KLINIKEBENCH: EVALUATING LANGUAGE MODELS BEYOND DIAGNOSTIC ACCURACY

Anonymous authors
Paper under double-blind review

ABSTRACT

Most clinical benchmarks evaluate language models (LMs) on diagnosis using complete case descriptions. In clinical practice, however, patients present information in different ways, and clinicians must obtain relevant history and determine which examinations are needed before reaching a diagnosis. Diagnostic accuracy alone therefore cannot establish whether an agent gathered essential information or conducted an appropriate clinical assessment. Furthermore, existing benchmarks lack professional clinicians' verification. To address this gap, we introduce KlinikeBench, a benchmark of 333 clinician-authored tasks, each providing an isolated sandbox environment with a virtual patient, clinical tools, and task-specific success criteria. More than 35 clinicians contributed to case authoring and benchmark evaluation. In an empirical study, clinicians gave simulated dialogues higher mean quality ratings than reference conversations, which is adapted from real conversation. In each task, an LM has a fixed budget of turns to communicate with the patient, ask about relevant history, request examinations, follow action constraints, and record a final diagnosis. We score these steps separately as well as together. Across 31 models and seven model families, the best-performing models (e.g., GPT-6-astra and Claude Opus 5) succeed on $< 30\%$ of tasks, even though their diagnosis accuracy reaches 90.7%. Some models benefit from talking with the patient; others diagnose well from a complete chart but perform much worse in conversation. Overall, KlinikeBench provides a testbed for evaluating the full clinical encounter and reveals a substantial gap between diagnostic accuracy and performance in interactive clinical assessment.

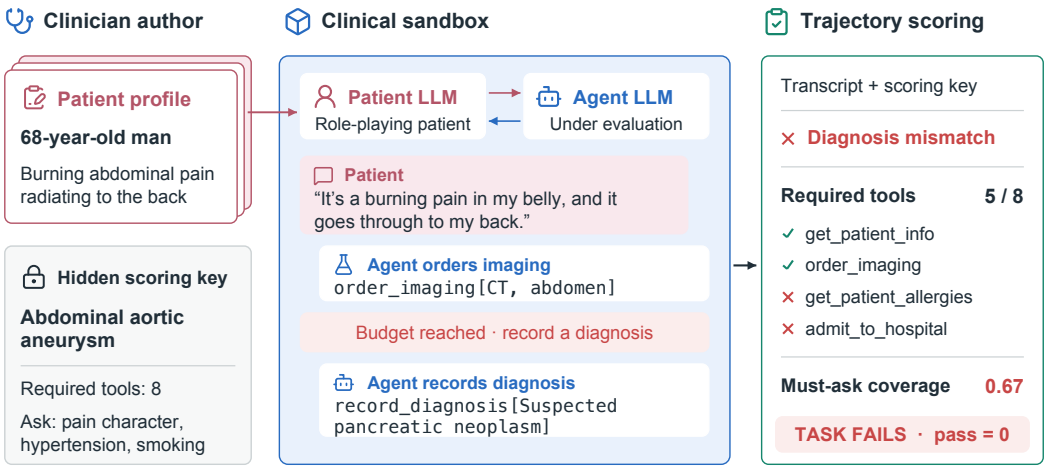


Figure 1: How models are evaluated in KlinikeBench. A clinician authors the patient profile and hidden scoring key. The evaluated agent interacts with an LLM-simulated patient and executes clinical tools in a sandbox. The verifier scores the transcript for diagnosis, required tool use, action constraints, and must-ask coverage. This example fails because the diagnosis is incorrect and required tools and history topics are missed. Tool arguments, simulator messages, and tool checks are abbreviated.

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.

1 INTRODUCTION

Language models (LMs) are increasingly used to support clinical work, from summarizing records to answering patient questions and assisting with diagnostic decisions (Elsevier, 2026; Bedi et al., 2026). Yet most established medical LM benchmarks test knowledge or diagnosis from a self-contained question or case description (Jin et al., 2021; Pal et al., 2022; Jin et al., 2019; Hendrycks et al., 2020). Such tasks provide the information needed to answer the question, leaving untested whether an agent can obtain that information through a clinical encounter. As clinical LM systems become more interactive, evaluation must account for how they gather evidence and carry out an assessment, as well as the conclusions they reach.

Recent clinical agent benchmarks move beyond static questions by exposing models to patient dialogue, tools, or electronic health records (Schmidgall et al., 2026; Jiang et al., 2025; Liu et al., 2026). These environments make information gathering and sequential decisions part of the task. However, introducing interaction does not by itself establish whether an agent has completed an appropriate assessment. Two agents can reach the same correct diagnosis after very different encounters: one may ask about a crucial symptom and order the indicated test, while the other may guess early and omit both. Conversely, an incorrect diagnosis may follow missed history, incomplete clinical assessment, or faulty interpretation of the evidence. Distinguishing these failures requires explicit assessment of the questions asked and actions taken throughout the encounter. Table 1 compares existing evaluations by their scoring, tools, data sources, and clinician verification. More detailed discussion on related work is in Section E.

Table 1: Comparison with medical benchmarks and clinical-agent evaluations. KlinikeBench verification uses a survey of clinician judgments.

Benchmark	**Metric**	**Interaction**	**Tools**	**Source**	**Clinician verification**
MedQA (2021)	MCQA acc.	Single turn	None	Board exams	Exam keys
MMLU-med. (2020)	MCQA acc.	Single turn	None	Exam material	Source answers
MedMCQA (2022)	MCQA acc.	Single turn	None	Entry exams	Expert questions
HealthBench Pro. (2026)	Rubric	Next reply	Varies	Clinical chats	$\geq$3 raters/item
AgentClinic (2026)	Diag. acc.	20 turns	Tests + aids	Clinical cases	Dialogue ratings
MedAgentBench (2025)	Task success	Multi-step	FHIR APIs	EHRs	Task authoring
DeepRare (2026)	Recall@k	Ranked list	$>$40	Rare diseases	Reference review
KlinikeBench (ours)	**pass@1**	**32 turns**	**140**	**Clinicians**	**Clinician survey**

In clinical practice, patients present information incrementally and may not volunteer relevant history. Clinicians must decide what to ask, which examinations to request, and when they have enough evidence to make a diagnosis. Inspired by this workflow, we introduce **KlinikeBench**, a benchmark of 333 clinician-authored tasks that evaluate the full clinical encounter. As illustrated in Figure 1, each task combines a patient profile, an interactive clinical sandbox, and a hidden scoring key. An LLM role-plays the patient, while the evaluated agent elicits history, executes clinical tools, and records a diagnosis within a fixed interaction budget. Tool handlers retrieve information, update patient state, and serve case-specific clinical test results. The verifier assesses the resulting transcript against the clinician-authored requirements, including diagnostic correctness, required tool use, action constraints, and must-ask topics.

To ensure the robustness of each task, KlinikeBench adopts clinician-led task construction. Furthermore, a data annotation platform is developed for clinicians to write patient personas, test the simulated encounters, and revise task drafts (Figure 2). In a separate dialogue-quality study, 30 clinicians gave virtual-patient conversations higher mean ratings than MTS-Dialog (Abacha et al., 2023), a dataset of realistic patient-clinician conversation (Section 4.1), confirming the quality of authored tasks. In terms of scoring criteria, our evaluation makes individual omissions visible: in the example in Figure 1, the agent requests abdominal imaging but records an incorrect diagnosis, completes only five of eight required tool uses, and covers 67% of the must-ask topics. Reporting these components separately identifies the deficiencies behind a failed encounter; requiring all criteria to pass measures whether the agent completes the task as a whole.

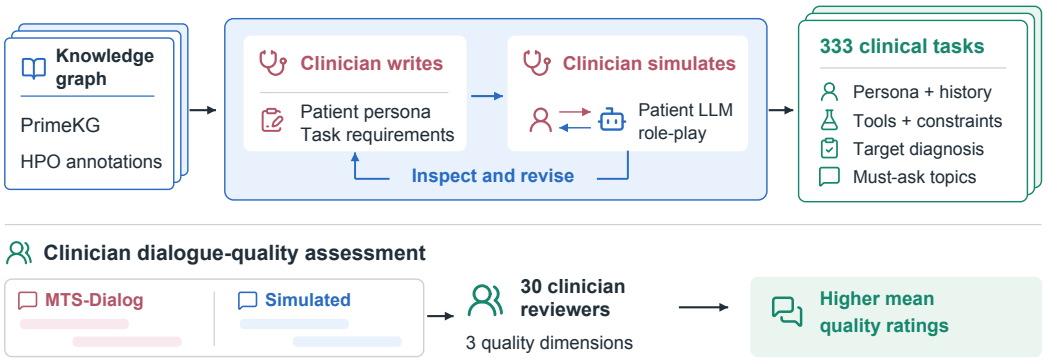


Figure 2: Clinicians author and revise tasks through live simulation. In a separate assessment, virtual-patient dialogues receive higher mean quality ratings than references (shown in Table 3).

Our contributions are threefold. **1) A clinician-authored interactive benchmark.** KlinikeBench provides 333 tasks pairing simulated patients and executable clinical tools with clinician-specified assessment requirements, supported by a platform for authoring and revising encounters. **2) An encounter-level evaluation protocol.** Strict `pass@1` requires correct diagnosis, complete clinical actions, and full must-ask coverage; component scores expose omissions that diagnostic accuracy alone can hide. **3) An empirical study of clinical-agent performance.** We evaluate 31 models across seven families: diagnostic accuracy reaches 90.7%, but the best strict `pass@1` is only 29.4%. We further examine interaction budgets, single- versus multi-turn diagnosis, patient and judge substitutions, clinician ratings of patient-dialogue quality, and supervised fine-tuning and reinforcement learning of Qwen3-4B. Together, these evaluations generate more than 13,000 agent trajectories, providing a resource for studying and developing LMs for disease diagnosis.

2 KlinikeBench

KlinikeBench is a benchmark of clinician-authored clinical encounters, each with a simulated patient, clinical tools, and evaluation criteria. The task is to question the patient, gather evidence using the tools, and record a correct diagnosis while satisfying the case-specific requirements (Figure 1).

2.1 Benchmark construction

A group of clinicians[1] authored the 333 tasks in KlinikeBench through the workflow shown in Figure 2. Our annotation platform supports autocompletion, case archiving, and live patient–clinician simulation for iterative revision (Appendix A). The following formalizes the task definition and the task authoring process. Because this work does not involve designing a custom evaluation harness, we use the terms agent and LM interchangeably throughout the remainder of the paper.

Disease-grounded case authoring. Clinicians select a target disease and associated findings, then construct the presentation from which the agent must infer the diagnosis. PrimeKG and HPO annotations supply disease–phenotype associations (Chandak et al., 2023; Köhler et al., 2021), supplemented by public case reports (Rose et al., 2025). For graph-derived cases, let $\mathcal{G} = (\mathcal{D} \cup \Phi, \mathcal{E})$ contain disease and phenotype nodes linked by HAS_PHENOTYPE edges. The candidate findings for disease d are

$$\mathcal{N}_\Phi(d) = \{\phi \in \Phi : (d, \text{HAS_PHENOTYPE}, \phi) \in \mathcal{E}\}. \tag{1}$$

An authored task is

$$t = (d, \Phi^+, \Phi^-, \pi, \mathcal{R}, M), \tag{2}$$

where Φ^+ and Φ^- specify present and absent findings, π is the patient persona, $\mathcal{R}$ contains tool-use requirements and action constraints, and M contains must-ask topics. Clinicians select positive

[1]The authoring group comprises 35 clinicians from Canada and China, all with direct patient-care experience. Among the 30 questionnaire respondents, 19 reported more than five years of clinical practice.

findings from $\mathcal{N}_\Phi(d)$, add case-specific details, and explicitly author negative findings with $\Phi^- \cap \Phi^+ = \emptyset$; missing graph edges do not imply absent symptoms. They also specify accepted alternative diagnoses $A(d)$.

LLM assistance and clinician revision. Clinicians write the persona and clinical requirements; the LLM assists with narrative refinement and structured formatting. Clinicians then inspect live patient–agent simulations and revise inconsistent responses, inappropriate disclosure, or infeasible requirements. Only approved tasks enter the released corpus $\mathcal{T}^\star$. The persona drives patient simulation, while the diagnosis and scoring criteria remain hidden from the evaluated agent.

Dialogue-quality assessment. Thirty clinicians assessed 50 conversations for progressive history disclosure, emotional appropriateness, and human-likeness using five-point scales. We describe the review protocol and report the ratings in Sections 3.5 and 4.1.

2.2 Task format and evaluation

Task and encounter workflow. Each task combines a clinical briefing, patient simulator, initial clinical state, callable tools, and hidden scoring criteria (Figure 1). The agent receives the briefing and tool definitions, then questions the patient and gathers clinical information within 32 turns, including tool calls. The simulator responds from the authored persona and visible dialogue. Across the corpus, 140 tool names support record retrieval, state updates such as test orders and referrals, and access to cached, case-specific study findings. Missing information may yield unavailable responses or default values; Appendix B details the tool interface and execution behavior. The agent ends the encounter by calling `record_diagnosis`, or is prompted to do so when the turn limit is reached. The resulting dialogue and tool trace are passed to the verifier.

Data split. We use all 333 tasks for the main model evaluation. To facilitate the usage to evaluate training algorithms in rare cases, we also provide a split of 283 training tasks and 50 held-out test tasks, with performance reported on the test set.

Evaluation metrics. For task i, the binary diagnosis gate D_i checks whether the final diagnosis matches the reference or an accepted alternative. The binary required-tool gate W_i passes only when all required tool calls are completed, and A_i checks argument-level action constraints. Must-ask coverage H_i is the fraction of must-ask topics explicitly raised by the agent, assessed by a fixed LLM judge. **Strict `pass@1`** is the fraction of tasks passing all four criteria with one rollout per task:

$$s_i = D_i\, W_i\, A_i\, \mathbf{1}[H_i = 1], \qquad \text{pass@1} = \frac{1}{N}\sum_{i=1}^{N} s_i, \tag{3}$$

where N is the number of tasks. We also report **diagnosis accuracy**, $N^{-1}\sum_i D_i$; **required-tool completion**, $N^{-1}\sum_i W_i$; and **must-ask coverage**, $N^{-1}\sum_i H_i$. All metrics are percentages: required-tool completion counts tasks with all required tools completed, while must-ask coverage averages topic coverage per task. Strict `pass@1` requires all gates and all must-ask topics.

2.3 Dataset statistics

Disease distribution. Figure 3 shows disease groups; Appendix A.3 reports corpus totals and interaction requirements. The corpus contains 304 distinct target-diagnosis labels: 278 occur in one task, 24 occur in two tasks, one occurs in three tasks, and one occurs in four. For visualization, we assign each task to one descriptive clinical group using its target diagnosis. The 226 cases are labeled as *Internal Medicine* include cardiovascular, digestive or liver, endocrine or metabolic, and respiratory cases.

Evaluation Criteria. The tasks require both history-taking and clinical actions: the median case specifies four must-ask topics and five required tools. Table 4 summarizes these requirements and exposed tools. Figure 4 shows the 15 most frequent tools, counted once per task across required and optional tools, excluding final diagnosis recording.

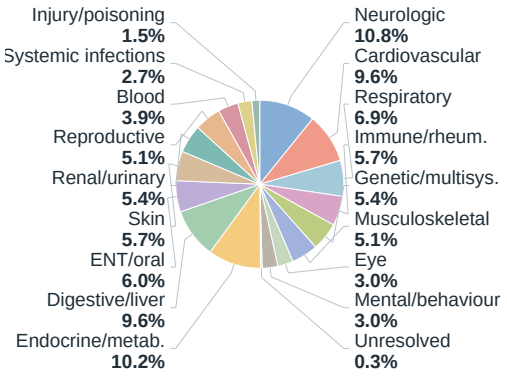


Figure 3: Distribution of diseases spanning 17 clinical subcategories.

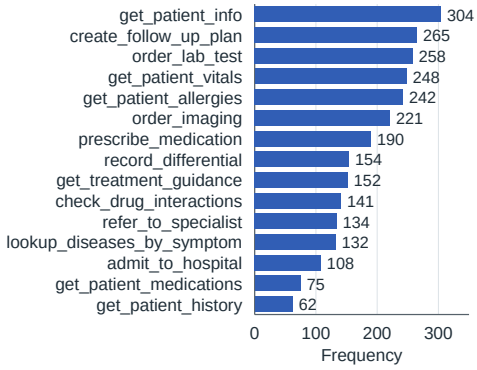


Figure 4: Top 15 tools by task frequency, excluding final diagnosis.

3 Experimental Setup

3.1 Implementation Details

We integrate KlinikeBench into Harbor (Team, 2026), packaging each case as a containerized task with clinical tools and a verifier. The evaluated model uses the ReAct-style `Terminus-2` harness to question the patient, execute tools through the `clinic` command interface, and record a diagnosis. The patient persona and scoring key remain inaccessible to the agent. Within the Harbor evaluation, each model receives the same tasks and tool interface, with one rollout per model–task pair.

3.2 Models

We evaluate 31 models from seven families: Claude, GPT, Gemini, Grok, DeepSeek, Kimi, and GLM (Table 7). Medium reasoning effort is used across all models.

3.3 Patient Simulator and Evaluators

As shown in Figure 5, the ranking of models are relatively stable with different models as the patient simulator and judges. `GPT-5.4-mini` serves as both the patient simulator and history-topic judge in the main evaluation, with separate prompts and contexts. The patient responds from the clinician-authored persona and encounter history. The verifier checks diagnosis, required tool use, and action constraints against the hidden scoring criteria. The history judge evaluates each must-ask topic from the dialogue and returns a binary decision: the agent must explicitly ask about the topic; information volunteered by the patient alone does not count. Patient simulation uses temperature 0.7 and history judging uses temperature 0. We report strict `pass@1` and the component scores defined in Equation (3).

3.4 Training Setup

We train Qwen3-4B (Yang et al., 2025) using the benchmark's Harbor environment and `Terminus-2` harness, with 283 training tasks and 50 held-out test tasks. We first apply supervised fine-tuning on 1,235 successful frontier-model trajectories from the train split, then compare two GRPO runs (Shao et al., 2024) initialized from the same Supervised FineTuning (SFT) checkpoint: one optimizes diagnosis correctness, and the other a weighted reward combining all four gates. Both runs use identical data order and hyperparameters. Appendix C provides the data preparation, optimization settings, and evaluation schedule.

Table 2: Performance of 18 models on 333 tasks; all 31 appear in Table 7. Scores are percentages: diagnosis accuracy, required-tool completion (Tools), and must-ask coverage. Strict `pass@1` requires all criteria (Section 2.2). **Bold**: overall best; underlining: best within families with multiple models. Citations link model reports or cards.

Model	Strict `pass@1`	Diagnosis	Tools	Must-ask
Claude Opus 5 (Anthropic, 2026b)	**29.4%**	**90.7%**	44.1%	89.9%
Claude Opus 5.5 (Anthropic, 2026c)	28.8%	**90.7%**	49.8%	84.4%
Claude Opus 4.8 (Anthropic, 2026a)	16.8%	89.2%	27.9%	85.5%
Claude Sonnet 5 (Anthropic, 2026e)	13.8%	87.4%	24.0%	85.1%
Claude Haiku 4.5 (Anthropic, 2025a)	2.7%	72.1%	8.1%	70.6%
GPT-6-astra (OpenAI, 2026f)	24.3%	87.4%	39.3%	88.0%
GPT-5.6-sol (OpenAI, 2026e)	23.7%	86.2%	36.6%	**90.8%**
GPT-5.4 (OpenAI, 2026b)	23.4%	80.2%	53.2%	73.5%
GPT-4o (Hurst et al., 2024)	3.9%	66.4%	12.9%	79.5%
Gemini-3.8-flash (Google DeepMind, 2026c)	18.9%	80.8%	**53.8%**	76.5%
Gemini-3.5-flash (Google DeepMind, 2026a)	15.6%	75.7%	43.8%	76.6%
Grok-4.7 (xAI, 2026b)	22.2%	83.2%	35.1%	87.5%
Grok-4.6 (xAI, 2026a)	21.9%	81.4%	36.9%	89.0%
Grok-4.3	21.3%	77.8%	45.6%	83.3%
DeepSeek-v4-pro (DeepSeek-AI et al., 2026)	8.1%	29.4%	13.8%	87.8%
DeepSeek-v4-flash (DeepSeek-AI et al., 2026)	3.9%	18.0%	9.3%	81.3%
Kimi-K2.6 (Moonshot AI, 2026)	23.4%	77.2%	48.7%	84.7%
GLM-5.1 (Zeng et al., 2026)	4.8%	24.9%	10.8%	30.8%

3.5 Robustness and validation setup

Judge and patient-simulator sensitivity. We assess how the choice of judge and patient simulator affects results on a 50-task subset using eight clinician models: Claude Opus 5, Claude Opus 5.5, Claude Sonnet 5, GPT-6-astra, GPT-5.6-sol, Grok-4.7, Gemini-3.8-flash, and Gemini-3.7-flash. We compare GPT-5.4-mini, GPT-5.4, Gemini-3.5-flash, and Grok-4.5 as judges, and GPT-5.4-mini, GPT-6-astra, Claude Sonnet 5, and Claude Haiku 4.5 as patient simulators. Figure 5 shows how the clinician-model rankings vary across these choices.

Interaction-budget sensitivity. While the main evaluations use a limit of 32 turns, the sensitivity of the evaluation results on this limit needs to be studied. We evaluate GPT-6-astra, Claude-Opus-5, and Sonnet-5 on a 50-task subset under the default task-dependent step cap, caps of 45 and 60, and no step cap. The step budget includes dialogue and tool calls. With GPT-5.4-mini as patient and judge, we compare strict `pass@1` and component scores on matched tasks (Table 8).

Clinician assessment. To assess the quality of virtual patients, thirty clinicians each rated 50 conversations: 35 conversations between our virtual patient and a clinician and 15 reference conversations from MTS-Dialog (Abacha et al., 2023). MTS-Dialog is a collection of real patient-clinician conversations adopted from real patient records. The 50 conversations are mixed without labels; and then the clinicians are required to assess each conversation from three dimensions: progressive history disclosure, emotional appropriateness, and human-likeness. Each dimension was scored on a 5-point Likert scale (1 = completely unrealistic, 5 = indistinguishable from real patients), yielding 1,500 ratings per dimension and 4,500 ratings in total.

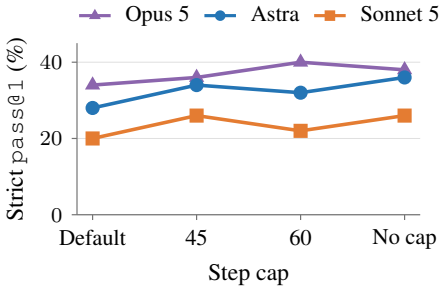
4 Results

We evaluate 31 models across seven families (Table 2; full results in Table 7). Models struggle to fully solve these tasks: Claude Opus 5 performs best at only 29.4% strict `pass@1`. We then examine interaction budgets, dialogue, evaluator sensitivity, patient quality, and small-model training.

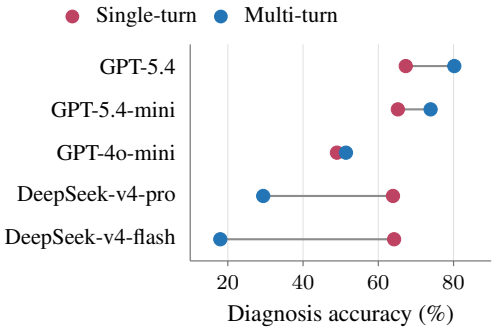
Correct diagnosis does not imply a complete assessment. Claude Opus 5 achieves 90.7% diagnosis accuracy but only 29.4% strict `pass@1` (Table 2). Across all 31 models, mean diagnosis accuracy is 73.6%, compared with 13.4% strict success and 27.9% required-tool completion. GPT-5.3-codex

illustrates this gap: it diagnoses 84.4% of cases correctly, yet completes the required tools on only 8.7% and passes 3.6% overall (Table 7). Different models lead on different components: Gemini-3.8-flash has the highest required-tool completion (53.8%), while GPT-5.6-sol has the highest must-ask coverage (90.8%). Reaching the correct diagnosis therefore does not establish that the necessary history and clinical actions were completed. Appendix D provides the cost–performance comparison.

Pass rates do not improve monotonically with more steps. We examine how the budget of interaction steps, which counts dialogue and tool calls, affects strict success while holding the ten-patient-turn limit fixed. Across the default (32-step), 45-step, 60-step, and uncapped conditions, strict pass@1 is 28/34/32/36% for GPT-6-astra, 34/36/40/38% for Claude Opus 5, and 20/26/22/26% for Claude Sonnet 5. Overall, increasing the length of conversation does not consistently increase success, but there is positive influence on the outcome.



Single-turn vs. multi-turn diagnosis. We compare interactive diagnosis with a single-turn variant that supplies the full intake snapshot, including the chief complaint, demographics, severity, duration, history, medications, allergies, and family history. Both settings score the final diagnosis using the same reference-matching criterion. Across all 333 tasks, GPT-5.4, GPT-5.4-mini, and GPT-4o-mini improve by 12.9, 8.7, and 2.4 percentage points in the interactive setting. DeepSeek-v4-pro and DeepSeek-v4-flash instead decline by 34.5 and 46.2 points. Their similar single-turn scores (63.9% and 64.2%) become 29.4% and 18.0% in conversation. This comparison tests evidence gathering and dialogue control as well as diagnosis, while also changing how information is presented.



4.1 Ablations and Patient-Quality Assessment

Judge substitutions broadly preserve rankings, but can change the leader. Re-scoring fixed trajectories with different must-ask judges broadly preserves clinician-model rankings (Figure 5a). GPT-6-astra remains first, sometimes tied with Claude Opus 5.5, except under Gemini-3.5-flash, which ranks Opus 5.5 first. Broad ordering is therefore more stable than the identity of the leader. Only must-ask coverage is re-evaluated; the other gates remain fixed.

Patient simulation shifts scores but broadly preserves rankings. With GPT-5.4-mini as the patient simulator, GPT-6-astra ranks first. With Claude Haiku 4.5, Claude Opus 5 and Opus 5.5 share the top rank. Despite this change at the top, the overall ordering remains broadly similar. Because patient substitutions require fresh encounters, rank changes may reflect both the simulator choice and rollout variability. Figure 5b shows the rankings, and Table 9 reports the available scores.

Clinicians rate virtual-patient conversations favorably. Thirty clinicians assessed 35 virtual-patient conversations and 15 MTS-Dialog references (Table 3). Virtual-patient mean ratings were 3.69 for history disclosure, 3.50 for emotional appropriateness, and 3.76 for human-likeness, compared with 3.43, 3.31, and 3.45 for the references. Ratings of four or five accounted for 53.2–64.9% of virtual-patient ratings. Emotional appropriateness received the lowest score in both groups. These findings support the ability of our clinician-authored task profiles to produce realistic patient conversations.

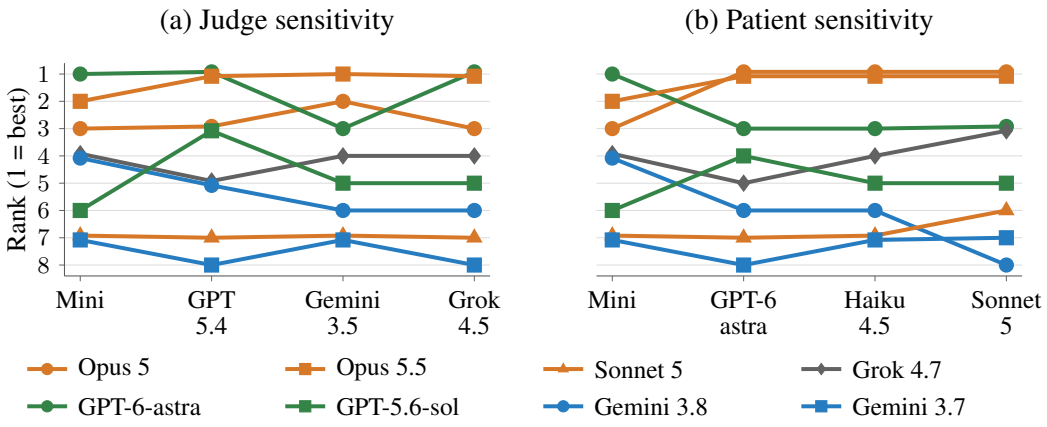


Figure 5: Each point ranks a clinician by strict `pass@1` over 50 tasks under the indicated judge (a) or patient (b). Mini: GPT-5.4-mini; Gemini: Flash variants. Ties are slightly offset; full scores: Table 9.

Table 3: Clinician ratings by conversation source (mean ± SD, scale 1–5). Thirty clinicians provide 1,050 ratings per dimension for 35 virtual-patient conversations and 450 for 15 MTS-Dialog references (Abacha et al., 2023). SD summarizes individual ratings.

Dimension	Virtual patient	MTS-Dialog
Progressive history disclosure	3.69 ± 0.96	3.43 ± 1.05
Emotional appropriateness	3.50 ± 0.96	3.31 ± 1.08
Human-likeness	3.76 ± 1.07	3.45 ± 1.19

4.2 Supervised fine-tuning and reinforcement learning

Standalone sandbox with Harbor enables the scalable RL training. We hereby evaluate Qwen3-4B on 50 held-out tasks after SFT and GRPO with diagnosis-only or multi-gate rewards (Equation (4)). Both RL runs share the SFT initialization and use Claude Opus 4.6 as patient and judge. We report the first 30 steps; Figure 11 in the Appendix shows extended results and training details.

Successful trajectories improve assessment beyond diagnosis. SFT raises strict `pass@1` from 0% to 8%, diagnosis accuracy from 26% to 39%, required-tool completion from 3% to 33%, and must-ask coverage from 39.7% to 77.5% (Appendix D, Table 10). The strict `pass@1` gain is 8 percentage points, with a 95% confidence interval from 2 to 15 percentage points. The intervals for the other three gains also lie above zero, supporting improvements in diagnosis, history-taking, and tool use.

RL gains remain uncertain. We found three targeted metrics have different levels of difficulty to optimise. While SFT improves the model's perform across three metrics. RL could improve diagnosis,

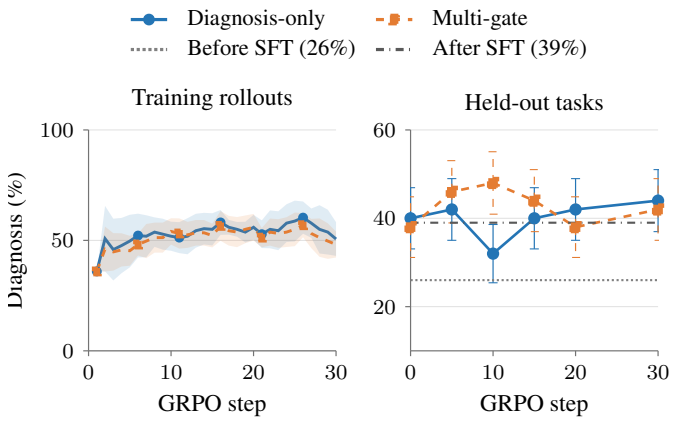


Figure 6: Qwen3-4B diagnosis during GRPO. Left: five-step training means with ±SD bands. Right: held-out accuracy with ±SE bars (50 tasks).

but struggles to improve the must-ask coverage. Figure 6 separates training diagnosis rates from held-out performance. At step 30, diagnosis-only and multi-gate RL reach 44% and 42% diagnosis accuracy, respectively, compared with 39% after SFT. Their strict `pass@1` scores are 10% and 6%, compared with 8% after SFT; required-tool completion is 32% in both runs. This is 2 percentage points above SFT for diagnosis-only RL and 2 points below it for multi-gate RL. However, both 95% confidence intervals include possible gains and losses relative to SFT (Appendix D). These early results do not establish that RL improves full-encounter success or that either reward is better.

5 Qualitative Analysis and Discussions

Overall, KlinikeBench serves as a strong proxy for evaluating LMs' capability in clinical settings, requiring not only correct diagnosis, but proper dialogue and clinical tool usage. Across 31 models, diagnostic accuracy reaches 90.7%, but strict success remains below 30%; component scores expose incomplete history-taking and clinical actions. Patient and judge substitutions show that evaluation choices have limited effects on rankings. Clinician ratings provide descriptive evidence of dialogue quality, and Qwen3-4B fine-tuning improves assessment, while early RL gains remain uncertain, motivating richer evaluation alongside explicit completion criteria. Despite the overall robustness, there are still challenges, motivating the future research on model alignment and benchmark design.

Passing the gates does not guarantee a coherent encounter. Figure 7 compares GPT-6-astra and Qwen3-4B after SFT and GRPO on the same held-out ventricular-tachycardia case, with Claude Opus 4.6 as the patient. Both trajectories pass the diagnosis and required-tool gates, cover all five must-ask topics, and receive a passing outcome. Their dialogue quality nevertheless differs: GPT-6-astra asks focused questions, explains its admission recommendation, and voluntarily records a specific diagnosis with low expressed confidence. Qwen covers the required topics but later produces garbled and incoherent statements, then records a broad diagnosis with high expressed confidence when forced by the step cap. This case shows that passing the gates does not establish dialogue coherence, diagnostic precision, or appropriate confidence.

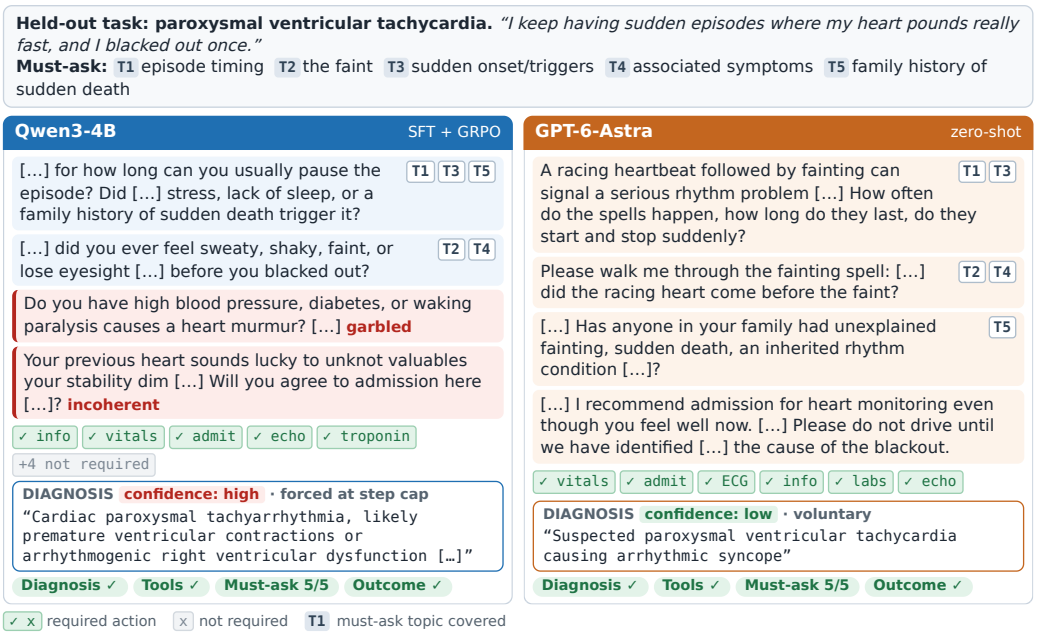


Figure 7: Two passing trajectories on the same held-out case. Qwen3-4B (SFT + GRPO) and GPT-6-astra (zero-shot) satisfy the reported gates but differ in dialogue coherence and diagnosis recording. Excerpts are verbatim with omissions; the patient is Claude Opus 4.6.

AI USE STATEMENT

We use LLMs to assist task narrative refinement and structured formatting, author cached study findings, simulate patients, judge must-ask coverage, and act as the evaluated clinicians (Sections 2.1 and 3). Clinicians review and revise task specifications before inclusion. Successful model-generated trajectories support supervised fine-tuning, and simulated encounters provide reinforcement-learning feedback (Section 3.4). AI assistants also supported manuscript drafting and revision, literature discovery and synthesis, reference formatting, code editing, figure and table preparation, spreadsheet organization, consistency checks, and interpretation of supplied evaluation results. The authors are responsible for the final text, claims, analyses, and artifacts.

ETHICS STATEMENT

KlinikeBench evaluates simulated encounters; tool calls update a sandbox and do not deliver clinical care. Clinicians author and revise the cases, and 30 clinicians assess dialogue quality. The MTS-Dialog references are scripted medical conversations, rather than recordings of actual patient encounters. These ratings and benchmark scores do not establish clinical safety or readiness for deployment. Models may produce incorrect diagnoses, unsafe recommendations, incoherent dialogue, or biased responses, including in trajectories that pass the scoring gates. The clinician-defined requirements and selected cases may also reflect practice-specific assumptions and do not establish performance across demographic groups or care settings. Distribution of tasks derived from public resources and case reports must respect source attribution, licenses, and privacy constraints.

REPRODUCIBILITY STATEMENT

We will release the full 333-task dataset in Harbor benchmark format. Sections 2 and 3 specify task construction, scoring, model settings, and evaluation protocols. Appendix A documents the authoring platform and corpus statistics; Appendix B describes tool execution. Appendix C gives the training/test split, SFT data preparation, reward definitions, hyperparameters, and evaluation schedule. Appendix D provides full model results, ablations, training scores, and uncertainty estimates. Patient and judge substitutions quantify sensitivity to these model choices; API model updates and stochastic generation may limit exact replication of individual trajectories.

REFERENCES

Asma Ben Abacha, Wen-wai Yim, Yadan Fan, and Thomas Lin. An empirical study of clinical note generation from doctor-patient encounters. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2291–2302, 2023.

Anthropic. Claude Haiku 4.5 System Card, 2025a. URL https://www-cdn.anthropic.com/7aad69bf12627d42234e01ee7c36305dc2f6a970.pdf.

Anthropic. Claude Sonnet 4.5 System Card, 2025b. URL https://www-cdn.anthropic.com/963373e433e489a87a10c823c52a0a013e9172dd.pdf.

Anthropic. Claude Opus 4.8 System Card, 2026a. URL https://anthropic.com/claude-opus-4-8-system-card.

Anthropic. Claude Opus 5 System Card, 2026b. URL https://anthropic.com/claude-opus-5-system-card.

Anthropic. Claude Opus 5.5 System Card, 2026c. URL https://www.anthropic.com/claude-opus-5-5-system-card.

Anthropic. Claude Sonnet 4.6 System Card, 2026d. URL https://anthropic.com/claude-sonnet-4-6-system-card.

Anthropic. Claude Sonnet 5 System Card, 2026e. URL https://anthropic.com/claude-sonnet-5-system-card.

Rahul K Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, et al. Healthbench: Evaluating large language models towards improved human health, 2025. *URL https://arxiv.org/abs/2505.08775*, 2025.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-Bench: Evaluating conversational agents in a dual-control environment. *arXiv preprint arXiv:2506.07982*, 2025.

Suhana Bedi, Hejie Cui, Miguel Fuentes, Alyssa Unell, Michael Wornow, Juan M Banda, Nikesh Kotecha, Timothy Keyes, Yifan Mai, Mert Oez, et al. Holistic evaluation of large language models for medical tasks with medhelm. *Nature medicine*, 32(3):943–951, 2026.

Shiyi Cao, Sumanth Hegde, Dacheng Li, Tyler Griggs, Shu Liu, Eric Tang, Jiayi Pan, Xingyao Wang, Akshay Malik, Graham Neubig, et al. Skyrl-v0: Train real-world long-horizon agents via reinforcement learning. 2025. *URL https://novasky-ai. notion. site/skyrl-v0*, 2025.

Payal Chandak, Kexin Huang, and Marinka Zitnik. Building a knowledge graph to enable precision medicine. *Scientific data*, 10(1):67, 2023.

Anyi Xu DeepSeek-AI, Bangcai Lin, Bing Xue, Bingxuan Wang, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *https://huggingface. co/deepseek-ai/DeepSeek-V4-Pro/blob/main/DeepSeek_V4. pdf*, 2026.

Elsevier. Clinician of the future 2026. Report Webpage, 2026. URL https://www.elsevier.com/insights/clinician-of-the-future/2026. Accessed: 2026-09-26.

Google DeepMind. Gemini 3.5 Flash Model Card, 2026a. URL https://deepmind.google/models/model-cards/gemini-3-5-flash/. Accessed: 2026-09-26.

Google DeepMind. Gemini 3.7 Flash Model Card, 2026b. URL https://deepmind.google/models/model-cards/gemini-3-7-flash/. Accessed: 2026-09-26.

Google DeepMind. Gemini 3.8 Flash Model Card, 2026c. URL https://deepmind.google/models/model-cards/gemini-3-8-flash/. Accessed: 2026-09-26.

Yikun Han, Joey Chan, Jingyuan Chen, Mengting Ai, Simo Du, and Yue Guo. Medconceal: A benchmark for clinical hidden-concern reasoning under partial observability. *arXiv preprint arXiv:2604.08788*, 2026.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Rebecca Soskin Hicks, Mikhail Trofimov, Dominick Lim, Rahul K Arora, Foivos Tsimpourlas, Preston Bowman, Michael Sharman, Chi Tong, Kavin Karthik, Arnav Dugar, et al. Healthbench professional: Evaluating large language models on real clinician chats. *arXiv preprint arXiv:2604.27470*, 2026.

Hebin Hu, Renke Dai, Ah-Hwee Tan, and Yilin Kang. Synthesis and evaluation of long-term history-aware medical dialogue. *arXiv preprint arXiv:2605.19766*, 2026.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Yixing Jiang, Kameron C Black, Gloria Geng, Danny Park, James Zou, Andrew Y Ng, and Jonathan H Chen. Medagentbench: a virtual ehr environment to benchmark medical llm agents. *Nejm Ai*, 2(9): AIdbp2500144, 2025.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. Pubmedqa: A dataset for biomedical research question answering. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 2567–2577, 2019.

Sebastian Köhler, Michael Gargano, Nicolas Matentzoglu, Leigh C Carmody, David Lewis-Smith, Nicole A Vasilevsky, Daniel Danis, Ganna Balagura, Gareth Baynam, Amy M Brower, et al. The human phenotype ontology in 2021. *Nucleic acids research*, 49(D1):D1207–D1217, 2021.

Y Liao, Y Meng, Y Wang, H Liu, Y Wang, and Y Wang. Automatic interactive evaluation for large language models with state aware patient simulator. arxiv. *Preprint posted online on July*, 21, 2024.

Ruoqi Liu, Imran Q Mohiuddin, Austin J Schoeffler, Kavita Renduchintala, Ashwin Nayak, Prasantha L Vemu, Shivam C Vedak, Kameron C Black, John L Havlik, Isaac Ogunmola, et al. Physicianbench: Evaluating llm agents in real-world ehr environments. *arXiv preprint arXiv:2605.02240*, 2026.

Moonshot AI. Kimi-K2.6 Model Card, 2026. URL https://huggingface.co/moonshotai/Kimi-K2.6. Accessed: 2026-09-26.

OpenAI. GPT-5.1 Instant and GPT-5.1 Thinking System Card Addendum, 2025. URL https://cdn.openai.com/pdf/4173ec8d-1229-47db-96de-06d87147e07e/5_1_system_card.pdf.

OpenAI. GPT-5.3-Codex System Card, 2026a. URL https://openai.com/index/gpt-5-3-codex-system-card/.

OpenAI. GPT-5.4 Thinking System Card, 2026b. URL https://openai.com/index/gpt-5-4-thinking-system-card/.

OpenAI. GPT-5.4 Thinking System Card: Appendix, GPT-5.4 mini, 2026c. URL https://deploymentsafety.openai.com/gpt-5-4-thinking/appendix-gpt-5.4-mini.

OpenAI. GPT-5.5 System Card, 2026d. URL https://deploymentsafety.openai.com/gpt-5-5.

OpenAI. GPT-5.6 System Card, 2026e. URL https://deploymentsafety.openai.com/gpt-5-6.

OpenAI. GPT-6 Astra System Card, 2026f. URL https://deploymentsafety.openai.com/gpt-6-astra.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pp. 248–260. PMLR, 2022.

Chuhan Qiao, Jianghua Huang, Daxing Zhao, Ziding Liu, Yanjun Shen, Bing Cheng, Wei Lin, and Kai Wu. Medconsultbench: A full-cycle, fine-grained, process-aware benchmark for medical consultation agents. *arXiv preprint arXiv:2601.12661*, 2026.

Daniel Philip Rose, Chia-Chien Hung, Marco Lepri, Israa Alqassem, Kiril Gashteovski, and Carolin Lawrence. Meddxagent: A unified modular agent framework for explainable automatic differential diagnosis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13803–13826, 2025.

Samuel Schmidgall, Rojin Ziaei, Carl Harris, Ji Woong Kim, Eduardo Pontes Reis, Jeffrey Jopling, and Michael Moor. Agentclinic: A multimodal benchmark for tool-using clinical ai agents. *npj Digital Medicine*, 2026. doi: 10.1038/s41746-026-02674-7.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

Harbor Framework Team. Harbor: A framework for evaluating and optimizing agents and models in container environments. *Zenodo*, 2026.

Tao Tu, Anil Palepu, Mike Schaekermann, Khaled Saab, Jan Freyberg, Ryutaro Tanno, Amy Wang, Brenna Li, Mohamed Amin, Nenad Tomasev, et al. Towards conversational diagnostic ai. *arXiv preprint arXiv:2401.05654*, 2024.

xAI. ModelCard: Grok 4.6, 2026a. URL https://media.x.ai/v1/website/card-4p6-4cd2dc57.pdf. Accessed: 2026-09-26.

xAI. ModelCard: Grok 4.7, 2026b. URL https://media.x.ai/v1/website/card4p7-3a96f40b.pdf. Accessed: 2026-09-26.

Jun Xiang, Zhijie Bao, Rong Hu, Kaizhou Qin, Wei Chen, and Zhongyu Wei. Obgynlongbench: Revealing the evidence-to-ehr gap in longitudinal ehr decision-making. *arXiv preprint arXiv:2609.07601*, 2026.

Zihan Xu, Yanzhen Chen, Xiaocheng Zhang, Zhiting Fan, Weiqi Zhai, Hongxia Xu, and Zuozhu Liu. Longmedbench: Benchmarking medical agents for long-horizon clinical decision-making. *arXiv preprint arXiv:2607.09322*, 2026.

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

Lin Yang, Yuancheng Yang, Xu Wang, Changkun Liu, and Haihua Yang. Medmt-bench: Can llms memorize and understand long multi-turn conversations in medical scenarios? *arXiv preprint arXiv:2603.23519*, 2026.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-Bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024.

Jia Yu, Zilong Wang, Xinyang Jiang, Dongsheng Li, and Shuo Wang. Medcua-bench: A screenshot-only benchmark for clinical computer-use agents. *arXiv preprint arXiv:2606.03203*, 2026.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Weizhi Zhang, Zechen Li, Hamid Palangi, Ben Graef, A Ali Heydari, Simon A Lee, Salman Rahman, Ray Luo, Zeinab Esmaeilpour, Erik Schenck, et al. Rubricstree: Scalable and evolving open-ended evaluation of personal health agents across health memory and medical skills. *arXiv preprint arXiv:2606.18203*, 2026a.

Zeyu Zhang, Ziqing Wang, and Kaize Ding. Medlocomo: A long-context multi-session medical dialogue benchmark for large language models. *arXiv preprint arXiv:2607.22566*, 2026b.

Weike Zhao, Chaoyi Wu, Yanjie Fan, Pengcheng Qiu, Xiaoman Zhang, Yuze Sun, Xiao Zhou, Shuju Zhang, Yu Peng, Yanfeng Wang, et al. An agentic system for rare disease diagnosis with traceable reasoning. *Nature*, 651(8106):775–784, 2026.

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

APPENDIX CONTENTS

A Clinician annotation platform . . . 16
A.1 Case authoring and LLM assistance . . . 16
A.2 Live simulation, inspection, and revision . . . 16
A.3 Released corpus statistics . . . 17
B Clinical tools . . . 18
B.1 Interface and task-specific availability . . . 18
B.2 State updates and returned findings . . . 18
B.3 Execution traces and scoring . . . 19
C Training details . . . 20
D Additional result details . . . 21
D.1 Complete model results . . . 21
D.2 Cost and performance . . . 22
D.3 Interaction-budget scores . . . 22
D.4 Full sensitivity results . . . 22
D.5 Uncertainty in strict success . . . 22
D.6 Extended GRPO gate trajectories . . . 23
E Related work . . . 25

A CLINICIAN ANNOTATION PLATFORM

The annotation platform supports the construction and revision of the interactive cases described in Section 2.1. It brings disease selection, task editing, live simulation, and trajectory review into a shared workspace. Clinicians author the patient persona and clinical requirements, inspect simulated encounters, and decide which changes are needed before approving a case. The LLM assists with drafting and structured formatting. Figures 8 and 9 show the authoring and simulation interfaces.

A.1 CASE AUTHORING AND LLM ASSISTANCE

The task generator starts from a target disease and associated findings (Figure 8). In the illustrated PrimeKG view, clinicians can search the disease index and select a condition. The interface then guides them through symptom selection, the addition of differentiating findings, LLM refinement, and review of the resulting draft. These interface steps implement the first stage of the construction process in Section 2.1: working backwards from a target diagnosis to a patient presentation that an agent must investigate.

Clinicians write the patient's background, history, communication style, and disclosure instructions, together with the accepted diagnosis, required clinical actions, and must-ask topics. LLM assistance helps refine the narrative and translate the authored content into the task schema. Clinicians review and revise the draft before committing it. The task queue and editor provide access to cases as they move through authoring and review; the released benchmark consists of 333 clinician-approved tasks.

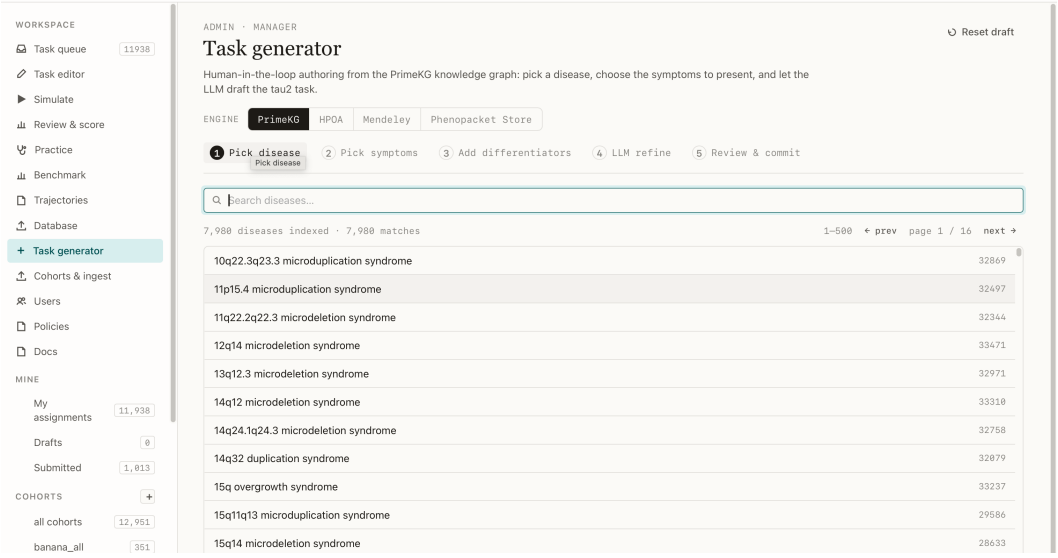


Figure 8: Task-authoring interface. The illustrated PrimeKG view supports disease search and a guided sequence of disease selection, symptom selection, differentiating findings, LLM refinement, and clinician review. Workspace counters describe the authoring workspace.

A.2 LIVE SIMULATION, INSPECTION, AND REVISION

The simulation workspace lets clinicians test how an authored persona behaves during an encounter (Figure 9). It places the case context and reference requirements beside the patient–agent dialogue, with execution counters and a record of tool calls on the right. The reference diagnosis is visible to the clinician reviewing the case; the evaluated agent accesses the clinical briefing and tools, while the scoring key remains hidden. Authors can select the clinician-agent and patient models, adjust the simulation settings, advance either participant one step at a time, or run the full encounter.

The human-annotation panel supports turn-level judgments, including *Correct*, *Partial*, and *Wrong*, together with qualitative tags and free-text notes. Clinicians use the dialogue and tool trace to

identify inconsistent patient responses, inappropriate disclosure, missing clinical information, or task requirements that need revision. They then return to the editor, revise the persona or criteria, and repeat the simulation. These annotations support case development; benchmark scores are computed separately using the metrics in Section 2.2.

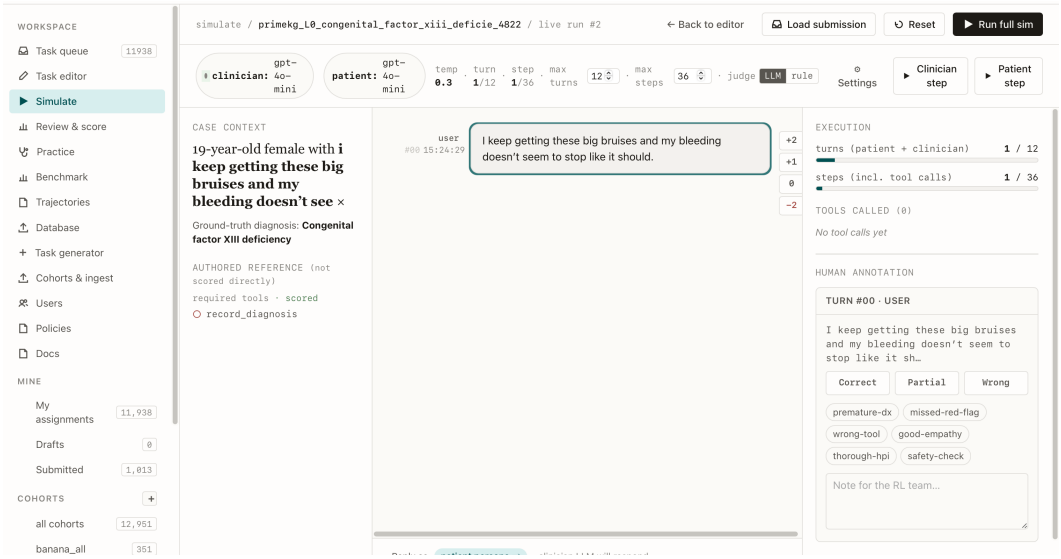


Figure 9: Live-simulation and review interface. Clinicians inspect the case context, dialogue, execution counters, and tool calls, and annotate individual turns before returning to the editor. The displayed settings belong to this authoring session; benchmark evaluation follows Section 3.1.

After clinician approval, the task specification supplies the patient persona, clinical tools, and hidden evaluation criteria used by KlinikeBench. The approved cases are packaged as Harbor tasks for model evaluation, as described in Section 3.1. This separates iterative case development from the fixed task specifications used in model comparisons.

A.3 Released corpus statistics

Table 4 summarizes the released cases and their history-taking and tool requirements.

Table 4: Corpus totals and per-task statistics.

Corpus	**Total**	**Per task**	**Mean**	**Median**	**Min–Max**
Clinical tasks	333	Must-ask topics	4.5	4	0–26
Distinct tool names	140	Required tools	5.8	5	1–19
Clinical subcategories	17	Exposed tools	10.7	10	3–24

B CLINICAL TOOLS

KlinikeBench exposes a task-specific subset of 140 tool names through the Harbor environment described in Section 3.1. Each task provides between 3 and 24 tools (mean 10.7). The catalogue comprises 21 built-in tools, 12 aliases of built-in tools, six additional state-update actions, and 101 named diagnostic studies. Table 5 summarizes their functions; Figure 4 reports their frequency in the task definitions.

B.1 INTERFACE AND TASK-SPECIFIC AVAILABILITY

The agent discovers tools through `clinic tools`, which returns their names, descriptions, and JSON parameter schemas. Most tools require the active `patient_id`; additional fields depend on the action. For example, a laboratory order takes `test_name`, an imaging order takes `imaging_type` and `body_region`, and a prescription takes `drug_name` and `dose`, with optional route and frequency. Task-specific tools use a generic schema accepting a patient identifier and additional descriptive arguments. The following example illustrates the command interface; the identifier must be replaced with that of the current patient.

```
clinic tools
clinic say "When did your symptoms begin?"
clinic call order_lab_test '{"patient_id":"P1","test_name":"CBC"}'
clinic transcript
```

Tool availability is derived from each task's required and optional lists. The diagnosis and differential-recording tools are included by default, while forbidden tool names are excluded. The daemon rejects calls to tools outside the exposed set and arguments that are not JSON objects. Each accepted call and its response are appended to the encounter transcript. `record_diagnosis` stores the final diagnosis, confidence, and reasoning, and ends the encounter; `record_differential` records candidate diagnoses without ending it.

B.2 STATE UPDATES AND RETURNED FINDINGS

Records and clinical actions. Read tools consult the available clinical database or the task's patient persona and initial state. Write tools update the simulated encounter: laboratory and imaging orders receive identifiers, while prescriptions, referrals, admissions, and plans are stored as structured records. These actions do not trigger external clinical services. In particular, a generic test order acknowledges an order; it does not itself generate a new measurement.

Case-specific studies. In the findings-enabled build, named studies retrieve precomputed responses indexed by task identifier and tool name. The cache contains 132 task–study entries spanning 101 tool names: 130 provide results and two mark a study as not indicated. Entries can include narrative findings, measurements, units, and reference ranges. They are authored offline by an LLM and checked for explicit diagnosis leakage before packaging. Tool execution reads the fixed cache without generating findings online. These responses are simulated clinical observations, not measurements from real patients.

Missing information and defaults. Response behavior depends on the handler. A requested laboratory result that is absent may return `not_available`; generic retrieval of ordered tests can instead return a result explicitly marked as mocked. Missing vital signs use fixed fallback values, and some history, medication, or allergy fields use default entries. The drug-interaction tool returns a mock database response. Knowledge-graph query tools and their aliases return `not_available` in the headless runtime. The implementation therefore distinguishes executable tool coverage from the availability of case-specific clinical evidence.

Table 5: Clinical tool catalogue. Counts refer to distinct exposed names across the corpus. The first six rows partition the 21 built-in tools; the final three describe the 119 additional names. Examples are representative, not exhaustive.

Function	Names	Example tools	Execution behavior
Patient records	6	`get_patient_info`, `get_patient_history`, `get_patient_vitals`	Return demographics, history, medications, allergies, family history, or vital signs.
Tests and imaging	3	`order_lab_test`, `get_lab_results`, `order_imaging`	Store orders and retrieve available laboratory results.
Treatment and referral	3	`prescribe_medication`, `refer_to_specialist`, `check_drug_interactions`	Store prescriptions or referrals; return an interaction-check response.
Diagnosis recording	2	`record_diagnosis`, `record_differential`	Store the final diagnosis or differential; only the final diagnosis ends the encounter.
Clinical knowledge	4	`get_treatment_guidance`, `get_disease_symptoms`	Expose knowledge-query interfaces; return unavailable status in the current headless runtime.
Encounter memory	3	`take_note`, `set_working_diagnosis`, `summarize_history_taken`	Store notes and interim hypotheses, or summarize questions already asked.
Aliases	12	`get_patient_labs`, `get_vital_signs`	Delegate to the corresponding built-in handler.
Additional state actions	6	`create_follow_up_plan`, `admit_to_hospital`	Append structured plans, admissions, or referrals to encounter state.
Named diagnostic studies	101	`Electromyography_EMG`	Return cached findings for the current task and study.

B.3 Execution traces and scoring

The runtime records tool names, JSON arguments, and returned responses alongside patient dialogue. The required-tools gate checks whether every required name appears in this trace; it does not independently verify the clinical usefulness of the returned data. Argument-level checks apply the task's configured action constraints. Aliases execute the corresponding handler but retain the invoked name in the trace, so agents must use the tool names exposed for their task. Final diagnosis and must-ask coverage are evaluated separately, as defined in Section 2.2.

A local dispatch audit of the findings-enabled corpus covered all 3,550 exposed task–tool pairs: 3,003 used built-in handlers, 37 used aliases, 378 used additional state actions, and 132 returned cached study entries. No pair reached the generic echo fallback or raised a handler exception under the representative audit arguments. This check establishes handler coverage for the released catalogue; it does not validate every possible argument or the clinical accuracy of every response.

C TRAINING DETAILS

We train Qwen3-4B (Yang et al., 2025) as the clinician using the same Harbor task format and `Terminus-2` harness as the frontier-model evaluation. The findings-enabled corpus is partitioned into 283 training tasks and 50 held-out test tasks (Section 2.2). The test set is a fixed random sample selected with seed 0 and is also used for the ablations. Of its 50 tasks, 42 have a diagnosis that does not occur in the training set. Table 6 summarizes the optimization settings.

Supervised fine-tuning. We collect trajectories from 27 frontier-model configurations on the training tasks, with GPT-5.4-mini as both patient simulator and judge. We retain only trajectories passing all four gates: correct diagnosis, all required tools, full must-ask coverage, and parameterised actions. This yields 1,249 trajectories covering 187 training tasks; removing sequences longer than 20,480 tokens leaves 1,235 trajectories. Each trajectory reconstructs the agent's original multi-turn chat, including the task prompt, assistant JSON replies containing analysis, plans, and shell commands, and the terminal output after each turn. Every assistant reply is prefixed with an empty thinking block, and the loss is computed only on assistant turns. We fine-tune all model parameters for three epochs using FSDP on eight A100 GPUs, with FP32 master weights and BF16 autocast.

Table 6: Training settings for Qwen3-4B. The two RL runs share these settings and differ only in the optimized reward. The SFT patient and judge generate and score the demonstration trajectories.

Setting	SFT	RL (GRPO)
Initialization	Qwen3-4B	SFT checkpoint
Learning rate	10^{-5}	10^{-6}
Schedule	Cosine; 5% warm-up	Constant
Batch	32 sequences (global)	8 tasks $\times$ 8 rollouts
Training duration	3 epochs	315 steps planned; results through 30
Sequence limit	20,480 tokens	32k tokens
Patient and judge	GPT-5.4-mini	Claude Opus 4.6
Rollout temperature	—	1.0
Rollout limits	—	32 agent turns; 15 minutes
Hardware	8 A100 GPUs	4 A100 GPUs per run
Execution	FSDP	Shared vLLM generation / FSDP training

Reinforcement learning. Starting from the SFT checkpoint, we apply GRPO (Shao et al., 2024) using SkyRL (Cao et al., 2025). We do not normalize advantages by the group's standard deviation and use no KL penalty. The loss averages sequence-level sums of token contributions. Each step samples eight training tasks and generates eight rollouts per task at temperature 1.0. Rollouts are capped at 32 agent turns and 15 minutes; we train on the full trajectory up to 32k tokens and filter out overlong rollouts. Generation with vLLM and training with FSDP share four A100 GPUs per run. Claude Opus 4.6 serves as both the patient simulator and must-ask judge during RL.

Reward comparison. Using the gate definitions in Section 2.2, we compare a diagnosis-only reward with a weighted multi-gate reward:

$$r_i^{\text{diag}} = D_i, \qquad r_i^{\text{multi}} = \frac{2D_i + W_i + A_i + H_i}{5}. \tag{4}$$

Both rewards lie in $[0, 1]$. The multi-gate reward uses fractional must-ask coverage H_i, providing credit for partial history-taking. The two runs share the same SFT initialization, data order, and hyperparameters, with 315 steps (nine epochs) planned per run. The reported results cover the first 30 steps. Every rollout logs all gates regardless of the reward being optimized. We evaluate each test task with one rollout before RL training, at steps 5, 10, and 15, and at subsequent recorded checkpoints. The reported early checkpoints include steps 20 and 30. Base and SFT summaries pool the two initial evaluation passes, giving two rollouts per task; RL checkpoints use one rollout per task.

D ADDITIONAL RESULT DETAILS

D.1 COMPLETE MODEL RESULTS

Table 7: Results (%) for 31 models on 333 tasks (metrics: Section 2.2). Strict `pass@1` requires correct diagnosis ∧ complete clinical actions ∧ full must-ask coverage. **Bold**: overall best; underlining: best within families with multiple models.

Model	Strict `pass@1`	Diagnosis	Tools	Must-ask
Claude Opus 5 (Anthropic, 2026b)	**29.4%**	**90.7%**	44.1%	89.9%
Claude Opus 5.5 (Anthropic, 2026c)	28.8%	**90.7%**	49.8%	84.4%
Claude Opus 4.8 (Anthropic, 2026a)	16.8%	89.2%	27.9%	85.5%
Claude Sonnet 5 (Anthropic, 2026e)	13.8%	87.4%	24.0%	85.1%
Claude Sonnet 4.5 (Anthropic, 2025b)	9.9%	82.3%	25.8%	76.7%
Claude Sonnet 4.6 (Anthropic, 2026d)	8.4%	83.8%	22.8%	73.8%
Claude Haiku 4.5 (Anthropic, 2025a)	2.7%	72.1%	8.1%	70.6%
GPT-6-astra (OpenAI, 2026f)	24.3%	87.4%	39.3%	88.0%
GPT-5.6-sol (OpenAI, 2026e)	23.7%	86.2%	36.6%	**90.8%**
GPT-5.4 (OpenAI, 2026b)	23.4%	80.2%	53.2%	73.5%
GPT-5.5 (OpenAI, 2026d)	16.8%	86.2%	30.0%	87.5%
GPT-5.4-mini (OpenAI, 2026c)	15.9%	73.9%	35.1%	80.6%
GPT-5.1 (OpenAI, 2025)	15.3%	73.9%	25.2%	89.6%
GPT-5.6-terra (OpenAI, 2026e)	8.4%	84.7%	19.5%	85.6%
GPT-5.6-luna (OpenAI, 2026e)	7.5%	81.1%	16.2%	84.4%
GPT-4.1	7.5%	74.2%	12.6%	89.1%
GPT-4o (Hurst et al., 2024)	3.9%	66.4%	12.9%	79.5%
GPT-5.3-codex (OpenAI, 2026a)	3.6%	84.4%	8.7%	78.3%
GPT-4.1-mini	2.1%	71.2%	6.9%	83.1%
GPT-4.1-nano	0.9%	55.6%	9.0%	56.9%
GPT-4o-mini	0.6%	51.4%	6.6%	68.6%
Gemini-3.8-flash (Google DeepMind, 2026c)	18.9%	80.8%	**53.8%**	76.5%
Gemini-3.5-flash (Google DeepMind, 2026a)	15.6%	75.7%	43.8%	76.6%
Gemini-3.7-flash (Google DeepMind, 2026b)	10.8%	79.9%	53.5%	67.1%
Grok-4.7 (xAI, 2026b)	22.2%	83.2%	35.1%	87.5%
Grok-4.6 (xAI, 2026a)	21.9%	81.4%	36.9%	89.0%
Grok-4.3	21.3%	77.8%	45.6%	83.3%
DeepSeek-v4-pro (DeepSeek-AI et al., 2026)	8.1%	29.4%	13.8%	87.8%
DeepSeek-v4-flash (DeepSeek-AI et al., 2026)	3.9%	18.0%	9.3%	81.3%
Kimi-K2.6 (Moonshot AI, 2026)	23.4%	77.2%	48.7%	84.7%
GLM-5.1 (Zeng et al., 2026)	4.8%	24.9%	10.8%	30.8%

Model citations point to verified reports or model cards. For GLM-5.1, we cite the GLM-5 technical report recommended by its official model card; Kimi-K2.6 is cited through its version-specific model card.

D.2 Cost and performance

Higher inference cost does not guarantee higher success. Among models with pricing data, Claude Opus 5 achieves the highest success rate (29.4%) at an estimated $0.30 per task, while Opus 5.5 reaches 28.8% at $0.20 (Figure 10). GPT-6-astra costs $1.09 per task with 24.3% success. Gemini-3.8-flash reaches 18.9% at $0.09, while GPT-5.6-luna has the lowest estimated cost ($0.02) and 7.5% success. These estimates account for model-specific token consumption and cached-input rates; they measure clinician inference cost rather than the total cost of running the benchmark.

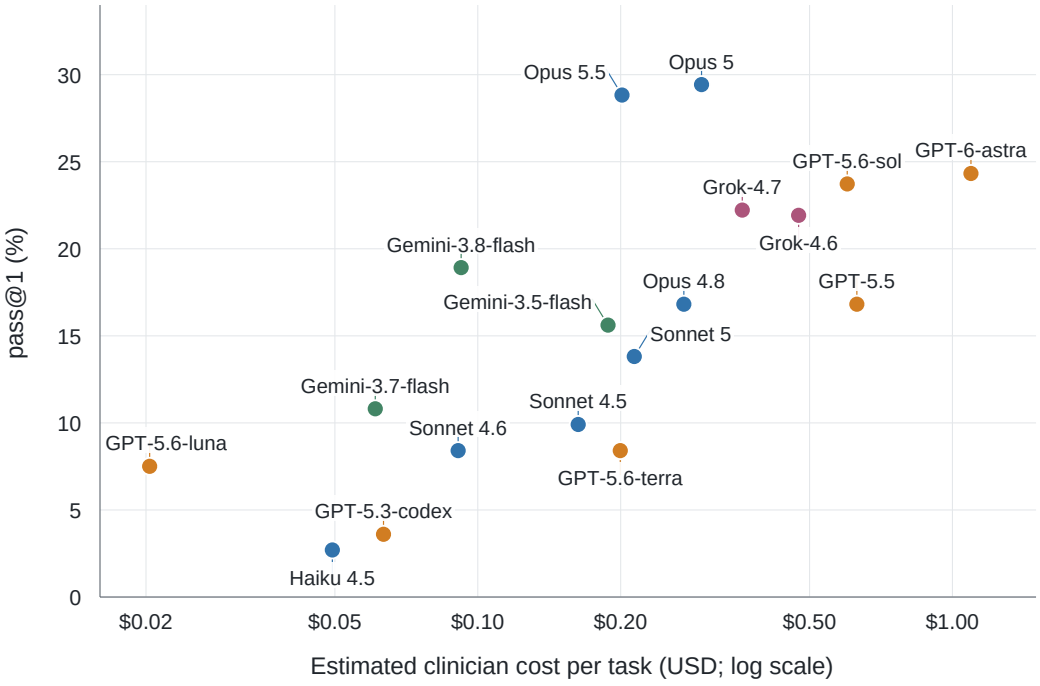


Figure 10: Strict `pass@1` versus estimated clinician cost per task for the 18 models with pricing data. Costs use the workbook's average input, cached-input, and output tokens and listed token rates; patient and judge costs are excluded. The cost axis is logarithmic. Colors indicate model families.

D.3 Interaction-budget scores

D.4 Full sensitivity results

Table 9 includes every patient and judge condition in the results workbook. A dash indicates a score not yet reported (queued or running), not zero. Rank-agreement statistics in the main text use conditions with all eight scores. The supplied patient-ranking figure additionally includes GPT-6-astra and Claude Sonnet 5 conditions whose numerical scores are partially unavailable in this export.

D.5 Qwen3-4B held-out results

Uncertainty in strict success. Paired-bootstrap 95% confidence intervals describe uncertainty in the change in strict `pass@1`, measured in percentage points. SFT improves over the base model by 8 points, with an interval from 2 to 15 points. At RL step 30, diagnosis-only RL changes the SFT score by $+2$ points, with an interval from -7 to $+12$ points; multi-gate RL changes it by -2 points, with an interval from -11 to $+7$ points. Negative values indicate worse performance than SFT and positive values indicate better performance. Both RL intervals span zero, so neither comparison establishes an improvement over SFT.

Table 8: Step-cap ablation for five clinicians on 50 matched tasks. Scores are percentages; Δ is the change in strict `pass@1` from the default run, in percentage points, with the reported paired-bootstrap 95% interval. The ten-patient-turn limit remains fixed.

Model	Step cap	pass@1	Diagnosis	Tools	Must-ask coverage	Δ [95% CI]
GPT-6-astra	Default	28	86	46	88.9	—
	45	34	88	60	90.4	$+6$ $[-6, 20]$
	60	32	86	56	91.2	$+4$ $[-8, 16]$
	None	36	88	52	89.0	$+8$ $[-2, 20]$
Claude Opus 5	Default	34	86	54	89.8	—
	45	36	84	62	90.4	$+2$ $[-10, 16]$
	60	40	88	60	89.9	$+6$ $[-8, 20]$
	None	38	88	66	89.0	$+4$ $[-10, 20]$
Grok-4.7	Default	22	78	46	85.9	—
	45	50	78	80	90.8	$+28$ $[16, 40]$
	60	38	78	72	86.2	$+16$ $[6, 28]$
	None	30	80	66	84.5	$+8$ $[-4, 20]$
Gemini-3.8-flash	Default	20	76	66	74.8	—
	45	22	74	68	76.1	$+2$ $[-12, 16]$
	60	14	72	60	71.0	-6 $[-20, 6]$
	None	20	70	70	74.8	0 $[-12, 12]$
Claude Sonnet 5	Default	20	76	42	87.5	—
	45	26	82	50	84.5	$+6$ $[-8, 20]$
	60	22	82	44	86.8	$+2$ $[-10, 14]$
	None	26	84	48	87.0	$+6$ $[-6, 20]$

Table 9: Strict `pass@1` (%) for all patient and judge substitutions. OP5/OP5.5: Claude Opus 5/5.5; S5: Claude Sonnet 5; G6/G5.6: GPT-6-astra/GPT-5.6-sol; GR4.7: Grok-4.7; GE3.8/GE3.7: Gemini-3.8/3.7-flash. Dashes mark unavailable scores.

Condition	OP5	G6	GR4.7	GE3.8	OP5.5	G5.6	S5	GE3.7
Judge changed; patient fixed to GPT-5.4-mini								
GPT-5.4-mini (original)	24	30	20	20	26	18	8	8
GPT-5.4-mini (repeat)	26	28	24	20	26	18	6	6
GPT-5.4	24	32	20	20	32	24	10	6
Gemini-3.5-flash	22	20	18	12	24	14	6	6
Grok-4.5	22	28	18	14	28	16	8	6
Patient changed; judge fixed to GPT-5.4-mini								
GPT-5.4-mini (baseline)	24	30	20	20	26	18	8	8
Gemini-3.5-flash	32	30	30	18	24	24	16	8.5
Grok-4.5	28	30	34	16	34	32	14	10
GPT-5.4	38	26	30	16	26	28	24	18
Claude Haiku 4.5	34	28	26	18	34	24	16	16
Claude Opus 4.6	28.6	32	32	24	32	26	16	16.3
GPT-5.5	34	30	36	16	28	20	20	16
Gemini-3.8-flash	28	34	32	10	36.2	26	18	16
Grok-4.7	27.7	30	34	14	34.8	16	14	15.2
GPT-6-astra	30	28	24	22	30	26	18	10
Claude Sonnet 5	32	30	30	6	32	24	18	14

D.6 Extended GRPO gate trajectories

Table 10: Qwen3-4B on 50 held-out tasks. Base and SFT pool two rollouts per task; each RL checkpoint uses one. RL rows report step 30. All scores are percentages.

Stage	Strict `pass@1`	Diagnosis	Required tools	Must-ask
Base	0.0%	26.0%	3.0%	39.7%
SFT	8.0%	39.0%	33.0%	77.5%
SFT + RL (diagnosis)	10.0%	44.0%	32.0%	76.8%
SFT + RL (multi-gate)	6.0%	42.0%	32.0%	73.1%

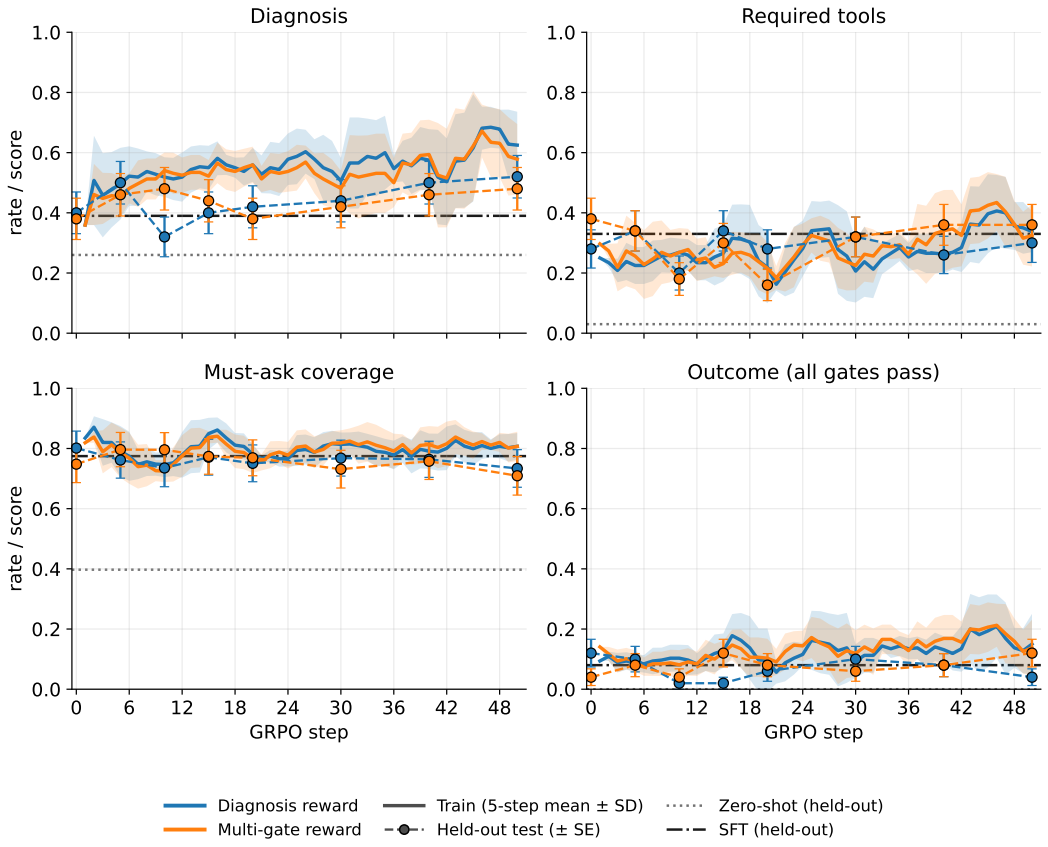


Figure 11: Qwen3-4B gate trajectories through GRPO step 50. The top row shows diagnosis and required-tool completion; the bottom row shows must-ask coverage and strict outcome. Blue and orange denote diagnosis-only and multi-gate rewards. Solid curves with shaded bands show five-step training means ± SD; dashed curves with circles and error bars show held-out results ± SE. Horizontal dotted and dash-dotted lines mark the zero-shot and SFT held-out baselines. Main-text comparisons use checkpoints through step 30.

E RELATED WORK

Medical knowledge and clinical response quality. MedQA, MedMCQA, and PubMedQA evaluate medical knowledge and reasoning from examination questions or biomedical text (Jin et al., 2021; Pal et al., 2022; Jin et al., 2019), providing a foundation for evaluating clinical language models (Singhal et al., 2023). MedHELM covers diverse medical tasks, while HealthBench uses physician-written rubrics to assess responses in health conversations (Bedi et al., 2026; Arora et al., 2025). RubricsTree structures health-agent evaluation around atomic, clinically verifiable criteria (Zhang et al., 2026a). KlinikeBench shares the emphasis on explicit clinical criteria, connecting the final diagnosis to the history elicited and actions performed during an encounter.

Interactive diagnostic dialogue. AMIE studies diagnostic conversations through simulation-based training and a blinded evaluation with patient actors and primary-care physicians (Tu et al., 2024). The State Aware Patient Simulator supports automated evaluation of multi-turn consultations (Liao et al., 2024). AgentClinic combines simulated patient dialogue, multimodal examinations, and tool use, with evaluation extending to patient-centered aspects of interaction (Schmidgall et al., 2026). KlinikeBench follows this interactive setting and pairs encounters with clinician-authored must-ask topics, required tools, and action constraints. Our clinician assessment separately examines history disclosure, emotional appropriateness, and human-likeness of the simulated dialogue.

Process-aware consultation benchmarks. MedConsultBench is especially close to our motivation: it evaluates the consultation cycle from history taking to follow-up, tracks information acquisition using atomic information units, and identifies deficiencies that a correct diagnosis can conceal (Qiao et al., 2026). MedConceal studies elicitation and management of hidden patient concerns under partial observability, with process and outcome evaluation (Han et al., 2026). KlinikeBench complements these efforts by coupling clinician-authored information requirements with executable tools and parameterized action constraints. We report each component and use their conjunction with diagnostic correctness to define strict `pass@1`, distinguishing models with similar diagnostic accuracy by their assessment behavior.

Clinical tools and workflow execution. MedAgentBench evaluates clinician-written tasks in an electronic health record environment with standardized FHIR interfaces (Jiang et al., 2025). PhysicianBench studies longer clinical workflows with physician-reviewed checkpoints and execution-grounded verification (Liu et al., 2026). MedCUA-Bench evaluates screenshot-based interaction with clinical software using task-completion and safety checks (Yu et al., 2026). KlinikeBench connects tool execution to patient questioning within the same encounter: information elicited through dialogue and results obtained through tools jointly support diagnosis. Case-specific requirements additionally check that required clinical actions are completed with appropriate parameters.

Longitudinal records and extended conversations. LongMedBench evaluates reasoning over longitudinal patient records, while ObGynLongBench studies decisions across pregnancy histories and the difficulty of using evidence embedded in electronic health records (Xu et al., 2026; Xiang et al., 2026). MedMT-Bench evaluates instruction following in extended medical dialogues; MediLongChat and MedLoCoMo examine history-aware and multi-session conversation (Yang et al., 2026; Hu et al., 2026; Zhang et al., 2026b). These settings emphasize retaining and using information across long contexts or successive visits. KlinikeBench focuses on a complementary horizon: actively gathering evidence and completing the required questions, examinations, and decisions within one assessment under an interaction budget.

Tool–user interaction and evaluation reliability. Beyond medicine, τ-bench evaluates tool use and policy compliance through simulated-user interaction, and τ^2-Bench requires coordinated actions by agent and user (Yao et al., 2024; Barres et al., 2025). KlinikeBench similarly evaluates communication and tool execution, using clinician-defined completion criteria. Since automated judgments can depend on the evaluator (Zheng et al., 2023), we examine judge substitutions on fixed trajectories and patient substitutions that generate fresh encounters, testing how these choices affect scores and rankings.