 9025–9049. OpenReview.net, 2024. URL https://openreview.net/forum?id=fibxvahvs3.

Evan Miller. Adding error bars to evals: A statistical approach to language model evaluations. arXiv:2411.00640, 2024. URL https://arxiv.org/abs/2411.00640.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT*)*, pp. 220–229, 2019. doi: 10.1145/3287560.3287596.

Mahmoud Mohammadi, Yipeng Li, Jane Lo, and Wendy Yip. Evaluation and benchmarking of LLM agents: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25)*, pp. 6129–6139. ACM, 2025. doi: 10.1145/3711896.3736570.

Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7076–7087. PMLR, 2020. URL https://proceedings.mlr.press/v119/mozannar20b.html.

Hussein Mozannar, Hunter Lang, Dennis Wei, Prasanna Sattigeri, Subhro Das, and David Sontag. Who should predict? Exact algorithms for learning to defer to humans. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 206 of *Proceedings of Machine Learning Research*, pp. 10520–10545. PMLR, 2023. URL https://proceedings.mlr.press/v206/mozannar23a.html.

Mahdi Pakdaman Naeini, Gregory F. Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using Bayesian binning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, pp. 2901–2907, 2015. doi: 10.1609/aaai.v29i1.9602.

Ruijia Niu, Dongxia Wu, Rose Yu, and Yi-An Ma. Functional-level uncertainty quantification for calibrated fine-tuning on LLMs. arXiv preprint arXiv:2410.06431, 2024. URL https://arxiv.org/abs/2410.06431.

Jeremy Nixon, Michael W. Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. Measuring calibration in deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 38–41, 2019. URL https://openaccess.thecvf.com/content_CVPRW_2019/html/Uncertainty_and_Robustness_in_Deep_Visual_Learning/Nixon_Measuring_Calibration_in_Deep_Learning_CVPRW_2019_paper.html.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs from preference data. In *International Conference on Learning Representations*, volume 2025, pp. 34433–34448, 2025. URL https://openreview.net/forum?id=8sSqNntaMr.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*, pp. 66880–66913. OpenReview.net, 2025. URL https://openreview.net/forum?id=KRnsX5Em3W.

Tejal Patwardhan, Rachel Dias, Elizabeth Proehl, Grace Kim, Michele Wang, Olivia Watkins, Simón Posada Fishman, Marwan Aljubeh, Phoebe Thacker, Laurance Fauconnet, Natalie S. Kim, Patrick Chao, Samuel Miserendino, Gildas Chabot, David Li, Michael Sharman, Alexandra Barr, Amelia Glaese, and Jerry Tworek. GDPval: Evaluating AI model performance on real-world economically valuable tasks. In *The Fourteenth International Conference on Learning Representations (ICLR)*, pp. 24005–24040, 2026. URL https://openreview.net/forum?id=hcuEdq6eKD.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. Conformal language modeling. In *International Conference on Learning Representations (ICLR)*, pp. 11654–11681, 2024. URL https://openreview.net/forum?id=pzUhfQ74c5.

Inioluwa Deborah Raji, Andrew Smart, Rebecca N. White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron, and Parker Barnes. Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (FAT*)*, pp. 33–44, 2020. doi: 10.1145/3351095.3372873.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of the 7th Conference on Robot Learning (CoRL)*, volume 229 of *Proceedings of Machine Learning Research*, pp. 661–682. PMLR, 2023a. URL https://proceedings.mlr.press/v229/ren23a.html.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J. Liu. Out-of-distribution detection and selective generation for conditional language models. In *International Conference on Learning Representations (ICLR)*, 2023b. URL https://openreview.net/forum?id=kJUS5nD0vPB.

Jie Ren, Yao Zhao, Tu Vu, Peter J. Liu, and Balaji Lakshminarayanan. Self-evaluation improves selective generation in large language models. In Javier Antorán, Arno Blaas, Kelly Buchanan, Fan Feng, Vincent Fortuin, Sahra Ghalebikesabi, Andreas Kriegler, Ian Mason, David Rohde, Francisco J. R. Ruiz, Tobias Uelwer, Yubin Xie, and Rui Yang (eds.), *Proceedings on "I Can't Believe It's Not Better: Failure Modes in the Age of Foundation Models" at NeurIPS 2023 Workshops*, volume 239 of *Proceedings of Machine Learning Research*, pp. 49–64. PMLR, 2023c. URL https://proceedings.mlr.press/v239/ren23a.html.

Yuanzhe Shen, Yide Liu, Zisu Huang, Ruicheng Yin, Xiaoqing Zheng, and Xuanjing Huang. SATER: A self-aware and token-efficient approach to routing and cascading. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 10515–10529, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/

v1/2025.emnlp-main.531. URL https://aclanthology.org/2025.emnlp-main.531/.

Zhengyan Shi, Sander Land, Acyr Locatelli, Matthieu Geist, and Max Bartolo. Understanding likelihood over-optimisation in direct alignment algorithms. arXiv preprint arXiv:2410.11677, 2024. URL https://arxiv.org/abs/2410.11677.

Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z. Ren, and Anirudha Majumdar. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Computing Surveys*, 58(3):1–38, 2026. doi: 10.1145/3744238.

Vaishnavi Shrivastava, Ananya Kumar, and Percy Liang. Language models prefer what they know: Relative confidence estimation via confidence preferences. arXiv preprint arXiv:2502.01126, 2025. URL https://arxiv.org/abs/2502.01126.

Jake C. Snell, Thomas P. Zollo, Zhun Deng, Toniann Pitassi, and Richard Zemel. Quantile risk control: A flexible framework for bounding the probability of high-loss predictions. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=p6jsTidUkPx.

Elias Stengel-Eskin, Peter Hase, and Mohit Bansal. LACIE: Listener-aware finetuning for calibration in large language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, volume 37, pp. 43080–43106. Curran Associates, Inc., 2024. doi: 10.52202/079017-1364. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/4b8eaf3bcdc105423a972ed90eb07217-Abstract-Conference.html.

Olly Styles, Sam Miller, Patricio Cerda-Mardini, Tanaya Guha, Victor Sanchez, and Bertie Vidgen. WorkBench: a benchmark dataset for agents in a realistic workplace setting. In *First Conference on Language Modeling (COLM)*, 2024. URL https://openreview.net/forum?id=4HNAwZFDcH.

Jiayuan Su, Jing Luo, Hongwei Wang, and Lu Cheng. API is enough: Conformal prediction for large language models without logit-access. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 979–995, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.54. URL https://aclanthology.org/2024.findings-emnlp.54/.

Haochen Sun, Shuwen Zhang, Lujie Niu, Lei Ren, Hao Xu, Hao Fu, Fangkun Zhao, Caixia Yuan, and Xiaojie Wang. Collab-Overcooked: Benchmarking and evaluating large language models as collaborative agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 4922–4951, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.249. URL https://aclanthology.org/2025.emnlp-main.249/.

Yiyou Sun, Xinyang Han, Weichen Zhang, Yuanbo Pang, Tianyu Wang, Yuhan Cao, Yixiao Huang, Chris Duroiu, et al. Agents' Last Exam. arXiv preprint arXiv:2606.05405, 2026. URL https://arxiv.org/abs/2606.05405.

Jingyu Tang, Chaoran Chen, Jiawen Li, Zhiping Zhang, Bingcan Guo, Ibrahim Khalilov, Simret Araya Gebreegziabher, Bingsheng Yao, Dakuo Wang, Yanfang Ye, et al. Dark patterns meet GUI agents: LLM agent susceptibility to manipulative interfaces and the role of human oversight. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, pp. 1–26. ACM, 2026. doi: 10.1145/3772318.3791568.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. Confidence improves self-consistency in LLMs. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20090–20111, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.1030. URL https://aclanthology.org/2025.findings-acl.1030/.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5433–5442, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.330. URL https://aclanthology.org/2023.emnlp-main.330/.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. AppWorld: A controllable world of apps and people for benchmarking interactive coding agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16022–16076, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.850. URL https://aclanthology.org/2024.acl-long.850/.

Andreas Tsamados, Luciano Floridi, and Mariarosaria Taddeo. Human control of AI systems: From supervision to teaming. *AI and Ethics*, 5(2):1535–1548, 2025. doi: 10.1007/s43681-024-00489-4.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*, volume 36, pp. 74952–74965, 2023. doi: 10.52202/075280-3275. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/ed3fea9033a80fea1376299fa7863f4a-Abstract-Conference.html.

UK AI Security Institute. Inspect AI: Framework for large language model evaluations, May 2024. URL https://github.com/UKGovernmentBEIS/inspect_ai. Released May 2024.

Michelle Vaccaro, Abdullah Almaatouq, and Thomas W. Malone. When combinations of humans and AI are useful: A systematic review and meta-analysis. *Nature Human Behaviour*, 8(12):2293–2303, 2024. doi: 10.1038/s41562-024-02024-1.

Clovis Varangot-Reille, Christophe Bouvard, Mathieu Ciancone, Antoine Gourru, Marion Schaeffer, and François Jacquenet. Doing more with less: A survey on routing strategies for resource optimisation in large language model-based systems. *Journal of Artificial Intelligence Research*, 86, 2026. doi: 10.1613/jair.1.19801.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Boda Sadallah, Kirill Grishchenkov, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, Maxim Panov, and Artem Shelmanov. Benchmarking uncertainty quantification methods for large language models with LM-Polygraph. *Transactions of the Association for Computational Linguistics*, 13:220–248, 2025. doi: 10.1162/tacl_a_00737. URL https://aclanthology.org/2025.tacl-1.11/.

Rajeev Verma and Eric Nalisnick. Calibrated learning to defer with one-vs-all classifiers. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 22184–22202. PMLR, 2022. URL https://proceedings.mlr.press/v162/verma22c.html.

Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pp. I–511–I–518, 2001. doi: 10.1109/CVPR.2001.990517.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 2024. doi: 10.1093/jrsssb/qkad009.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. BrowseComp: A simple yet challenging benchmark for browsing agents. arXiv preprint arXiv:2504.12516, 2025. URL https://arxiv.org/abs/2504.12516.

Laura Weidinger, Maribeth Rauh, Nahema Marchal, Arianna Manzini, Lisa Anne Hendricks, Juan Mateos-Garcia, Stevie Bergman, Jackie Kay, Conor Griffin, Ben Bariach, Iason Gabriel, Verena Rieser, and William Isaac. Sociotechnical safety evaluation of generative AI systems. *arXiv preprint arXiv:2310.11986*, 2023.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556, 2025. doi: 10.1162/tacl_a_00754. URL https://aclanthology.org/2025.tacl-1.26/.

Johnathan Xie, Annie S. Chen, Yoonho Lee, Eric Mitchell, and Chelsea Finn. Calibrating language models with adaptive temperature scaling. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 18128–18138, Miami, Florida, USA, November 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1007. URL https://aclanthology.org/2024.emnlp-main.1007/.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In Amir Globerson, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37 (NeurIPS 2024), Datasets and Benchmarks Track*, volume 37, pp. 52040–52094, 2024b. doi: 10.52202/079017-1650. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/5d413e48f84dc61244b6be550f1cd8f5-Abstract-Datasets_and_Benchmarks_Track.html.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can LLMs express their uncertainty? An empirical evaluation of confidence elicitation in LLMs. In *International Conference on Learning Representations (ICLR)*, pp. 23650–23678, 2024. URL https://openreview.net/forum?id=gjeQKFxFpZ.

Frank F. Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Z. Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, Mingyang Yang, Hao Yang Lu, Amaad Martin, Zhe Su, Leander Maben, Raj Mehta, Wayne Chi, Lawrence Jang, Yiqing Xie, Shuyan Zhou, and Graham Neubig. TheAgentCompany: Benchmarking LLM agents on consequential real world tasks. In *Advances in Neural Information Processing Systems 38 (NeurIPS 2025), Datasets and Benchmarks Track*, pp. 10291–10321, 2025. doi: 10.52202/085713-0315. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/0d744742f6fac4d1134c019b7cef3c8a-Abstract-Datasets_and_Benchmarks_Track.html.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *The Thirteenth International Conference on Learning Representations (ICLR)*, pp. 9965–10017, 2025. URL https://openreview.net/forum?id=roNSXZpUDN.

Asaf Yehudai, Lilach Eden, Alan Li, Guy Uziel, Yilun Zhao, Roy Bar-Haim, Arman Cohan, and Michal Shmueli-Scheuer. A survey on evaluation of LLM-based agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 26690–26714, San Diego, California, United States, 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.1330. URL https://aclanthology.org/2026.findings-acl.1330/.

Murong Yue, Jie Zhao, Min Zhang, Liang Du, and Ziyu Yao. Large language model cascades with mixture of thought representations for cost-efficient reasoning. In *International Conference on Learning Representations (ICLR)*, pp. 21691–21728, 2024. URL https://openreview.net/forum?id=6okaSfANzh.

Jieyu Zhang, Ranjay Krishna, Ahmed H. Awadallah, and Chi Wang. EcoAssistant: Using LLM assistant more affordably and accurately. arXiv preprint arXiv:2310.03046, 2023. URL https://arxiv.org/abs/2310.03046.

Zheng Zhang, Cuong C. Nguyen, Kevin Wells, Thanh-Toan Do, David Rosewarne, and Gustavo Carneiro. Coverage-constrained human-AI cooperation with multiple experts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18055–18062, 2026. doi: 10.1609/aaai.v40i21.38866.

Zhexin Zhang, Shiyao Cui, Yida Lu, Jingzhuo Zhou, Junxiao Yang, Hongning Wang, and Minlie Huang. Agent-SafetyBench: Evaluating the safety of LLM agents. arXiv preprint arXiv:2412.14470, 2024. URL https://arxiv.org/abs/2412.14470.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. SLiC-HF: Sequence likelihood calibration with human feedback. arXiv preprint arXiv:2305.10425, 2023a. URL https://arxiv.org/abs/2305.10425.

Yao Zhao, Misha Khalman, Rishabh Joshi, Shashi Narayan, Mohammad Saleh, and Peter J. Liu. Calibrating sequence likelihood improves conditional language generation. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023b. URL https://openreview.net/forum?id=0qSOodKmJaN.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*, pp. 15585–15606. OpenReview.net, 2024. URL https://openreview.net/forum?id=oKn9c6ytLx.

Liming Zhu, Qinghua Lu, Ming Ding, Sung Une Lee, and Chen Wang. Designing meaningful human oversight in AI. *AI and Ethics*, 6(3):286, 2026. doi: 10.1007/s43681-026-01147-7.

Thomas P. Zollo, Todd Morrill, Zhun Deng, Jake C. Snell, Toniann Pitassi, and Richard Zemel. Prompt risk control: A rigorous framework for responsible deployment of large language models. In *International Conference on Learning Representations (ICLR)*, pp. 4045–4067, 2024. URL https://openreview.net/forum?id=5tGGWOijvq.

A Full Deployment-Qualification Methodology

Section 3 formulated reliable deployment as a constrained optimization problem over oversight policies. We now develop the quantities, policy-optimization procedures, and statistical qualification methods needed to operationalize that formulation.

The key distinction is between *agent performance* and *deployed-system reliability*. An oversight policy determines how autonomous agent execution is combined with human intervention. Its value depends not only on how often the agent succeeds on its own, but also on how effectively the policy uses observable workflow information to introduce oversight, how effective that oversight is, and what it costs.

We first develop the trajectory-invariant terminal accept-or-escalate setting, where many candidate policies can be evaluated from the same saved execution evidence, and describe statistical certification of the candidate family on held-out cases. We then formalize two extensions not empirically instantiated in this paper: risk-sensitive qualification and trajectory-dependent oversight, where intervention changes the execution trajectory itself.

A.1 Qualification Quantities

For agent m, workflow w, and oversight policy π, let $u^\pi(x) \in \{0, 1\}$ denote the workflow-specific success outcome on instance x defined in Section 3.1. For a sampled instance x_i, write $u_i^\pi := u^\pi(x_i)$.

We define the reliability of the human–AI system as

$$R_{m,w}(\pi) = \mathbb{E}_{x\sim\mathcal{D}_w}\left[u^\pi(x)\right], \tag{9}$$

The expectation is over workflow instances x drawn from the population represented by $\mathcal{D}_w$ and, when execution is stochastic, over the induced randomness in agent execution, the environment, oversight process, and interacting actors. Since $u^\pi(x)$ is binary, $R_{m,w}(\pi)$ is the probability that an execution under policy π satisfies the workflow-specific success criterion on a randomly drawn workflow instance. Deployment qualification is an aggregate statistical claim about the human–AI system induced by π, not a property assigned to any individual task instance.

Let $K_w(Z^\pi(x))$ denote the workflow-specific operating cost associated with an execution on instance x under policy π. This may include agent execution as well as human review and intervention. The expected operating cost is

$$C_{m,w}(\pi) = \mathbb{E}_{x\sim\mathcal{D}_w}[K_w(Z^\pi(x))]\,. \tag{10}$$

Conceptually, this cost can be decomposed into agent and human components, where both terms may depend on π, as intervention may increase human effort while changing the subsequent agent execution.

$$C_{m,w}(\pi) = C_{\text{agent}}(m, w, \pi) + C_{\text{human}}(m, w, \pi), \tag{11}$$

Equations 9 and 10 define the two canonical quantities in the deployment objective of Equation 2. They are properties of the deployed human–AI configuration under a specified policy, not intrinsic properties of the agent alone.

A.2 Terminal Accept-or-Escalate Oversight

We first consider the trajectory-invariant terminal accept-or-escalate setting used in our CliniCARE-Bench and τ^2 retail studies. In this setting, the agent completes its work before an oversight decision is made. The completed execution provides a result together with an observable routing signal s_i; the policy then determines whether the result is accepted autonomously or routed to a declared human-review path.

To evaluate such policies, READY first executes the agent on representative workflow instances under autonomous execution. For each task instance x_i, this produces a saved trajectory Z_i, an observable routing signal s_i, and an autonomous-success indicator $u_i \in \{0, 1\}$.

Here, (Z_i, s_i, u_i) are evaluation data used for policy qualification; at deployment time, the routing decision is made from observable information such as s_i, while u_i is generally unknown.

The terminal routing decision occurs only after Z_i has been generated, so the trajectory is invariant to the candidate routing policy. READY can evaluate many accept-or-escalate policies by replaying the same saved evaluation runs instead of re-executing the agent for every candidate policy.

For a scalar routing signal s_i, consider a threshold policy π_τ that accepts the completed result autonomously when $s_i \geq \tau$ and routes it to human review otherwise. The policy is intended for signals whose larger values preferentially rank successful autonomous executions above failures. This ordering is a diagnostic and a source of power, not a requirement for validity of the qualification test; its relationship to fixed-sequence optimality is discussed in Appendix E.1. The autonomous coverage under threshold τ is

$$c(\tau) = \Pr(s \geq \tau), \tag{12}$$

and its corresponding human-review rate is

$$e(\tau) = 1 - c(\tau). \tag{13}$$

Coverage is useful in this terminal setting because it is simply the complement of review burden; it is not required as a universal READY deployment metric. Among autonomously accepted cases, define

$$a(\tau) = \Pr(u = 1 \mid s \geq \tau), \tag{14}$$

the success probability of work accepted without review.

Let $a_h(\tau)$ denote the probability that a case routed by policy π_τ is successfully resolved by the declared human-review path. The reliability of the deployed human–AI system is then

$$R(\tau) = c(\tau)a(\tau) + \big(1 - c(\tau)\big)a_h(\tau). \tag{15}$$

The first term corresponds to work accepted autonomously; the second corresponds to work routed to human review. At zero coverage, the accepted-work contribution is defined as zero; at full coverage, the review contribution is zero. These conventions avoid conditioning on an empty subset.

The review-path performance $a_h(\tau)$ can be either measured from the review process or introduced as a deployment assumption. It is important to note that because routing deliberately selects a non-random subset of cases, $a_h(\tau)$ may vary with the policy: cases routed under a more selective policy may be systematically more difficult than cases routed under another policy. When a constant review-success assumption a_h is available, Equation 15 reduces to

$$R(\tau) = c(\tau)a(\tau) + \big(1 - c(\tau)\big)a_h. \tag{16}$$

This formulation makes clear why autonomous benchmark performance does not determine deployment reliability. Two agents with similar autonomous success rates can support substantially different human–AI operating points because the oversight policies available to them may differ in how effectively they identify cases requiring intervention.

A.3 Cost at a Target Reliability

In terminal accept-or-escalate deployment, the agent has already executed before the routing decision is made. Let k_m denote its expected execution cost per case and let k_h denote the incremental cost of the declared human-review path. Under a constant-cost approximation,

$$C(\tau) = k_m + k_h\, e(\tau) = k_m + k_h\big(1 - c(\tau)\big). \tag{17}$$

For target reliability Y, the population-level policy optimization problem is

$$\tau^\star(Y) = \arg\min_{\tau} C(\tau) \qquad \text{subject to} \qquad R(\tau) \geq Y, \tag{18}$$

together with any additional workflow-specific risk constraint.

Equation 18 defines the ideal population operating point. In practice, $R(\tau)$ and $C(\tau)$ are unknown, and READY solves the problem statistically: the candidate thresholds are tested against the target Y on the held-out qualification set in a prespecified order, and the certified operating point is the lowest-cost threshold in the certified set (Appendix A.5). READY does not require a particular candidate family or test order. What is required for qualification is that both be fixed before the qualification cases are examined; development evidence may inform that choice and supports the routing diagnostics of Appendix A.4.

When k_m and $k_h > 0$ are constant, minimizing operating cost is equivalent to minimizing human-review burden, or equivalently maximizing autonomous coverage, among feasible policies. Writing

$$e^\star(Y) = e(\tau^\star(Y))\,, \tag{19}$$

the population cost of reliable deployment is

$$C^\star(Y) = k_m + k_h e^\star(Y). \tag{20}$$

This provides an economically meaningful basis for comparing systems: how much operating cost is required for each system to satisfy the same workflow-specific reliability requirement. Across systems, a comparison of total operating cost must also account for differences in agent execution cost k_m; Section 5.3 makes this comparison with recorded agent costs and a swept review price.

The constant-cost model is useful for exposition but is not required by READY. Review effort may vary substantially across cases, and different policies may trigger different forms of review, correction, or escalation. When case-level costs are available, READY estimates $C(\pi)$ directly from the costs induced by the candidate policy instead of assigning a single constant k_h to every reviewed case.

A.4 Quality of the Routing Signal

For terminal accept-or-escalate policies, the value of oversight depends in part on whether the routing signal identifies cases for which autonomous execution is likely to fail. A useful signal should rank higher-risk cases below lower-risk cases, allowing a more selective policy to remove failures from autonomous handling preferentially.

A natural diagnostic is the selective-risk curve,

$$r(\tau) = \Pr(u = 0 \mid s \geq \tau), \tag{21}$$

or equivalently $r(c)$ when operating points are indexed by autonomous coverage. Sweeping the routing threshold traces the risk–coverage relationship: a stronger routing signal achieves lower selective risk at the same autonomous coverage.

We summarize this curve using the area under the risk–coverage curve (**AURC**),

$$\mathrm{AURC} = \int_0^1 r(c)\, dc, \tag{22}$$

with lower values indicating a more favorable risk–coverage tradeoff.

AURC is influenced both by routing quality and by the base autonomous success rate $a_0 = \Pr(u = 1)$. We also report the **AUROC** of the routing signal s for predicting execution success u. AUROC

measures ranking discrimination and, unlike AURC, is not directly determined by the prevalence of successful executions.

These metrics are diagnostics for the terminal routing policy class rather than universal READY qualification quantities. Their practical importance is ultimately determined by the operating policies they support: at a fixed reliability target, better routing can reduce the amount of human intervention, and the operating cost, required for qualification.

A.5 Statistical Qualification

Optimizing an oversight policy on the evidence that also reports its reliability does not establish that the policy will satisfy its deployment requirements on future cases. READY therefore casts qualification as a multiple-testing problem in the Learn-then-Test framework of Angelopoulos et al. (2025), which generalizes risk-controlling prediction sets (Bates et al., 2021): every candidate policy is tested against the reliability target on held-out evidence, and the certified set inherits a finite-sample guarantee from the multiple-testing procedure.

Partition and candidate family. The evaluation sample S_w is partitioned by disjoint index sets $I_{\text{dev}}, I_{\text{qual}} \subseteq [N_w]$ into a development set $S_{\text{dev}} = \{x_i : i \in I_{\text{dev}}\}$ and a held-out qualification set $S_{\text{qual}} = \{x_i : i \in I_{\text{qual}}\}$. The development set supports the routing diagnostics of Appendix A.4 and any data-driven choice of the candidate family or of the order in which candidates are tested; the qualification set is used only for certification, and the family and order are fixed before it is examined. In our empirical studies the family is $\{\pi_\tau : \tau \in \mathcal{G}\} \cup \{\pi_\infty\}$ for the prespecified percentage-point grid $\mathcal{G} = \{0, 0.01, \ldots, 1\}$, where π_∞ routes every case to review. Adjacent grid thresholds that induce identical routing on a sample are computationally collapsed without changing the family or its order. For each qualification case i and candidate τ define

$$d_i(\tau) = \mathbf{1}\{s_i \geq \tau\}, \tag{23}$$

where $d_i = 1$ denotes autonomous acceptance and $d_i = 0$ denotes human review.

Measured human-review outcomes. When the outcome of the declared human-review path is observed for each escalated case, let $h_i \in \{0, 1\}$ denote whether that review path produces a successful result. The realized success of the deployed system on qualification case i under π_τ is

$$v_i = d_i u_i + (1 - d_i) h_i, \tag{24}$$

and the empirical deployment reliability is

$$\widehat{R}(\pi_\tau) = \frac{1}{N_{\text{qual}}} \sum_{i \in I_{\text{qual}}} v_i, \qquad N_{\text{qual}} = |I_{\text{qual}}|. \tag{25}$$

In the terminal setting, v_i is the realized policy-level success outcome corresponding to the general success event u_i^π in Appendix A.1; u_i here denotes the success of the autonomous execution before routing. At confidence level $1 - \alpha$, READY certifies a policy only if the one-sided lower confidence bound implied by the test below clears the target,

$$R_{\text{LCB}}^{1-\alpha}(\pi) \geq Y. \tag{26}$$

A point estimate above the target is not sufficient: the held-out evidence must support a reliability lower bound that itself clears the deployment requirement.

Human-review performance as a deployment assumption. In retrospective benchmark analysis, the declared human-review path may not be executed for every escalated case. Its effectiveness can instead be introduced explicitly as a deployment assumption. Suppose, for example, that every routed case is assumed to be successfully resolved with probability a_h. For each qualification case define the policy-level conditional success value

$$y_i(a_h) = d_i u_i + (1 - d_i) a_h \in [0, 1]. \tag{27}$$

Under the constant-review assumption, the population reliability of the policy is $R(\pi_\tau; a_h) = \mathbb{E}[y_i(a_h)]$. Writing n_{acc} for the number of accepted qualification cases and $\widehat{p}_{\text{acc}}$ for the observed success rate among them, the empirical estimate is

$$\widehat{R}(\pi_\tau; a_h) = \frac{n_{\text{acc}}}{N_{\text{qual}}} \widehat{p}_{\text{acc}} + \frac{N_{\text{qual}} - n_{\text{acc}}}{N_{\text{qual}}} a_h, \tag{28}$$

where both the accepted-case success rate and the accepted fraction are estimated from the qualification sample. For $n_{\text{acc}} = 0$, $\widehat{p}_{\text{acc}}$ is undefined, but its weighted contribution is zero; equivalently, that contribution is $k_{\text{acc}}/N_{\text{qual}}$, where k_{acc} counts accepted successes. Under full review every $y_i(a_h) = a_h$, so the reliability of π_∞ is exactly a_h by assumption, with no sampling uncertainty, and π_∞ meets any target $Y \le a_h$ by assumption, so it enters the certified set without a test (at $Y = a_h$ every loss equals ε, and no test could reject).

Testing one candidate. Fix a candidate τ and define the bounded loss $L_i(\tau) = 1 - y_i(a_h) \in [0, 1]$ on each qualification unit, so that $R(\pi_\tau; a_h) \ge Y$ is equivalent to $\mathbb{E}[L_i(\tau)] \le \varepsilon$ with $\varepsilon = 1 - Y$. The null hypothesis to be rejected is $H_\tau : \mathbb{E}[L_i(\tau)] > \varepsilon$. Any p-value that is valid for the mean of independent $[0, 1]$-valued observations may be used. The Hoeffding–Bentkus p-value of Bates et al. (2021),

$$p_\tau^{\text{HB}} = \min\Big\{\exp\big(-n\, h_1(\min\{\bar{L}_\tau, \varepsilon\}, \varepsilon)\big),\ e\ \Pr\big[\text{Bin}(n, \varepsilon) \le \lceil n\bar{L}_\tau \rceil\big]\Big\}, \tag{29}$$

with $h_1(a, b) = a \log \frac{a}{b} + (1 - a) \log \frac{1-a}{1-b}$, depends only on the sample mean $\bar{L}_\tau$. We use instead the betting p-value of Waudby-Smith & Ramdas (2024), which adapts to the variance of the loss and is therefore much tighter here, where every reviewed unit contributes the constant $1 - a_h$: for a predictable sequence $0 \le \lambda_t \le c/(1 - \varepsilon)$, the capital process

$$K_n(\varepsilon) = \prod_{t=1}^{n} \big(1 + \lambda_t(\varepsilon - L_t)\big) \tag{30}$$

satisfies $\mathbb{E}[K_n(\varepsilon)] \le 1$ under H_τ, so $p_\tau^{\text{bet}} = \min\{1, 1/K_n(\varepsilon)\}$ is a valid p-value by Markov's inequality. We take $c = 3/4$ and the variance-adaptive bet $\lambda_t = \min\{\sqrt{2\log(1/\alpha)/(n\hat{\sigma}_{t-1}^2)},\ c/(1 - \varepsilon)\}$, where $\hat{\sigma}_{t-1}^2$ is the running variance of $L_1, \ldots, L_{t-1}$ with a prior variance of $1/4$; because the ordering of an exchangeable sample is arbitrary, we average K_n over 64 fixed permutations of the qualification units, which preserves $\mathbb{E}[K_n] \le 1$. The one-sided lower confidence bound reported for a candidate is the inversion of this test,

$$R_{\text{LCB}}^{1-\alpha}(\pi_\tau; a_h) = \sup\{\, Y' : p_\tau(1 - Y') \le \alpha \,\}, \tag{31}$$

so that $R_{\text{LCB}} \ge Y$ if and only if H_τ is rejected at level α.

Testing the family. Rejecting H_τ for many candidates and reporting the cheapest would inflate the error probability, so the candidates are tested by fixed-sequence testing in a prespecified order: π_∞ first, then π_τ for decreasing τ, that is, from review-everything toward accept-everything. Hypotheses are rejected sequentially while $p_\tau \le \alpha$, and testing stops at the first non-rejection. The induced routing on the sample changes only at observed values of s_i, so testing the tie-group boundaries of the stated confidences is exhaustive. The certified set $\widehat{\mathcal{T}}_\alpha(Y)$ is the set of rejected candidates, and the certified operating point is its lowest-cost member (Equation 5). Fixed-sequence testing controls the family-wise error rate at α without any adjustment for the number of candidates (Angelopoulos et al., 2025): with probability at least $1 - \alpha$, no certified threshold has $R(\pi_\tau; a_h) < Y$, and in particular the reported operating point meets the target. The guarantee is distribution-free and holds in finite samples; it requires that the qualification units satisfy the independence assumptions of the bounded-mean test and that the family and order be fixed before S_{qual} is examined. The guarantee does not imply that an untested candidate beyond the first non-rejection is infeasible. Thus the selected point is lowest-cost within the certified prefix. It coincides with the globally cheapest certifiable threshold when feasibility is nested in the prespecified order; Appendix E.1 states sufficient conditions and the interpretation when that property is not assumed.

Conditional certificates. The declared value a_h is held fixed within the test: the p-value propagates sampling variation in autonomous outcomes and in routing burden, but does not validate the review-success assumption itself. The qualification statement is explicitly conditional, of the form "certified at target Y under the declared review assumption a_h." If a_h is itself estimated from separate human-review data, uncertainty in that estimate must also be propagated rather than treating a_h as known exactly. One conservative approach substitutes a lower confidence bound $a_{h,\mathrm{LCB}}$ for a_h in the loss, with error probability allocated across the review estimate and the certificate to control their joint uncertainty.

Sampling unit. The statistical sampling unit must match the unit over which the deployment claim is intended to generalize, as in clustered error analysis of model evaluations (Miller, 2024). Multiple stochastic executions of the same task instance are not treated as independent new workflow cases. When repeated runs are present, the loss of a task is the mean of $1 - y_i(a_h)$ over its runs, which remains in $[0, 1]$, and the test is applied to the task-level losses. Similarly, when cases are naturally clustered by patient, customer, account, site, or another higher-level unit, qualification must preserve that dependence structure by forming the loss at the cluster level.

A policy that fails the statistical criterion is reported as *not certified at target* Y; the certified set is not enlarged by re-examining qualification outcomes, and any change to the candidate family or test order after examining them requires a new held-out qualification evaluation.

Unless otherwise stated, READY certificates are *pointwise* for a pre-specified deployment target and system. Simultaneous claims across multiple targets or systems require an additional multiplicity correction, for example a Bonferroni allocation of α across the reporting grid, at the price of more conservative certified sets.

A.6 Split-Resampling Validity Check

The certificate of Section A.5 is a finite-sample statement about one held-out partition. To check that it behaves as claimed when the partition itself is redrawn, we resample the split $1{,}000$ times per system from the full cohort (cases for CliniCARE, tasks for τ^2 retail, so that clustering is respected), choose a threshold on one half by three rules, and evaluate the chosen threshold on the other half at $Y = 0.75$, $a_h = 0.90$. The rules are: a *naive* pick, the cheapest threshold whose point-estimate reliability on the selection half is at least Y; a *margin* pick, the same with target $Y + \delta$ for a fixed safety margin ($\delta = 0.05$ CliniCARE, 0.11 retail) but no lower-bound test; and the *certified* pick, the cheapest threshold passed by the fixed-sequence betting test at $\alpha = 0.05$. A draw counts as a violation when the held-out half's realized reliability falls below Y. No new model runs are involved.

Figure 5 shows the outcome. Pooled over systems, the naive pick violates the target on the held-out half in 31.1% of CliniCARE draws and 20.1% of retail draws, and against full-cohort reliability in 16.3% and 11.3%; for individual systems the held-out violation rate reaches 53% (GPT-5.6-Sol) and 50% (GPT-5.6-Luna, retail). The certified pick violates the held-out half in 0.9% and 1.5% of draws, never exceeds 3.6% for any system, and never falls below Y on the full cohort, so the $\alpha = 0.05$ guarantee holds with room to spare. Held-out violation is a conservative check, since the evaluation half is itself a noisy estimate of the population, which is why even a valid procedure shows a non-zero rate. The price of validity is review: the certified pick averages 36.9% review on CliniCARE against 20.2% for the naive pick, and 29.7% against 2.7% on retail. The margin pick is about as valid as certification on CliniCARE (1.4%) at similar review (36.2% versus 36.9%) but costs more review on retail (38.2% versus 29.7%), where 57 tasks make the margin coarse. Certification thus matches an ad-hoc margin where the margin happens to fit, costs less review where it does not, and removes the need to choose δ at all.

A.7 Formal Extension: Risk-Sensitive Qualification

This subsection specifies an extension that is not instantiated in our experiments. Reliability treats every unsuccessful execution as the same binary event. For workflows in which failures differ materially in consequence, READY may additionally define a workflow-specific deployment loss

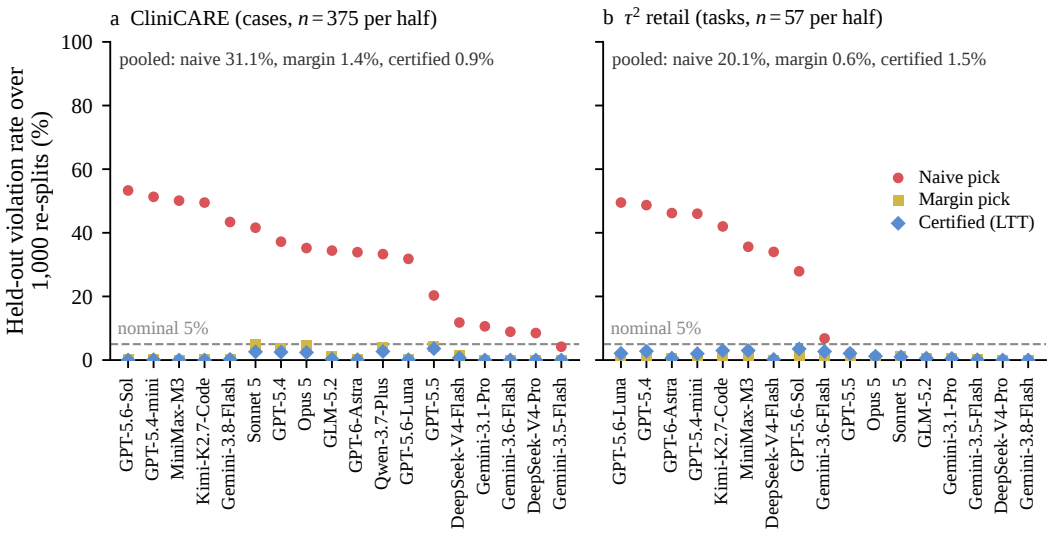


Figure 5: **The certified threshold holds its target on redrawn splits; a naive pick does not.** Held-out violation rate $P(\widehat{R}_B < Y)$ over 1,000 random re-splits of the full cohort, per system and rule, at $Y = 0.75$, $a_h = 0.90$. The threshold is chosen on half A and evaluated on half B; systems are ordered by the naive rate. Dashed line: nominal $\alpha = 5\%$. Pooled rates are printed in each panel; review burdens are given in the text.

$$L_w(Z^\pi) \geq 0 \tag{32}$$

and impose the constraint

$$\rho_w(L_w(Z^\pi)) \leq B_w, \tag{33}$$

where B_w is the declared risk tolerance and ρ_w is a workflow-specific risk functional. Examples include a bound on the probability of a specified severe event or a tail-risk measure such as Conditional Value at Risk (CVaR).

This constraint is optional. The common READY objective remains reliable deployment at minimum operating cost; an additional risk constraint is introduced when binary success alone does not adequately represent the consequences of failure. Estimation and statistical qualification of risk-sensitive constraints are discussed further in Appendix E.2.

A.8 Formal Extension: Trajectory-Dependent Oversight

This subsection likewise specifies an extension beyond the empirical scope of this paper. The terminal formulation above is especially convenient because the routing decision does not change the saved agent trajectory. More general enterprise workflows allow intervention during execution.

Let h_t denote the observable workflow history at decision point t, and let

$$\pi(o_t \mid h_t) \tag{34}$$

denote an oversight policy over intervention decisions o_t, such as approval, correction or takeover. Executing the workflow under π induces a policy-dependent trajectory

$$Z_i^\pi \sim P(\cdot \mid m, x_i, E_w, \pi). \tag{35}$$

The qualification objective remains the one defined in Equation 2. The difference is how the evidence needed to estimate that objective is generated.

For trajectory-invariant terminal policies, candidate policies can be evaluated by replaying saved trajectories under different routing rules. For a trajectory-dependent policy, changing π changes what the agent, oversight actor, or interacting user subsequently observes and does. Reliability, cost, and risk must instead be estimated by executing or appropriately simulating the candidate policy in the evaluation runtime.

The phases introduced in Section 3 should therefore be understood as a logical evaluation flow rather than necessarily a one-pass computation. For trajectory-dependent oversight, policy optimization may repeatedly invoke agent execution and workflow evaluation while searching over candidate policies.

Policy-in-the-loop evaluation may change both sides of the deployment tradeoff. Intervention can improve the probability of successful completion, but can also change agent execution length, human effort, recovery behavior, and the distribution of subsequent outcomes. These effects are captured by the same general quantities $R_{m,w}(\pi)$, $C_{m,w}(\pi)$, and, when applicable, $\rho_w(L_w(Z^\pi))$.

This setting is naturally connected to simulation-based and black-box optimization, where objective and constraint values need not admit closed-form expressions and can instead be estimated through stochastic simulation or system execution (Amaran et al., 2016; Audet & Hare, 2017). READY does not prescribe an optimization algorithm. The policy representation and optimization method may be workflow-dependent. What is shared across workflows is the deployment objective and qualification procedure: candidate policies are evaluated in terms of the reliability, cost, and risk of the resulting human–AI system, candidate policies are certified on held-out evidence, and the lowest-cost certified policy is reported.

Empirical scope. Section 4 applies the core qualification methodology to CliniCARE-Bench and τ^2 retail in the trajectory-invariant, terminal accept-or-escalate setting with mean binary reliability. It does not instantiate the risk-sensitive or trajectory-dependent extensions above. The CliniCARE analysis is conducted on the benchmark's full four-way adjudication space using a signal-based accept-or-escalate policy that is agnostic to the class labels, with sensitivity analyses in Appendix C and a class-conditioned routing control in Appendix C.8. The τ^2 retail analysis uses binary benchmark reward and task-clustered qualification, as specified in Appendix D.1.

B Evaluation Runtime and Open Testbed

This appendix expands the runtime and workflow interface summarized in Section 3.1.

READY does not introduce a new general-purpose agent-evaluation harness; it adds the workflow-level semantics and deployment-qualification layer that turn executions into a reliability–oversight–cost deployment profile. The reference implementation follows the three-layer architecture of Figure 1: non-executing *specification artifacts* define the workflow, evaluation semantics, and deployment assumptions; an *evaluation runtime* instantiates them and executes the agent, environment, and any oversight or user actors; and the resulting evidence is passed to the qualification methodology of Section 3.

Three logically distinct specifications cross these boundaries so that the same evidence can be reused under different deployment requirements. The *workflow specification* identifies the workflow w, population $\mathcal{D}_w$, evaluation sample S_w, and environment E_w; the *evaluation specification* defines the evaluator g_w, the success predicate ϕ_w, and any routing signal exposed to an oversight policy; and *deployment assumptions* (target Y, human-oversight model, cost model, risk tolerance) are kept separate from measured evidence. The development and held-out qualification partitions of Section 3.3 are versioned with the configuration so that certification uses only the qualification partition and the candidate family and test order are fixed before it is examined. For workflows requiring interaction, READY distinguishes an *oversight actor* (an organization-side reviewer who may approve, correct, escalate, or take over) from a *user actor* (an external counterparty); either may be human or simulated but must be part of the evaluation configuration. In τ^2 retail, the benchmark customer simulator is the user actor (Appendix D.1); CliniCARE-Bench uses patient-scoped EHR tools without a simulated customer. Neither study executes the human-review path, so reviewer effectiveness is a deployment assumption rather than an observed actor outcome.

New enterprise workflows can be added without changing the qualification methodology or execution infrastructure: a contribution supplies an executable workflow evaluation together with the specification artifacts above, and the same interface serves public benchmarks and enterprise-internal evaluations without requiring proprietary data to be released. Contributions are validated against workflow, evaluation, execution, and qualification criteria by an independent reviewer, and each result is tied to a versioned evaluation lineage so a deployment claim can be reproduced, audited, and requalified as assumptions, evaluators, or agents change.

C SUPPLEMENTARY ANALYSES FOR CLINICARE-BENCH

This appendix supplements the four-way deployment-qualification analysis of Section 5 with a step-by-step reproduction of the certified operating point, partition and sample-size sensitivity checks, and complete per-system results. All certified operating points follow the global-routing family and the fixed-sequence betting test of Section 4.2, computed on the qualification partition with the family and test order fixed in advance. Appendix C.8 adds a class-conditioned control and a separately identified descriptive frontier comparison.

C.1 AUTONOMOUS PERFORMANCE AND ROUTING QUALITY (DEVELOPMENT PARTITION)

Figure 6 shows the risk–coverage curves referenced in Section 5.1, and Table 2 reports autonomous success and routing quality for all 18 systems on the development partition. These routing diagnostics contextualize the independently computed qualification frontier in Figure 3.

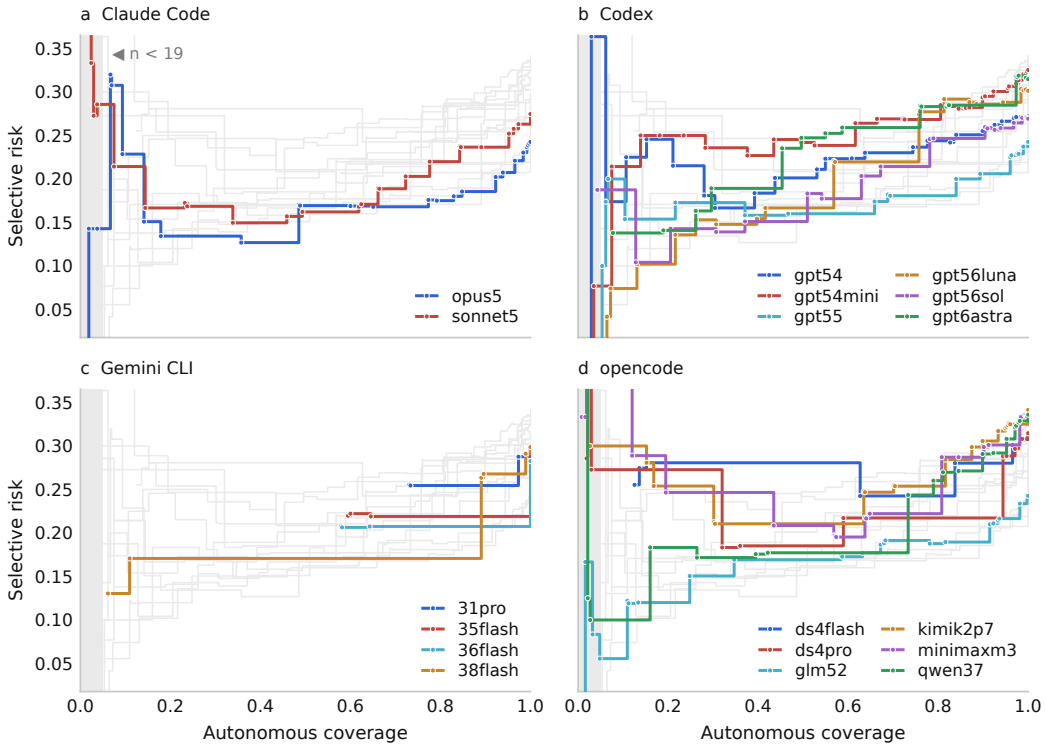


Figure 6: **Risk–coverage behavior of the stated-confidence routing signal on the development partition**, by agent harness. Sweeping the routing threshold traces the tradeoff between autonomous coverage and failure risk among autonomously accepted cases; lower curves indicate more effective selective routing. Each panel colors one harness's systems over all 18 systems in grey. Dots mark each system's reachable operating points: only a threshold at a tie-group boundary of its stated confidences can be selected. The shaded band marks coverage levels supported by fewer than 19 accepted cases, where the selective-risk estimate is unstable.

System	a_0	AURC ↓	AUROC ↑	Succ@50%
`Claude Code`				
Opus 5	75.7	0.169	0.680	83.1
Sonnet 5	72.5	0.203	0.681	83.7
`Codex`				
GPT-5.6-Sol	73.1	0.189	0.640	81.8
GPT-5.6-Luna	69.9	0.195	0.645	79.9
GPT-5.5	75.7	0.172	0.652	83.7
GPT-5.4	72.8	0.222	0.601	79.2
GPT-5.4-mini	67.5	0.248	0.628	75.9
GPT-6-Astra	68.5	0.230	0.617	75.2
`Gemini CLI`				
Gemini-3.8-Flash	70.1	0.238	0.598	74.7
Gemini-3.6-Flash	71.7	0.223	0.619	79.3
Gemini-3.5-Flash	70.4	0.235	0.618	78.0
Gemini-3.1-Pro	70.9	0.261	0.563	74.5
`opencode`				
DeepSeek-V4-Pro	68.5	0.229	0.664	79.2
DeepSeek-V4-Flash	68.8	0.263	0.602	75.4
GLM-5.2	75.7	0.168	0.652	82.8
Qwen-3.7-Plus	66.4	0.219	0.711	79.8
MiniMax-M3	66.4	0.255	0.676	79.8
Kimi-K2.7-Code	65.9	0.261	0.661	76.1

Table 2: **Autonomous performance and routing quality on the development partition.** $n = 375$ cases per system. a_0 is four-way verdict accuracy under full autonomy. AURC summarises the risk–coverage curve of Figure 6; lower is better. The deployment frontier uses threshold-specific coverage and success together with the review assumption. AUROC measures ranking quality independent of baseline accuracy; higher is better. Succ@50% is selective success over the half of cases each system is most confident in. GLM-5.2 and GPT-5.5 coincide on both a_0 (284 of 375 cases) and AUROC to the reported precision; their risk–coverage curves differ (Figure 6), so they are distinct systems that share one point in Figure 2.

C.2 Reproducing the Certified Operating Point

The Review%, $\widehat{R}$, and LCB columns of Table 1 are produced as follows for one system and one target Y; $N_{\text{qual}} = 375$, $a_h = 0.90$, $\alpha = 0.05$, $\varepsilon = 1 - Y$.

1. **Candidates.** Begin with the prespecified percentage-point grid $\mathcal{G} = \{0, 0.01, \ldots, 1\}$. The candidate sequence is review-everything (π_∞), then the grid thresholds in descending order. For computation, adjacent thresholds that induce identical routing are collapsed; the retained sequence therefore corresponds to the observed tie-group boundaries, and its final threshold accepts every case.
2. **Route and score.** For a candidate τ set $d_i = \mathbf{1}\{s_i \geq \tau\}$ (Equation 23), form $y_i(a_h) = d_i u_i + (1 - d_i)a_h$, and the loss $L_i = 1 - y_i(a_h) \in [0, 1]$. The review burden is $(N_{\text{qual}} - \sum_i d_i)/N_{\text{qual}}$ and the point estimate $\widehat{R} = 1 - \bar{L}$ (Equation 28).
3. **p-value.** Compute the betting p-value $p_\tau = \min\{1, 1/\bar{K}_n(\varepsilon)\}$ of Equation 30, with $\bar{K}_n$ the average final capital over 64 fixed permutations of the 375 cases, $c = 3/4$, and the variance-adaptive bet of Appendix A.5. For π_∞ every $L_i = 1 - a_h$ and reliability equals a_h by assumption, so π_∞ is certified whenever $Y \leq a_h$.
4. **Fixed-sequence test.** Walk the candidate sequence in order and reject H_τ while $p_\tau \leq \alpha$; stop at the first $p_\tau > \alpha$. The rejected candidates form $\widehat{\mathcal{T}}_\alpha(Y)$.
5. **Certified operating point.** Report the last rejected candidate, which has the lowest review burden in $\widehat{\mathcal{T}}_\alpha(Y)$ (Equation 5), together with its Review% and $\widehat{R}$.
6. **Lower bound.** The LCB is the largest Y' on a 0.005 grid for which the same candidate's p-value at $\varepsilon' = 1 - Y'$ is at most α (Equation 31); by construction LCB $\geq Y$ for every

test-certified point. For π_∞, we instead report the declared reliability a_h, which is exact under the full-review assumption.

For τ^2 retail the same steps are applied to the 57 task-level losses, each the mean of $1 - y_i(a_h)$ over the task's four trials.

C.3 Partition and Sample-Size Sensitivity

Table 1 reports certificates computed on one prespecified qualification half, so its review burdens also reflect which cases entered that half. Certification needs no development-side policy selection, so the same procedure can be run separately on the development half and on the full cohort to test sensitivity to the evaluation partition and sample size. Table 3 reports the Sonnet 5–GPT-5.4 comparison and the mean certified review burden over all systems under each choice. The direction of the comparison is the same in all six rows, and in both workflows; the size of the gap varies with the sample, as expected with 375 cases and especially with 57 tasks. Certifying on the full cohort lowers mean review in both workflows because larger samples yield larger certified sets.

Workflow	Calibration set	Sonnet 5	GPT-5.4	Gap (pp)	Mean, all systems
CliniCARE	Qualification half, 375 cases*	12.0	29.3	17.3	37.5
CliniCARE	Development half, 375 cases	22.4	36.3	13.9	40.4
CliniCARE	Full cohort, 750 cases	11.5	24.8	13.3	31.7
τ^2 retail	Qualification half, 57 tasks*	7.5	34.2	26.7	24.0
τ^2 retail	Development half, 57 tasks	8.8	20.6	11.8	33.7
τ^2 retail	Full cohort, 114 tasks	2.9	19.5	16.6	19.4

Table 3: **Certified review burden (%) at $Y = 0.75$ when the certification procedure is run on different calibration sets.** *: the calibration set used for the reported tables. The development-half and full-cohort rows are partition and sample-size sensitivity analyses; the full cohort has twice the sample size and yields larger certified sets here. $a_h = 0.90$ throughout.

C.4 Sensitivity to the Human-Review Assumption

The main analysis treats human-review success a_h as a declared deployment assumption (Section 4.2). Here we recompute the certified sets of Section 5.2 under alternative values of a_h, reporting how the certified review burden changes with the assumed quality of the human review path. We repeat the certification at $a_h \in \{0.80, 0.85, 0.90, 0.95, 1.00\}$ over a dense 31-target sensitivity sweep ($Y = 0.65, 0.655, \ldots, 0.80$), giving $31 \times 18 = 558$ certified points per assumption. Nothing about the execution evidence changes; only the declared assumption does.

Table 4 reports the per-system consequences at the representative target $Y = 0.75$. The aggregate picture over the full sweep is that the review assumption materially changes the oversight required across systems:

a_h	Mean certified Review%	Certified points at 100% review
0.80	42.2	65/558
0.85	33.1	10/558
0.90	29.0	2/558
0.95	26.3	0/558
1.00	24.6	0/558

Weakening the assumed review path from $a_h = 0.90$ to 0.80 raises mean certified review by 13 percentage points and leaves 65 of 558 certified points at full review, against 2 at $a_h = 0.90$ and none at $a_h \geq 0.95$. A larger a_h lowers the loss of every reviewed case, and in this sweep certified review burden is nonincreasing in a_h for every system. It moves in steps, because each system can only move between the thresholds its stated confidence actually reaches. Sonnet 5 is unchanged at 12.0% review for every a_h, while GPT-5.4 falls from 46.9% to 27.2%; Gemini-3.1-Pro is unchanged at 33.6% throughout, which reflects the granularity limit of Section 5.2 rather than a response to the assumption.

System	$a_h = 0.80$	$a_h = 0.85$	$a_h = 0.90$	$a_h = 0.95$	$a_h = 1.00$
`Claude Code`					
Opus 5	22.4	22.4	22.4	10.4	8.5
Sonnet 5	12.0	12.0	12.0	12.0	12.0
`Codex`					
GPT-5.6-Sol	32.5	32.5	32.5	22.1	22.1
GPT-5.6-Luna	56.0	56.0	56.0	41.6	41.6
GPT-5.5	15.5	15.5	15.5	15.5	15.5
GPT-5.4	46.9	39.2	29.3	27.2	27.2
GPT-5.4-mini	39.5	36.3	29.3	29.1	29.1
GPT-6-Astra	57.3	49.6	43.2	43.2	43.2
`Gemini CLI`					
Gemini-3.8-Flash	85.6	85.6	85.6	85.6	85.6
Gemini-3.6-Flash	38.4	38.4	38.4	38.4	38.4
Gemini-3.5-Flash	38.7	38.7	38.7	38.7	38.7
Gemini-3.1-Pro	33.6	33.6	33.6	33.6	33.6
`opencode`					
DeepSeek-V4-Pro	44.0	44.0	44.0	44.0	44.0
DeepSeek-V4-Flash	84.5	37.1	37.1	37.1	37.1
GLM-5.2	32.5	32.0	32.0	32.0	32.0
Qwen-3.7-Plus	61.9	61.9	61.9	31.5	31.5
MiniMax-M3	60.8	42.7	33.9	33.9	33.9
Kimi-K2.7-Code	67.7	67.7	29.1	29.1	29.1

Table 4: **Certified review burden at $Y = 0.75$ under alternative human-review assumptions.** $N_{\text{qual}} = 375$. The certified set is recomputed for each a_h and the lowest-review certified threshold is reported.

C.5 Workflow Success Predicate: Process-Aware Qualification

The main analysis scores an autonomous execution as successful when the four-way verdict matches the reference adjudication (Equation 7). Here we repeat the principal comparisons using the process-aware variant $u_{i,\text{df}}$ (Equation 8), which additionally requires the absence of a disqualifying process defect, to test sensitivity of the routing-quality and frontier conclusions to the deployment-success definition. This analysis covers the 16 originally graded systems; Gemini-3.8-Flash and GPT-6-Astra were added after the process-grading pass and are excluded here (all u-predicate results for both appear in the other tables). CliniCARE records the defect as a single per-run indicator, and it is common: 21.9% of the 12,000 runs carry one, ranging from 12.7% (GPT-5.5) to 30.0% (Kimi-K2.7-Code). A defect can co-occur with a correct verdict, so substituting $u_{i,\text{df}}$ for u_i lowers autonomous success by 4.5 to 13.1 percentage points depending on the system.

The routing-quality conclusion of Section 5.1 is stable under the substitution, while the deployment profiles shift substantially. Routing quality measures a similar underlying property under either predicate: the Spearman correlation between AUROC under u and under u_{df} across the 16 systems is 0.86, and per-system AUROC moves by at most 0.032 (Table 5). Qwen-3.7-Plus remains among the best-ranking signals and Gemini-3.1-Pro the worst. The deployment consequences, by contrast, shift substantially. Mean certified review across the 31 targets and 16 systems (496 certified points) rises from 26.7% to 49.6%, and at $Y = 0.75$ all three Gemini systems in this cohort move to full review: Gemini-3.6-Flash and Gemini-3.5-Flash from roughly 38% review and Gemini-3.1-Pro from 33.6%, so that under the process-aware success definition none of them has a certified operating point that deploys. This strengthens rather than weakens the deployability message of Section 5.2: GPT-5.4 and Sonnet 5 continue to certify deploying operating points, GPT-5.4 at 39.2% review (from 29.3%) and Sonnet 5 at 29.6% (from 12.0%), while GLM-5.2 moves from 32.0% to 44.8%. A harder success predicate does not change which systems can be certified, since sufficient review always reaches $Y < a_h$, but it changes the review each certificate costs; reporting review burden alongside the certificate therefore remains necessary.

System	a_0		AUROC		Review%		Certified point under u_{df}	
	u	u_{df}	u	u_{df}	u	u_{df}	$\widehat{R}$	95% LCB
`Claude Code`								
Opus 5	75.7	69.6	0.680	0.691	22.4	25.1	0.810	0.755
Sonnet 5	72.5	65.1	0.681	0.687	12.0	29.6	0.808	0.755
`Codex`								
GPT-5.6-Sol	73.1	67.7	0.640	0.631	32.5	32.5	0.805	0.750
GPT-5.6-Luna	69.9	64.3	0.645	0.631	56.0	56.0	0.829	0.780
GPT-5.5	75.7	70.9	0.652	0.664	15.5	21.3	0.808	0.755
GPT-5.4	72.8	68.3	0.601	0.630	29.3	39.2	0.806	0.755
GPT-5.4-mini	67.5	57.3	0.628	0.624	29.3	49.1	0.799	0.750
`Gemini CLI`								
Gemini-3.6-Flash	71.7	58.7	0.619	0.612	38.4	100.0†	0.900	0.900*
Gemini-3.5-Flash	70.4	58.7	0.618	0.597	38.7	100.0†	0.900	0.900*
Gemini-3.1-Pro	70.9	58.7	0.563	0.571	33.6	100.0†	0.900	0.900*
`opencode`								
DeepSeek-V4-Pro	68.5	58.1	0.664	0.652	44.0	68.5	0.865	0.825
DeepSeek-V4-Flash	68.8	56.3	0.602	0.633	37.1	84.5	0.891	0.850
GLM-5.2	75.7	66.1	0.652	0.678	32.0	44.8	0.811	0.760
Qwen-3.7-Plus	66.4	54.1	0.711	0.679	61.9	61.9	0.816	0.770
MiniMax-M3	66.4	55.2	0.676	0.653	33.9	60.8	0.819	0.770
Kimi-K2.7-Code	65.9	53.1	0.661	0.668	29.1	67.7	0.828	0.785

Table 5: **Routing quality and certified operating points under the process-aware success definition.** a_0 and AUROC are on S_{dev} ($n = 375$); Review%, $\widehat{R}$ and the LCB are on S_{qual} at $Y = 0.75$, $a_h = 0.90$, with the certified set recomputed under each predicate. AUROC is the probability that a successful run states higher confidence than a failed run, ties credited at 0.5, using the corresponding success predicate in each column. † denotes zero autonomous coverage; * reports the declared full-review reliability $a_h = 0.900$, not a test-inverted LCB.

C.6 Agent Execution Cost

Table 6 gives the agent cost per case k_m used in Section 5.3. The evaluation harness records the number of input tokens, the part of them served from cache, and the number of output tokens; k_m is the mean over the available run-level records for each system of these three quantities priced at the per-token list prices of the evaluation gateway at the time of the runs, with cached input priced at its own rate. Coverage is 746–750 of 750 CliniCARE runs depending on the system and all 456 retail trials per system. The same prices and formula are used for both workflows. CliniCARE cases are long: each run reads on the order of a million input tokens of patient record, mostly from cache, so k_m ranges from \$0.05 (GPT-5.6-Luna) to \$3.31 (Kimi-K2.7-Code) per case. Retail trials are shorter and cheaper (\$0.01 to \$0.49). The opencode systems are priced conservatively. That route does not report cache reads, so their input tokens are priced at the full input rate and their k_m values are upper bounds. The harness also records its own cost estimate for the Anthropic and Gemini routes; it applies a different price book, so we use the gateway prices throughout for comparability.

Combining these agent costs with the certified review burdens gives the representative cross-workflow rankings in Table 7. The full cost curves are shown in Figure 4.

C.7 Complete Per-System Results

This section reports per-system certified operating points across targets, predicted-verdict class frequencies and within-class autonomous success, and the number of distinct routing-score values. Development-set routing diagnostics are in Appendix C.1.

Certified operating points across targets. Table 8 extends the single-target view of Table 1 to the five targets of the sensitivity grid, $Y \in \{0.65, 0.70, 0.75, 0.80, 0.85\}$, spanning the range in which most systems have certified deploying operating points; at $Y = 0.85$ review burden ranges

System	CliniCARE (per case)				τ^2 retail (per trial)			
	Input tok.	Cached	Output	k_m ($)	Input tok.	Cached	Output	k_m ($)
`Claude Code`								
Opus 5	1623	1508	20	1.83	630	616	4	0.49
Sonnet 5	1086	1005	13	0.50	692	678	4	0.20
`Codex`								
GPT-5.6-Sol	960	880	9	1.12	274	253	3	0.31
GPT-5.6-Luna	1132	1043	10	0.05	265	244	3	0.01
GPT-5.5	1139	997	10	1.51	270	254	3	0.29
GPT-5.4	1149	1041	13	0.73	221	207	4	0.14
GPT-5.4-mini	1150	1073	14	0.20	211	195	4	0.05
GPT-6-Astra	1022	967	5	1.77	237	218	2	0.49
`Gemini CLI`								
Gemini-3.8-Flash	3034	2687	25	0.55	413	335	10	0.12
Gemini-3.6-Flash	2530	2095	24	0.57	371	278	9	0.12
Gemini-3.5-Flash	2471	2042	24	1.17	432	355	10	0.26
Gemini-3.1-Pro	1317	990	8	0.95	330	259	4	0.24
`opencode`								
DeepSeek-V4-Pro	635	0	5	0.86	205	0	2	0.28
DeepSeek-V4-Flash	830	0	5	0.19	225	0	2	0.05
GLM-5.2	769	0	6	1.10	196	0	2	0.28
Qwen-3.7-Plus	473	0	5	0.20	—	—	—	—
MiniMax-M3	1288	0	9	0.40	226	0	2	0.07
Kimi-K2.7-Code	3439	0	11	3.31	188	0	2	0.19

Table 6: **Agent execution cost per case.** Mean token usage per case (thousands; cached input is the part of input served from cache) over the available run-level records (746–750 of 750 CliniCARE runs depending on the system, and all 456 retail trials), and the resulting agent cost k_m at the evaluation gateway's per-token list prices as of 2026-09-12 (input, cached input, and output priced separately; list prices change over time, and GPT-5.6-Sol's had already fallen by the time of writing). The opencode route does not report cache reads, so those input tokens are priced at the full input rate and their k_m is an upper bound. Qwen-3.7-Plus was not run on retail.

Workflow	$k_h = \$2$	$k_h = \$10$	$k_h = \$50$
CliniCARE	Sonnet 5 ($0.74) GPT-5.4-mini ($0.79) DeepSeek-V4-Flash ($0.93)	Sonnet 5 ($1.70) GPT-5.5 ($3.06) GPT-5.4-mini ($3.13)	Sonnet 5 ($6.50) GPT-5.5 ($9.26) Opus 5 ($13.03)
τ^2 retail	Gemini-3.8-Flash ($0.12) MiniMax-M3 ($0.19) Gemini-3.6-Flash ($0.20)	Gemini-3.8-Flash ($0.12) Gemini-3.5-Flash ($0.44) DeepSeek-V4-Pro ($0.46)	Gemini-3.8-Flash ($0.12) Gemini-3.5-Flash ($1.16) DeepSeek-V4-Pro ($1.18)

Table 7: **Lowest-cost certified systems at $Y = 0.75$ for representative review prices.** For each k_h, the table reports the three systems with lowest total cost per case $k_m + k_h e$; $a_h = 0.90$.

from 52.0% to 100% and three Gemini systems are certified only at full review. The step-function behaviour noted in Section 5.2 is visible throughout: review burden is flat across adjacent targets for systems whose confidence takes few distinct values, then jumps. Gemini-3.1-Pro is pinned at 100% from $Y = 0.80$ upward and Gemini-3.8-Flash at 85.6% from $Y = 0.70$ to 0.85.

Routing behaviour by verdict class. Table 9 breaks the qualification partition down by predicted verdict. Pooled across systems, autonomous success is strongly class-dependent (0.863 for *Yes*, 0.662 for *No*, 0.526 for *Indeterminate: Lack of Data* and 0.330 for *Indeterminate: Medically Ambiguous*), while mean stated confidence varies far less (0.898, 0.896, 0.856, 0.812). GPT-6-Astra is the outlier in class mix. It assigns 34.7% of qualification cases to *Indeterminate: Lack of Data* (reference rate 15.7%) at only 40.0% within-class success, an abstention-heavy profile whose con-

sequences for class-based escalation appear in Appendix C.8. Systems are therefore somewhat but not proportionately less confident on the verdicts they get wrong more often.

The result that matters for the policy design of Section 4.2 is that the routing signal retains ranking power *within* every predicted class: pooled within-class AUROC is 0.642 for *Yes*, 0.609 for *No*, 0.607 for *ILD* and 0.659 for *IMA*. The signal is thus not a proxy for the predicted class. This supports the use of confidence for routing, but does not exclude additional information from predicted class; Appendix C.8 evaluates a class-conditioned control.

	$Y = 0.65$		$Y = 0.70$		$Y = 0.75$		$Y = 0.80$		$Y = 0.85$	
System	Rev%	LCB	Rev%	LCB	Rev%	LCB	Rev%	LCB	Rev%	LCB
`Claude Code`										
Opus 5	0.0	0.695	0.8	0.700	22.4	0.785	26.7	0.800	52.0	0.850
Sonnet 5	0.0	0.690	2.9	0.710	12.0	0.760	29.6	0.805	55.2	0.850
`Codex`										
GPT-5.6-Sol	0.0	0.670	22.1	0.745	32.5	0.795	44.5	0.815	78.9	0.850
GPT-5.6-Luna	16.0	0.650	41.6	0.745	56.0	0.810	56.0	0.810	77.3	0.850
GPT-5.5	0.0	0.705	0.0	0.705	15.5	0.775	35.7	0.830	53.1	0.850
GPT-5.4	5.6	0.650	19.2	0.705	29.3	0.750	49.9	0.805	72.3	0.850
GPT-5.4-mini	8.3	0.650	16.0	0.700	29.3	0.750	49.1	0.805	73.1	0.850
GPT-6-Astra	18.1	0.670	25.1	0.700	43.2	0.770	53.9	0.800	81.9	0.850
`Gemini CLI`										
Gemini-3.8-Flash	0.0	0.650	9.1	0.700	85.6	0.850	85.6	0.850	85.6	0.850
Gemini-3.6-Flash	0.0	0.680	38.4	0.800	38.4	0.800	38.4	0.800	100.0†	0.900*
Gemini-3.5-Flash	0.0	0.670	38.7	0.800	38.7	0.800	38.7	0.800	100.0†	0.900*
Gemini-3.1-Pro	0.0	0.685	33.6	0.790	33.6	0.790	100.0†	0.900*	100.0†	0.900*
`opencode`										
DeepSeek-V4-Pro	2.9	0.650	44.0	0.790	44.0	0.790	68.5	0.845	69.9	0.850
DeepSeek-V4-Flash	0.0	0.665	14.9	0.730	37.1	0.780	84.5	0.850	84.5	0.850
GLM-5.2	0.0	0.650	10.9	0.705	32.0	0.780	44.8	0.810	71.5	0.850
Qwen-3.7-Plus	11.2	0.655	19.2	0.700	61.9	0.835	61.9	0.835	74.1	0.850
MiniMax-M3	13.6	0.665	33.9	0.755	33.9	0.755	60.8	0.830	79.7	0.850
Kimi-K2.7-Code	7.5	0.660	29.1	0.750	29.1	0.750	67.7	0.820	79.7	0.850

Table 8: **Certified operating points across five reliability targets.** $N_{\text{qual}} = 375$, $a_h = 0.90$. Rev% is the review burden of the lowest-review certified threshold and LCB its one-sided 95% bound (Appendix C.2). The $Y = 0.75$ block reproduces Table 1. † denotes zero autonomous coverage; the corresponding * entry is the declared reliability $a_h = 0.900$, not a test-inverted LCB.

C.8 Class-Conditioned Routing Control: Always-Review-Indeterminate

The main-text oversight policy acts only on the routing signal and is agnostic to the four adjudication classes (Section 4.2). As an alternative policy family, we contrast it with a policy that also escalates on the *predicted class*: every indeterminate adjudication is routed to human review regardless of confidence,

$$\pi_{\text{IR}}(s_i, \widehat{y}_i) = \begin{cases} \text{review}, & \widehat{y}_i \in \{\text{ILD}, \text{IMA}\}, \\ \text{accept}, & \widehat{y}_i \in \{\text{Yes}, \text{No}\} \ \wedge \ s_i \geq \tau, \\ \text{review}, & \text{otherwise}. \end{cases} \tag{36}$$

This baseline tests predicted verdict class as an additional routing rule. It routes every indeterminate adjudication to review even though an indeterminate verdict can be a correct task-level outcome under ϕ_w; deployment-level escalation remains a separate decision whose value must be established empirically. We certify π_{IR} exactly as the signal-only policy, applying the fixed-sequence betting test to its own threshold family, and report the comparison in Table 10.

System	*Yes*		*No*		*ILD*		*IMA*		
	%	Succ.	%	Succ.	%	Succ.	%	Succ.	$\lvert\{s_i\}\rvert$
`Claude Code`									
Opus 5	42.4	86.8	38.4	71.5	16.0	58.3	3.2	58.3	25
Sonnet 5	43.7	85.4	36.5	73.0	16.0	63.3	3.7	28.6	26
`Codex`									
GPT-5.6-Sol	38.4	91.7	39.2	67.3	18.7	54.3	3.7	42.9	19
GPT-5.6-Luna	34.1	86.7	41.1	61.0	18.4	40.6	6.4	25.0	21
GPT-5.5	41.1	90.9	42.1	68.4	13.9	61.5	2.9	63.6	25
GPT-5.4	35.7	88.8	42.1	65.8	17.9	47.8	4.3	31.2	32
GPT-5.4-mini	39.5	83.8	42.7	60.0	12.3	45.7	5.6	19.0	33
GPT-6-Astra	29.6	93.7	30.7	78.3	34.7	40.0	5.1	21.1	16
`Gemini CLI`									
Gemini-3.8-Flash	36.5	92.0	42.7	64.4	14.7	54.5	5.9	36.4	8
Gemini-3.6-Flash	42.9	86.3	39.7	67.1	13.3	66.0	4.0	40.0	5
Gemini-3.5-Flash	44.3	84.9	40.5	66.4	11.7	63.6	3.5	38.5	5
Gemini-3.1-Pro	39.7	87.2	37.1	71.2	21.3	56.2	1.9	71.4	4
`opencode`									
DeepSeek-V4-Pro	38.7	86.2	41.6	62.2	14.7	54.5	4.8	33.3	15
DeepSeek-V4-Flash	49.1	81.0	35.7	70.1	11.7	56.8	3.5	30.8	10
GLM-5.2	45.9	80.8	39.2	68.0	11.2	57.1	3.7	35.7	23
Qwen-3.7-Plus	35.2	87.9	42.4	58.5	14.9	39.3	7.5	35.7	18
MiniMax-M3	40.0	79.3	41.9	62.4	9.9	56.8	8.3	22.6	19
Kimi-K2.7-Code	39.7	84.6	43.5	61.3	9.1	55.9	7.7	24.1	14

Table 9: **Predicted-class mix and within-class autonomous success on the qualification partition.** % is the share of the 375 cases receiving each predicted verdict; Succ. is four-way verdict accuracy within that share. *ILD* and *IMA* abbreviate the two indeterminate classes of Equation 6. $\lvert\{s_i\}\rvert$ is the number of distinct stated-confidence values the system emits on this partition, the granularity limit discussed in Section 5.2.

Predicted class is informative but not a substitute for certified routing. The class-conditioned control is competitive on this benchmark. Averaged over the 31-target, 18-system sweep (558 certified points), π_{IR} routes 28.9% of cases to review against 29.0% for the signal-only policy; it requires strictly more review in 337 of 558 cells, but by small amounts, and less review in the remainder. The single-target comparison could be dismissed as threshold-grid overshoot, so the right-hand columns of Table 10 instead report, for each policy, the minimum review burden at which a deploying operating point *achieves* a given reliability on the qualification partition. This compares policies at common achieved-reliability requirements, although their reachable operating points remain discrete. These columns are descriptive frontier comparisons rather than certified policy selections. The control needs more review on 10 of 18 systems at $\widehat{R} = 0.80$ and 7 of 17 at $\widehat{R} = 0.85$, with mean differences (π_{IR} minus π_τ) of -5.5 and -6.4 percentage points. GPT-6-Astra is the strongest case for the control: its indeterminate verdicts are both frequent and unreliable (34.7% of cases at 40.0% success), so escalating on the class alone reaches $\widehat{R} = 0.85$ at 39.7% review where the signal-only policy needs 70.9%.

The reason is visible in Table 9. Indeterminate adjudications really are much less often correct than definite ones on this benchmark (0.480 against 0.762 pooled on the qualification partition), so predicted class carries genuine information about autonomous failure. Predicted class is therefore a useful routing feature in this cohort.

What the fixed class rule gives up. The limitation of π_{IR} is structural rather than budgetary. No threshold can accept an indeterminate case, so the fixed class constraint sets a hard floor on each system's review burden and restricts the frontier to policies that never accept indeterminate cases, unlike the signal-only family of Section 4.2. It also inherits the benchmark's class-success profile: its competitiveness reflects that indeterminate verdicts are less reliable in this cohort, while the fixed class constraint cannot adapt if that relationship changes. Certification can detect a resulting

failure to meet the reliability target. The signal-based policy makes the same trade explicit and tunable, and Appendix C.7 shows the routing signal remains informative *within* each class, so the two are complementary sources of information rather than substitutes. A policy using both could exploit complementary information, but any improvement would require its own certification. The distinction is between an adjudication outcome and an escalation decision; predicted class may inform that decision when its value is evaluated.

	Certified at $Y = 0.75$			Min. Review% to achieve $\widehat{R}$			
	Review%			$\widehat{R} = 0.80$		$\widehat{R} = 0.85$	
System	π_τ	π_{IR}	$\widehat{R}_{\mathrm{IR}}$	π_τ	π_{IR}	π_τ	π_{IR}
`Claude Code`							
Opus 5	22.4	19.2	0.816	10.4	19.2	37.9	34.4
Sonnet 5	12.0	19.7	0.818	12.0	19.7	29.6	34.7
`Codex`							
GPT-5.6-Sol	32.5	22.4	0.818	22.1	22.4	44.5	41.1
GPT-5.6-Luna	56.0	33.6	0.804	56.0	33.6	56.0	54.9
GPT-5.5	15.5	16.8	0.812	15.5	16.8	35.7	28.0
GPT-5.4	29.3	27.5	0.812	29.3	23.2	59.2	44.3
GPT-5.4-mini	29.3	31.7	0.803	29.1	31.2	53.6	45.9
GPT-6-Astra	43.2	39.7	0.875	43.2	39.7	70.9	39.7
`Gemini CLI`							
Gemini-3.8-Flash	85.6	25.3	0.820	85.6	25.3	85.6	86.1
Gemini-3.6-Flash	38.4	45.6	0.864	38.4	45.6	43.5	45.6
Gemini-3.5-Flash	38.7	45.3	0.859	38.7	45.3	43.2	45.3
Gemini-3.1-Pro	33.6	23.2	0.820	33.6	23.2	—	52.8
`opencode`							
DeepSeek-V4-Pro	44.0	54.9	0.860	44.0	54.9	68.5	54.9
DeepSeek-V4-Flash	37.1	26.1	0.825	37.1	26.1	84.5	45.3
GLM-5.2	32.0	23.5	0.811	32.0	23.5	44.8	40.3
Qwen-3.7-Plus	61.9	38.9	0.820	61.9	30.4	61.9	64.0
MiniMax-M3	33.9	39.2	0.825	33.9	39.2	60.8	64.8
Kimi-K2.7-Code	29.1	33.6	0.814	29.1	33.6	67.7	70.4

Table 10: **Signal-based routing against the always-review-indeterminate control**, $a_h = 0.90$, $N_{\mathrm{qual}} = 375$. Left block: the lowest-review certified threshold of each policy at $Y = 0.75$, with the same fixed-sequence betting test applied to each policy's own threshold family; $\widehat{R}_{\mathrm{IR}}$ is the point estimate of the control's certified point. Right block: the minimum review burden at which each policy attains the stated reliability among its own reachable operating points on S_{qual}, a descriptive frontier comparison at common achieved-reliability requirements rather than a certified selection; — marks a reliability no operating point of that policy reaches.

D Supplementary Analyses for τ^2 Retail

This appendix supplements the τ^2 retail methods in Section 4.1, the shared protocol in Section 4.2, and the results in Section 5.4: the execution protocol, the routing-quality table, and the certified operating points at every target of the sensitivity grid.

D.1 Protocol

Workflow and cases. The workflow is the retail domain of τ^2-bench (Barres et al., 2026): an agent resolves a customer's request (orders, exchanges, returns, address and payment changes) by conversing with a simulated customer and calling the retailer's backend tools under a written domain policy. The evaluation sample is the benchmark's 114 retail tasks; the customer is the benchmark's own user simulator (GPT-5.2 at low reasoning effort, the leaderboard setting) and is the *user actor* of Appendix B, external to the organization and not an oversight actor. Each system runs every task 4 times with a fresh simulator seed per trial, giving 456 trials per system and 7,752 trials over

the 17 systems. The systems are the CliniCARE cohort except Qwen-3.7-Plus, whose model route was not deployable behind our API gateway. The evaluator g_w is the benchmark's reward (final database state matches the reference, required actions performed, required information communicated, natural-language assertions judged by GPT-5.2) and the success predicate ϕ_w is reward $= 1$; mean reward over trials is the benchmark's *pass*^{1}, so this is the same metric used by the public leaderboard. Direct score comparisons additionally require matched harness, simulator, and reporting settings.

Routing signal and report protocol. As in CliniCARE, the routing signal s_i is a stated confidence: the agent ends each trial by writing a short report that closes with `## Confidence: NN%`. A trial with no report scores 0 and receives confidence 0, so every positive routing threshold sends it to review; at the accept-everything threshold $\tau = 0$, it remains an autonomous failure. Some CLI harnesses require completion safeguards to enforce this common conversation-and-report protocol. Claude Code uses its native `Stop` hook to prevent the run from ending while the report is missing. The opencode harness has no equivalent hook, so it resumes the same session at most three times while the report is missing. Both mechanisms provide only a protocol reminder, carry no task content or answer guidance, and leave any report still missing after the limit as a scored failure. Codex and Gemini CLI completed the protocol without an additional continuation mechanism. Residual missing reports were 58 trials for DeepSeek-V4-Flash, 8 for Kimi-K2.7-Code, 2 for GPT-5.6-Sol, and none elsewhere.

Sampling unit. Rows of the evidence table are trials, but the four trials of a task share its scenario, so tasks are the statistical unit (Appendix A.5). The 114 tasks are split once, stratified, into 57 development and 57 qualification tasks (228 trials each). The loss of a task is the mean of $1 - y_i(a_h)$ over its four trials, which remains in $[0, 1]$, and the fixed-sequence betting test of Section 4.2 is applied to the 57 task-level losses with $a_h = 0.90$ held fixed. Every other step is identical to CliniCARE.

D.2 Autonomous Performance and Routing Quality

Table 11 reports a_0 on the 228 development trials per system. These development estimates differ from pass^1 computed over all 456 trials; the paired routing comparison in Section 5.4 uses development estimates.

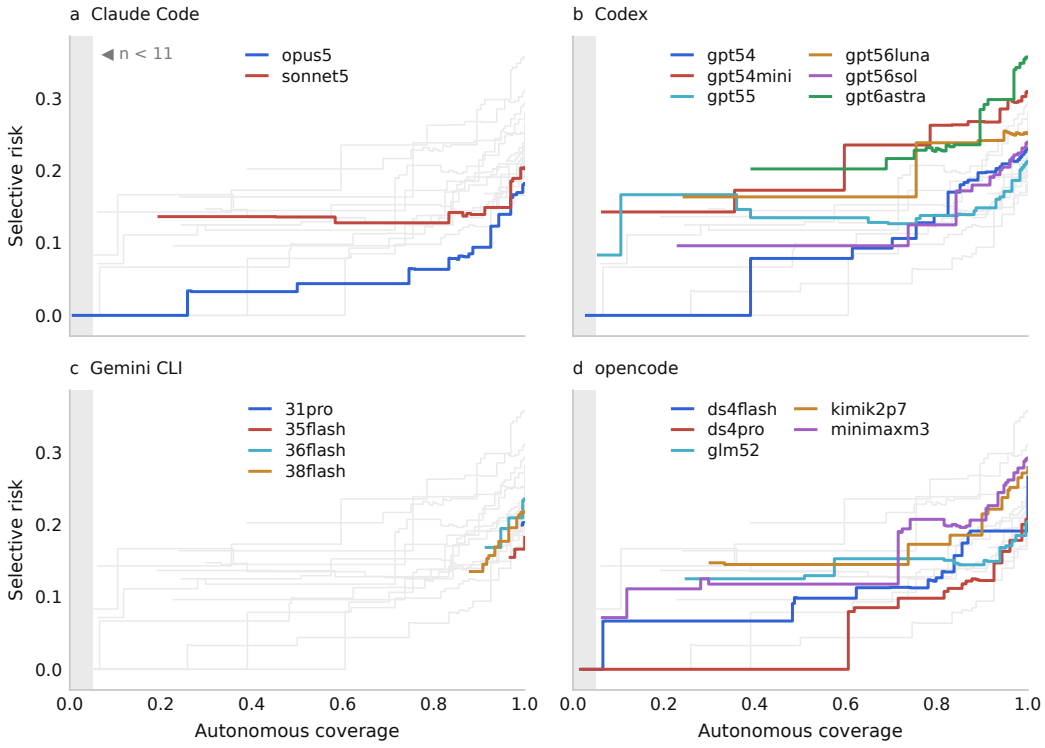


Figure 7: τ^2 **retail: risk–coverage curves on the development partition**, by agent harness; each panel colours one harness's systems over all 17 in grey. Three of the four Gemini CLI systems emit four to six distinct confidence values, so their curves consist of a few long steps and expose almost no intermediate operating points; Gemini-3.8-Flash emits 13.

System	a_0	AURC ↓	AUROC ↑	Succ@50%
`Claude Code`				
Opus 5	81.6	0.056	0.867	95.6
Sonnet 5	79.4	0.140	0.692	86.8
`Codex`				
GPT-5.6-Sol	75.9	0.128	0.760	88.1
GPT-5.6-Luna	74.6	0.210	0.571	78.0
GPT-5.5	78.5	0.141	0.709	86.9
GPT-5.4	76.8	0.105	0.784	91.3
GPT-5.4-mini	68.9	0.215	0.663	78.2
GPT-6-Astra	64.5	0.225	0.749	79.1
`Gemini CLI`				
Gemini-3.8-Flash	78.1	0.140	0.719	86.5
Gemini-3.6-Flash	76.3	0.171	0.674	83.2
Gemini-3.5-Flash	81.6	0.155	0.595	84.5
Gemini-3.1-Pro	79.8	0.199	0.508	80.1
`opencode`				
DeepSeek-V4-Pro	78.9	0.082	0.820	92.2
DeepSeek-V4-Flash	73.2	0.115	0.825	90.1
GLM-5.2	79.4	0.140	0.678	87.1
MiniMax-M3	70.6	0.163	0.749	83.2
Kimi-K2.7-Code	71.9	0.170	0.739	83.8

Table 11: τ^2 **retail: autonomous performance and routing quality on the development partition.** 228 trials (57 tasks) per system. a_0 is mean reward under full autonomy; AURC is the area under the risk–coverage curve of Figure 7 (lower is better); AUROC is the probability that a passing trial states higher confidence than a failing one, ties credited at 0.5; Succ@50% is success on the half of trials each system is most confident about. As on CliniCARE, AUROC shows no clear correlation with a_0 (Pearson -0.13), while AURC is tied to it more closely (Pearson -0.60).

D.3 Certified Operating Points

Table 12 is the τ^2 retail counterpart of Table 1 at the shared target $Y = 0.75$, and Table 13 extends it to the five-target grid. Figure 8 plots the corresponding reliability–oversight frontier.

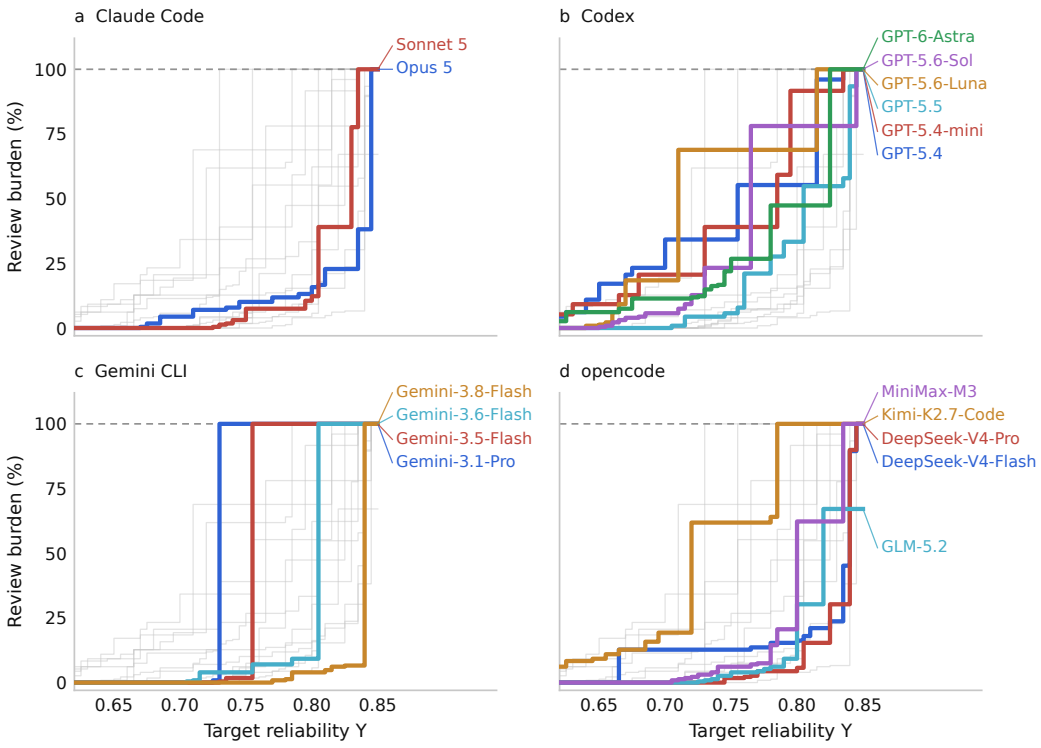


Figure 8: τ^2 **retail: reliability–oversight frontier on the held-out qualification partition**, by agent harness, at $a_h = 0.90$; each curve is the review burden of the lowest-review certified threshold as the target increases. Three of the four Gemini CLI systems emit four to six distinct confidence values, so their frontiers consist of a few long steps and jump from a low review share directly to 100%; Gemini-3.8-Flash, with 13 distinct values, is certified with no review at $Y = 0.75$ and at 4.0% review at $Y = 0.80$.

System	Review%	$\widehat{R}$	95% LCB
`Claude Code`			
Opus 5	10.1	0.876	0.765
Sonnet 5	7.5	0.887	0.790
`Codex`			
GPT-5.6-Sol	23.2	0.858	0.760
GPT-5.6-Luna	68.9	0.878	0.810
GPT-5.5	5.7	0.845	0.750
GPT-5.4	34.2	0.843	0.750
GPT-5.4-mini	39.0	0.869	0.780
GPT-6-Astra	26.8	0.868	0.775
`Gemini CLI`			
Gemini-3.8-Flash	0.0	0.860	0.765
Gemini-3.6-Flash	4.0	0.851	0.750
Gemini-3.5-Flash	1.8	0.858	0.750
Gemini-3.1-Pro	100.0†	0.900	0.900*
`opencode`			
DeepSeek-V4-Pro	1.8	0.875	0.755
DeepSeek-V4-Flash	12.7	0.869	0.760
GLM-5.2	4.0	0.878	0.765
MiniMax-M3	6.1	0.858	0.760
Kimi-K2.7-Code	61.8	0.850	0.775

Table 12: τ^2 **retail: certified operating points on the held-out qualification partition at** $Y = 0.75$**.** 228 qualification trials (57 tasks), $a_h = 0.90$. Tasks are the sampling unit: the per-task loss averages the four trials, and the fixed-sequence betting test of Section 3.3 is applied to the 57 task-level losses. Columns as in Table 1; † denotes zero autonomous coverage. For full review, * reports the declared reliability $a_h = 0.900$, which has no sampling uncertainty under the assumption, rather than a test-inverted LCB.

	$Y = 0.65$		$Y = 0.70$		$Y = 0.75$		$Y = 0.80$		$Y = 0.85$	
System	Rev%	LCB	Rev%	LCB	Rev%	LCB	Rev%	LCB	Rev%	LCB
`Claude Code`										
Opus 5	0.0	0.665	4.4	0.705	10.1	0.765	15.8	0.800	100.0†	0.900*
Sonnet 5	0.0	0.720	0.0	0.720	7.5	0.790	12.3	0.800	100.0†	0.900*
`Codex`										
GPT-5.6-Sol	0.4	0.650	5.7	0.705	23.2	0.760	78.1	0.840	100.0†	0.900*
GPT-5.6-Luna	1.3	0.650	18.4	0.705	68.9	0.810	68.9	0.810	100.0†	0.900*
GPT-5.5	0.0	0.700	0.0	0.700	5.7	0.750	33.3	0.800	100.0†	0.900*
GPT-5.4	17.1	0.665	34.2	0.750	34.2	0.750	55.3	0.810	100.0†	0.900*
GPT-5.4-mini	9.2	0.660	20.6	0.725	39.0	0.780	91.7	0.830	100.0†	0.900*
GPT-6-Astra	6.1	0.660	11.4	0.715	26.8	0.775	47.4	0.820	100.0†	0.900*
`Gemini CLI`										
Gemini-3.8-Flash	0.0	0.765	0.0	0.765	0.0	0.765	4.0	0.805	100.0†	0.900*
Gemini-3.6-Flash	0.0	0.700	0.0	0.700	4.0	0.750	9.2	0.800	100.0†	0.900*
Gemini-3.5-Flash	0.0	0.725	0.0	0.725	1.8	0.750	100.0†	0.900*	100.0†	0.900*
Gemini-3.1-Pro	0.0	0.725	0.0	0.725	100.0†	0.900*	100.0†	0.900*	100.0†	0.900*
`opencode`										
DeepSeek-V4-Pro	0.0	0.740	0.0	0.740	1.8	0.755	5.7	0.805	100.0†	0.900*
DeepSeek-V4-Flash	0.0	0.660	12.7	0.760	12.7	0.760	16.2	0.800	100.0†	0.900*
GLM-5.2	0.0	0.720	0.0	0.720	4.0	0.765	30.3	0.815	67.1	0.850
MiniMax-M3	0.0	0.700	0.0	0.700	6.1	0.760	62.3	0.850	100.0†	0.900*
Kimi-K2.7-Code	9.2	0.650	19.3	0.715	61.8	0.775	100.0†	0.900*	100.0†	0.900*

Table 13: τ^2 **retail: certified operating points across five reliability targets.** 228 qualification trials (57 tasks), $a_h = 0.90$; task-level fixed-sequence betting test. The $Y = 0.75$ block reproduces Table 12. † denotes zero autonomous coverage; the corresponding * entry is the declared reliability $a_h = 0.900$, not a test-inverted LCB.

D.4 A Second Routing Signal from Trial Agreement

All main-text results route on one signal type, the agent's stated confidence. The τ^2 retail study runs every task four times, so the trials themselves supply a second, consistency-based signal. For trial j of task t we define the *agreement* signal as the fraction of the other three trials of t that passed. Outcome agreement here stands in for final-state agreement, which a deployer could compute by re-running a task without knowing the reward, at a multiple of the agent cost. The evidence table stores only rewards, so the signal uses sibling outcomes and is an idealized upper bound on what a consistency signal can deliver, not a deployable policy. As a control that separates "a different signal" from "less noise," we also average stated confidence over the four trials (*mean confidence*). Both signals are task-level, so the task remains the sampling unit and the certification procedure of Section D.3 applies unchanged.

Table 14 reports routing quality on S_{dev} and the certified review burden at $Y = 0.75$, $a_h = 0.90$ on S_{qual} under each signal. Three observations. First, the headline of Section 5.1 does not depend on the signal. Autonomous accuracy shows no clear correlation with routing quality under stated confidence (Pearson $r = -0.13$), under agreement ($r = -0.24$), and under mean confidence ($r = -0.22$). Second, agreement is the stronger signal on average (mean AUROC 0.768 versus 0.712; mean certified review 13.9% versus 24.0%), whereas mean confidence is not (0.706; 26.5%), so the gain comes from the signal type rather than from averaging. Third, the two signals rank systems' routing quality almost independently (Spearman $\rho = 0.04$ between the AUROC columns): Opus 5 has the best stated-confidence signal but a middling agreement signal, and Gemini-3.1-Pro, certified only at 100% review under its four-valued stated confidence, is certified at 6.1% review under agreement. Routing quality is thus a property of a (system, signal) pair, and coarse verbalized confidence can understate how well a system could be routed by a stronger signal, although agreement itself is an idealized upper bound.

		AUROC (S_{dev})			Review% at $Y = 0.75$	
System	a_0	Confidence	Agreement	Mean conf.	Confidence	Agreement
`Claude Code`						
Opus 5	81.6	0.867	0.705	0.811	10.1	16.7
Sonnet 5	79.4	0.692	0.743	0.720	7.5	5.3
`Codex`						
GPT-5.6-Sol	75.9	0.760	0.813	0.733	23.2	19.3
GPT-5.6-Luna	74.6	0.571	0.808	0.553	68.9	19.3
GPT-5.5	78.5	0.709	0.759	0.764	5.7	4.8
GPT-5.4	76.8	0.784	0.767	0.705	34.2	20.2
GPT-5.4-mini	68.9	0.663	0.780	0.720	39.0	25.4
GPT-6-Astra	64.5	0.749	0.741	0.744	26.8	24.6
`Gemini CLI`						
Gemini-3.8-Flash	78.1	0.719	0.792	0.729	0.0	0.0
Gemini-3.6-Flash	76.3	0.674	0.863	0.753	4.0	5.7
Gemini-3.5-Flash	81.6	0.595	0.683	0.574	1.8	5.7
Gemini-3.1-Pro	79.8	0.508	0.755	0.532	100.0	6.1
`opencode`						
DeepSeek-V4-Pro	78.9	0.820	0.773	0.812	1.8	7.0
DeepSeek-V4-Flash	73.2	0.825	0.843	0.793	12.7	7.9
GLM-5.2	79.4	0.678	0.739	0.594	4.0	7.5
MiniMax-M3	70.6	0.749	0.778	0.780	6.1	5.3
Kimi-K2.7-Code	71.9	0.739	0.709	0.684	61.8	55.3

Table 14: τ^2 **retail under a consistency-based routing signal.** a_0 and AUROC on S_{dev} (228 trials, 57 tasks); AUROC credits ties at 0.5. Agreement is the pass fraction of a task's other three trials (four distinct values); mean confidence averages stated confidence over the four trials. Review% is the lowest-review certified threshold on S_{qual} (fixed-sequence betting test, $a_h = 0.90$). Across systems, Pearson $r(a_0, \text{AUROC}) = -0.13$, -0.24, and -0.22 for the three signals.

E Routing-Signal Ordering and Tail-Risk Qualification

E.1 Threshold Routing, Ordering, and Fixed-Sequence Search

Consider the trajectory-invariant terminal-oversight setting. For task instance i, autonomous execution produces an observable routing signal $s_i \in \mathbb{R}$ and an evaluation outcome $u_i \in \{0, 1\}$, where $u_i = 1$ denotes successful autonomous execution. The outcome u_i is available to the evaluator during policy development and qualification, but is generally not known when the routing decision is made.

A threshold policy π_τ accepts the agent output autonomously when $s_i \geq \tau$ and routes it to human review otherwise:

$$\pi_\tau(s_i) = \begin{cases} \text{accept}, & s_i \geq \tau, \\ \text{review}, & s_i < \tau. \end{cases} \tag{37}$$

For threshold τ, define the autonomous coverage

$$c(\tau) = \Pr(s \geq \tau), \tag{38}$$

and the success probability among autonomously accepted cases,

$$a(\tau) = \Pr(u = 1 \mid s \geq \tau). \tag{39}$$

One useful diagnostic property is *monotone selective success*: increasing the threshold does not decrease the success probability of the retained autonomous cases:

$$\tau_1 < \tau_2 \quad \Longrightarrow \quad a(\tau_1) \leq a(\tau_2), \tag{40}$$

for thresholds with nonzero autonomous coverage. Equivalently, increasingly selective autonomous acceptance should preferentially remove higher-risk cases.

A stronger sufficient condition is pointwise monotonicity,

$$s_1 < s_2 \quad \Longrightarrow \quad \Pr(u = 1 \mid s = s_1) \leq \Pr(u = 1 \mid s = s_2), \tag{41}$$

but READY does not require this stronger condition. In finite evaluation samples, local departures from monotonicity may occur; what matters operationally is whether the signal induces a sufficiently ordered risk–coverage relationship to support reliable selective routing.

Importantly, monotonicity is distinct from calibration. A routing signal may provide a useful ordering of cases even when s_i is not numerically equal to the probability of successful execution. Calibration is required only when the numerical value of the signal is interpreted probabilistically; threshold-based routing principally requires discrimination and ordering.

Neither form of signal ordering is required for validity of the p-values or fixed-sequence family-wise error control. It affects which candidates the sequence reaches and therefore the cost of the reported certified policy. Monotone selective success by itself does not guarantee that deployment reliability in Equation 15 improves whenever review increases: under a constant a_h, the cases newly routed to review must also have autonomous success no greater than a_h. If this nested-feasibility condition holds, feasible thresholds form a prefix of the ordered family, so once a threshold fails the reliability requirement, still cheaper thresholds cannot restore it. The sequence then stops either at the first infeasible threshold or earlier through lack of power, so the reported point approaches the globally cheapest feasible threshold as the qualification sample grows. Without that condition, the fixed-sequence procedure remains valid but reports the cheapest member of the certified prefix, and testing non-contiguous candidates would require a different multiple-testing procedure.

E.2 Formal Extension: Tail-Risk Estimation for Risk-Sensitive Qualification

Appendix A.7 formalizes workflow-specific risk as an optional extension to the common READY reliability constraint. This appendix specifies one implementation using conditional value-at-risk (CVaR); it is not instantiated in the empirical studies in this paper. The construction is not required for workflows in which binary deployment success adequately captures the consequences of failure. We use β for the tail quantile level, reserving α for statistical error probability, consistently with the main text.

Let

$$L_i^\pi = L_w(Z_i^\pi) \geq 0 \tag{42}$$

denote the workflow-specific loss incurred on qualification instance i under a fixed oversight policy π. The loss function is defined as part of the workflow specification and may encode, for example, the severity of a failed action, downstream recovery cost, or another consequence not represented adequately by the binary success indicator u_i^π.

For tail quantile level $\beta \in (0, 1)$, the value-at-risk is

$$\mathrm{VaR}_\beta(L^\pi) = \inf\left\{\ell : \Pr(L^\pi \leq \ell) \geq \beta\right\}. \tag{43}$$

CVaR measures the expected loss in the upper tail beyond this quantile. For estimation, we use the equivalent optimization representation

$$\mathrm{CVaR}_\beta(L^\pi) = \min_{\eta \in \mathbb{R}} \left[\eta + \frac{1}{1-\beta}\mathbb{E}\left[(L^\pi - \eta)_+\right]\right], \tag{44}$$

where $(x)_+ = \max(x, 0)$. This form remains well defined for discrete empirical loss distributions and avoids requiring a separate estimator of the tail quantile.

Held-out estimator. The candidate policy family and the risk specification (L_w, β, B_w) are fixed before the qualification set is examined. Given losses

$$L_1^{\widehat{\pi}}, \ldots, L_N^{\widehat{\pi}}, \tag{45}$$

the empirical CVaR estimator is

$$\widehat{\mathrm{CVaR}}_\beta = \min_{\eta \in \mathbb{R}} \left[\eta + \frac{1}{(1-\beta)N}\sum_{i=1}^{N}\left(L_i^{\widehat{\pi}} - \eta\right)_+\right]. \tag{46}$$

For trajectory-invariant policies, the losses may be computed by replaying each candidate policy over saved qualification trajectories. For trajectory-dependent policies, the losses must be obtained from qualification executions or simulations performed under $\widehat{\pi}$ itself.

Statistical qualification. A point estimate of CVaR is not sufficient to establish a deployment-risk constraint. READY therefore constructs a one-sided upper confidence bound

$$\mathrm{CVaR}_{\mathrm{UCB}}^{1-\alpha}\left(\widehat{\pi}\right) \tag{47}$$

at confidence level $1 - \alpha$ and qualifies the policy with respect to the tail-risk constraint only if

$$\mathrm{CVaR}_{\mathrm{UCB}}^{1-\alpha}\left(\widehat{\pi}\right) \leq B_w. \tag{48}$$

In the reference procedure, this upper bound is obtained by the same route as the mean-reliability certificate (Appendix A.5): a p-value for the tail-risk null hypothesis is computed on the held-out

qualification set, candidate policies are tested in a prespecified order, and the bound is the inversion of that test. For CVaR and other quantile-based risk measures, Quantile Risk Control gives distribution-free finite-sample upper bounds (Snell et al., 2023), and Prompt Risk Control applies such bounds to LLM deployment (Zollo et al., 2024). Where no finite-sample p-value is available for the chosen risk functional, a resampling approximation may be substituted, at the cost of only asymptotic validity.

The sampling unit must match the unit of statistical generalization. If multiple stochastic runs are collected for the same task instance, all runs from that instance are carried together as one unit. Likewise, if observations are clustered by patient, customer, account, or another natural unit, the loss is formed at that cluster level rather than treating individual observations as independent.

Uncertain downstream outcomes. If L_i^π depends on an outcome that is not directly observed in the evaluation, for example, the success or cost of a hypothetical human-review path, the corresponding uncertainty must be propagated into the risk estimate. READY may do this by allocating error probability between the review-parameter estimate and the qualification test. A risk quantity computed by substituting a single assumed review-success rate or recovery cost should therefore be reported as conditional on that assumption rather than as an empirically qualified tail-risk estimate.

Pre-specification and tail support. The loss function L_w, tail level β, risk tolerance B_w, and confidence level $1 - \alpha$ must be specified before examining the held-out qualification results. They must not be tuned to make a candidate policy pass qualification.

Tail-risk estimates can be statistically weak when the qualification set contains few observations in the relevant tail. For example, an $\beta = 0.99$ CVaR estimate is informed primarily by roughly the worst one percent of the sampled outcomes. READY therefore reports the effective number of observations contributing to the estimated tail together with the point estimate and confidence bound. If the available qualification sample does not support a sufficiently precise upper bound, the appropriate result is *insufficient evidence for risk qualification*, rather than a relaxed risk threshold.

F Scope, Limitations, and Future Directions

This appendix expands the scope and limitations discussed in Section 6.

READY evaluates and statistically qualifies AI-agent deployment under explicit workflow, oversight, reliability, and cost assumptions; it does not certify an agent universally or capture every organizational consideration in production. Our empirical study focuses on the trajectory-invariant, terminal accept-or-escalate setting on two workflows, a transparent test of the core methodology, while the trajectory-dependent policies of Appendix A.8 require policy-in-the-loop execution or simulation and are left to future validation. Likewise, the risk-sensitive extension requires a prespecified graded deployment loss and adequate tail support; our binary-outcome studies do not empirically validate it. The τ^2 retail study reproduces the deployment pattern on a second workflow with 57 independent qualification tasks; larger task sets would sharpen system-by-system effect estimates. Each certificate is a per-system, per-target statement at level $\alpha = 0.05$; simultaneous control across the reporting grid would require a multiplicity correction over targets and systems, at the price of more review (Appendix A.5). Reported review burden is sample-specific, and Appendix C.3 shows its sensitivity to the evaluation partition and sample size. READY also separates measured evidence from deployment assumptions: when the human-review path is not executed, its success rate, latency, and cost are supplied as assumptions the profile is conditional on, and future evaluations should measure reviewer effectiveness directly. Finally, a qualification claim applies to the population and versioned configuration under which evidence was collected and does not guarantee the same reliability under distribution shift, so post-deployment monitoring and requalification remain important. These bounds define the natural extensions, namely risk-sensitive qualification, trajectory-dependent oversight, direct measurement of human-review effectiveness, and validation on more workflow types, all supported by the open interface of Section 3.1 without changing the qualification objective.

G Extended Related Work

This appendix expands the positioning in Section 2, retaining the full citation record for each of the six families READY draws on.

G.1 Capability and Work-Product Benchmarks

Evaluation of frontier models has progressed from static knowledge exams to executable professional work. Closed-ended exams such as Humanity's Last Exam (Center for AI Safety et al., 2026) measure expert knowledge at the frontier of human difficulty, but score isolated answers rather than an agent's ability to carry out a task or be relied upon in operation. Agentic benchmarks instead evaluate whether a system can *do* the work: Agents' Last Exam (Sun et al., 2026) targets authentic, long-horizon computer-use workflows; GDPval (Patwardhan et al., 2026) compares model deliverables against expert work products and estimates time or cost savings; xbench (Chen et al., 2025b) tracks profession-aligned productivity and technology–market fit; and recent work formalizes how to design and report benchmarks for knowledge work (Hua et al., 2026). A large surrounding literature exercises the enterprise and tool-use skills these workflows depend on. General-assistant and web benchmarks include GAIA (Mialon et al., 2024), WebArena and VisualWebArena (Zhou et al., 2024; Koh et al., 2024), and OSWorld (Xie et al., 2024b); enterprise task suites include WorkArena (Drouin et al., 2024), CRMArena and CRMArena-Pro (Huang et al., 2025a; 2026), and Terminal-Bench (Merrill et al., 2026); conversational tool-use with a simulated user is studied by τ-bench and τ^2-bench (Yao et al., 2025; Barres et al., 2026); and retrieval- and browsing-heavy work is covered by BrowseComp (Wei et al., 2025) and deep-research benchmarks (Du et al., 2026; Abaskohi et al., 2026). Recent surveys catalog this rapidly growing space (Yehudai et al., 2026; Mohammadi et al., 2025), and enterprise-grade benchmarks now span software engineering (Jimenez et al., 2024), simulated companies (Xu et al., 2025), workplace tasks (Styles et al., 2024; Trivedi et al., 2024), and enterprise data analytics (Lei et al., 2025), building on general agent benchmarks such as AgentBench (Liu et al., 2024). A parallel line evaluates the safety and trustworthiness of agent behavior beyond task success (Andriushchenko et al., 2025; Levy et al., 2026; Zhang et al., 2024). These evaluations primarily characterize whether the work gets done and how well; Kapoor et al. (2025) show that accuracy-only agent benchmarks can favor needlessly costly systems and advocate jointly reporting cost and accuracy on held-out data, while τ-bench's *pass*^{k} captures consistency across k attempts (Yao et al., 2025). READY consumes rather than replaces this evidence, adding a reliability–oversight–cost qualification layer.

G.2 Outcome, Process, and Evidence-Based Evaluation

Process- and trajectory-oriented evaluations examine intermediate execution behavior in addition to final outcomes, including whether agents follow appropriate procedures, use tools correctly, and satisfy task-specific intermediate requirements (Sun et al., 2025; Chen et al., 2025a). Related work on factuality and grounding evaluates whether generated outputs are supported by available evidence, and recent studies show that conclusions about hallucination or grounding can depend substantially on the evaluation protocol itself (Geigle et al., 2024; Janiak et al., 2025). Other approaches use consistency or model-internal signals to detect unsupported generation (Manakul et al., 2023; Binkowski et al., 2025; Orgad et al., 2025), while work on reasoning faithfulness demonstrates that plausible model explanations need not faithfully reflect the processes that produced an answer (Lanham et al., 2023; Turpin et al., 2023). Broader reviews argue that deployment-oriented assessment must extend beyond aggregate task completion to execution quality, evidence, safety, and other workflow-specific requirements (Kehkashan et al., 2026). READY does not define a universal truthfulness or grounding metric; such properties enter through the workflow evaluator g_w when they matter for success, and any process requirement used for qualification must be supported by observable execution evidence, not unrecorded model reasoning.

G.3 Selective Prediction, Abstention, and Learning to Defer

READY's terminal accept-or-escalate setting is an instance of selective prediction, in which a system answers on a subset of cases and abstains on the remainder, trading coverage against error among accepted cases. This risk–coverage formulation has a long history: the optimal reject rule dates to

Work / family	Primary question	Executable task	Interactive	Reliability	Operating cost	Human oversight	Statistical qual.
READY (ours)	Under what conditions and at what cost can an organization reliably deploy an agent on a specified workflow?	✓	✓	✓	✓	✓	✓
Capability and work-product benchmarks							
Humanity's Last Exam (Center for AI Safety et al., 2026)	Does the model know the answer to closed-ended, frontier-difficulty questions?	×	×	×	×	×	×
Agents' Last Exam (Sun et al., 2026)	Can a generalist computer-use agent complete authentic, long-horizon professional work?	✓	×	×	×	×	×
GDPval (Patwardhan et al., 2026)	Is the model-generated work product as good as an expert's?	✓	×	×	∼	×	×
τ^2-bench (Barres et al., 2026)	Can an agent coordinate with a user and tools to complete a task reliably across trials?	✓	✓	✓	×	×	×
CRMArena-Pro (Huang et al., 2026)	Can an agent perform professional CRM tasks across business functions and interaction types?	✓	∼	×	×	×	×
AI Agents That Matter (Kapoor et al., 2025)	Do agent benchmarks reward accuracy while ignoring cost?	✓	×	×	✓	×	∼
Process, evidence, and trajectory evaluation							
Process / trajectory eval. (Sun et al., 2025; Chen et al., 2025a)	Are the intermediate steps, tool use, and evidence correct, not just the final answer?	✓	×	×	×	×	×
Selective prediction, abstention, and learning to defer							
Learning to defer (Madras et al., 2018; Mozannar & Sontag, 2020)	Should the model predict this case or defer it to a human?	×	×	×	∼	✓	×
Selective risk & conformal abstention (Geifman & El-Yaniv, 2017; Quach et al., 2024)	Can the model abstain so that accepted outputs meet a risk or coverage guarantee?	×	×	✓	×	∼	✓
Distribution-free risk control (Bates et al., 2021; Angelopoulos et al., 2025)	Which members of a prespecified family of decision rules provably keep expected loss below a target?	×	×	✓	×	×	✓
Confidence, uncertainty, and calibration							
Confidence / calibration (Kadavath et al., 2022; Kuhn et al., 2023)	How likely is this output correct, and is that confidence calibrated?	×	×	∼	×	×	×
Cost-aware routing, cascades, and deferral economics							
Cost-aware routing & cascades (Chen et al., 2024b; Ong et al., 2025)	Can we hit a quality bar at lower cost by routing between models?	∼	×	∼	✓	×	×
Human oversight, scalable oversight, and assurance							
Human oversight & assurance (Vaccaro et al., 2024; Mitchell et al., 2019)	Does human review of AI improve outcomes and provide deployment assurance?	×	∼	∼	×	✓	∼

Table 15: **READY as the integration point across six related-work families.** Each row group names a family with a representative work, scored on six axes: an *executable* multi-step agent task; an *interactive* user or counterparty; measured *reliability* against a target; operating *cost*; *human oversight*; and a *statistically qualified* deployment claim on held-out data. ✓ = addressed, ∼ = partial/implicit, × = not addressed. No single family above addresses all six axes, and READY combines them into one deployment-qualification claim. Its human-oversight axis is instantiated here through a declared review-success assumption rather than executed review.

Chow (1970), selective classification was formalized in later work (El-Yaniv & Wiener, 2010), and confidence-based selection was extended to deep networks (Geifman & El-Yaniv, 2017). The reject-option and abstention settings are surveyed by Hendrickx et al. (2024) and, for language models, by Wen et al. (2025), with optimal-strategy characterizations in Franc et al. (2023). A closely related literature studies *learning to defer* (Madras et al., 2018; Mozannar & Sontag, 2020), with subsequent work on calibrated deferral (Verma & Nalisnick, 2022), exact deferral algorithms (Mozannar et al., 2023), human-complementary predictors (Charusaie et al., 2022), and coverage-constrained multi-expert cooperation (Zhang et al., 2026). Distribution-free guarantees for selective decisions are surveyed in Angelopoulos & Bates (2021); Campos et al. (2024). Risk-controlling prediction sets (Bates et al., 2021) and Learn-then-Test (Angelopoulos et al., 2025) certify, with finite-sample validity, which members of a prespecified family of decision rules keep an expected loss below a target. The closest applications certify when to trust a model and when to escalate. Pareto Testing selects low-cost configurations that satisfy several risk constraints at once (Laufer-Goldshtein et al., 2023), Prompt Risk Control certifies prompts for deployment under mean and tail-risk measures (Zollo et al., 2024; Snell et al., 2023), Conformal Alignment certifies which foundation-model outputs can be trusted (Gui et al., 2024), and Trust or Escalate guarantees human agreement for LLM judges that escalate uncertain cases to stronger judges (Jung et al., 2025). READY instantiates this machinery for oversight policies, with the failure of the whole human–AI system, reviewed cases included, as the controlled risk, the independent task as the sampling unit, and review burden and agent cost as the quantities minimized over the certified set.

Recent work extends these ideas to language models: AbstentionBench (Kirichenko et al., 2025) evaluates appropriate abstention on unanswerable questions; conformal methods construct coverage guarantees, including without logit access (Kumar et al., 2023; Su et al., 2024); and uncertainty-aware planners request human assistance when uncertainty is high (Ren et al., 2023a). The risk–coverage curve and its area (AURC), which we use in Section 3.2, come directly from this literature (El-Yaniv & Wiener, 2010; Geifman & El-Yaniv, 2017). READY builds on selective autonomy but changes what is optimized: an abstention or deferral rate alone says nothing about what happens to deferred cases, so READY models the outcome of the declared human-review path, evaluates the reliability of the human–AI system together with its operating cost, certifies a prespecified policy family on held-out evidence, and selects the lowest-cost member of the certified set. It also distinguishes *task-level abstention*—a valid workflow outcome, such as a correct indeterminate verdict in CliniCARE-Bench—from *deployment-level escalation*, the separate decision to route a completed result to human review (Appendix C).

G.4 Confidence, Uncertainty, and Calibration as Routing Signals

Terminal oversight requires an observable signal s_i that ranks completed executions by their likelihood of autonomous success, and many candidate signals exist. Models can self-assess correctness through elicited probabilities such as *P(True)* and *P(IK)* (Kadavath et al., 2022) or token-level self-evaluation (Ren et al., 2023c). Verbalized confidence (Lin et al., 2022; Tian et al., 2023; Ji et al., 2025; Liu et al., 2025a), relative confidence preferences (Shrivastava et al., 2025), semantic uncertainty (Kuhn et al., 2023; Ji et al., 2025), confidence-weighted self-consistency (Taubenfeld et al., 2025), input-clarification ensembles (Hou et al., 2024), functional uncertainty (Niu et al., 2024), and logit-based methods (Ma et al., 2025) offer further ways to estimate uncertainty. Surveys of confidence estimation and uncertainty quantification organize this space (Geng et al., 2024; Shorinwa et al., 2026), as do hallucination surveys (Ji et al., 2023; Huang et al., 2025b); representative signals include semantic entropy (Farquhar et al., 2024), black-box elicitation (Xiong et al., 2024), internal-state detectors (Chen et al., 2024a), and meaning-aware scores (Duan et al., 2024; Bakman et al., 2024), with large-scale comparisons of which are actually discriminative (Vashurin et al., 2025). Selective generation (Ren et al., 2023b) and conformal generation (Quach et al., 2024) apply such signals to the accept-or-defer decision directly. READY does not prescribe a signal; for threshold routing the score need not be a calibrated probability, only order cases so that more selective acceptance preferentially removes higher-risk executions. Calibration (Guo et al., 2017; Xie et al., 2024a; Li et al., 2025), its behavior in language models (Braverman et al., 2020; Jiang et al., 2021; Chen et al., 2023; Tian et al., 2023), and training-time methods for improving it (Kapoor et al., 2024; Stengel-Eskin et al., 2024; Zhao et al., 2023b;a; Shi et al., 2024), together with careful measurement of calibration (Naeini et al., 2015; Nixon et al., 2019), remain relevant when the signal is interpreted probabilistically, but calibration is not a prerequisite for qualification. Signals that require

repeated sampling or extra inference improve discrimination at a cost, which READY folds into the operating-cost quantity $C_{m,w}(\pi)$.

G.5 Cost-Aware Routing, Cascades, and Deferral Economics

Selecting the oversight policy that meets a reliability target at minimum operating cost connects READY to a large literature on cost-aware inference. Model *cascades* invoke a cheap model first and escalate when a confidence signal is low (Farinhas et al., 2025; Dekoninck et al., 2025; Shen et al., 2025): FrugalGPT learns a cascade to hit a target accuracy at reduced cost (Chen et al., 2024b), AutoMix escalates under noisy self-verification (Aggarwal et al., 2024), mixture-of-thought cascades defer only hard reasoning cases (Yue et al., 2024), and EcoAssistant extends the pattern to code-executing agents (Zhang et al., 2023). A parallel line *routes* each query to the cheapest adequate model: RouteLLM learns a router from preference data (Ong et al., 2025), Hybrid LLM routes by predicted difficulty (Ding et al., 2024), and Zooter routes by predicted expertise (Lu et al., 2024), with RouterBench providing a standard cost–quality evaluation (Hu et al., 2024). Recent surveys formalize routing and cascading as a shared performance–cost optimization problem (Varangot-Reille et al., 2026), descending from classical test-time cost-sensitive cascades (Viola & Jones, 2001). READY shares the goal of reducing cost subject to a performance requirement but differs in three deployment-relevant respects: the deferral target is a *human reviewer*, not a stronger model; READY replaces an average cost–quality operating point with a *statistically qualified* reliability floor on held-out data; and the quality bar is a versioned workflow-specific success predicate that may include process and policy requirements. As Sec. 5.4 demonstrates, changing that predicate can change the qualified operating point even with fixed executions and routing signals; an accuracy-only routing analysis would not reveal this deployment consequence. Cascade and routing methods are complementary: their escalation logic can supply the routing signal s_i.

G.6 Human Oversight, Scalable Oversight, and Deployment Assurance

The notion that supervision is a costly, budgeted resource originates in the scalable-oversight literature (Amodei et al., 2016); subsequent work measures whether a human assisted by an unreliable model outperforms either alone (Bowman et al., 2022), studies oversight when the supervised system is stronger than its overseer (Burns et al., 2024), and proposes debate and amplification as mechanisms (Irving et al., 2018; Christiano et al., 2018; Lang et al., 2025). More recent work emphasizes that effective oversight requires meaningful human agency rather than nominal presence (Zhu et al., 2026; Tsamados et al., 2025), while empirical studies show human–AI teaming does not reliably outperform the stronger component and depends on interaction design and reviewer expertise (Tang et al., 2026; Liu et al., 2025b). A meta-analysis of 106 studies finds human–AI combinations on average underperform the better of human or AI alone (Vaccaro et al., 2024), complementary performance requires more than model explanations (Bansal et al., 2021), and overreliance is a central failure mode of human review (Buçinca et al., 2021; Lai et al., 2021); mandated oversight can provide false assurance when reviewers cannot perform the checking function (Green, 2022). Assurance and auditing frameworks—model cards (Mitchell et al., 2019), internal algorithmic auditing (Raji et al., 2020), and socio-technical safety evaluation (Weidinger et al., 2023)—report evaluated performance under intended deployment conditions. Rather than assuming human review improves outcomes, READY treats the review path's success rate as a measured quantity or explicit assumption and qualifies the deployed system on held-out evidence, complementing model cards and audit frameworks with a statistically supported deployment profile.

Evaluation infrastructure. General evaluation harnesses such as Inspect (UK AI Security Institute, 2024), Harbor (Harbor Framework Team, 2026), and AgentBeats (Liu et al., 2026) provide execution and scoring infrastructure. READY adds workflow-specific deployment semantics and statistical qualification to the resulting evidence. Its reference interface separates execution from the success predicate and deployment assumptions (Appendix B), allowing existing benchmarks to support oversight-aware evaluation without redefining their tasks.