MEDCTA: A BENCHMARK FOR CLINICAL TOOL AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

To make clinically grounded decisions, medical AI agents are expected to go beyond simple recognition and be capable of tool retrieval, evidence acquisition, and integration. Existing benchmarks largely evaluate isolated perception or single-turn question answering, and therefore provide limited visibility into failures of planning, tool recruitment, and rollout reliability. We introduce `MedCTA`, a benchmark for evaluating *medical tool agents* on clinician-validated, step-implicit tasks grounded in realistic multimodal clinical inputs, including radiology images, pathology slides, and reports. `MedCTA` comprises of real-world clinical tasks with 1,926 clinician-verified rollouts over **5** diverse deployed tools, and supports process-aware evaluation of tool selection, argument validity, execution stability, trajectory fidelity, and outcome quality. We benchmark **18** open- and closed-source multimodal models and find that even frontier systems remain brittle in multi-step clinical tool use: autonomous rollouts are dominated by protocol failures, premature stopping, and incorrect tool recruitment, while gold-standard tool routing yields large but still incomplete gains. These results show that strong backbone perception does not translate into reliable agentic behavior in clinical settings. `MedCTA` provides a rigorous testbed for auditing, diagnosing, and advancing trustworthy medical AI agents. The dataset and evaluation suite are available at https://anonymous.4open.science/r/MedCTA/README.md.

1 WHY CLINICAL TOOL AGENTS?

Multimodal agentic systems are pushing `LLMs` beyond single-turn prediction toward *interactive problem solving*, where a model plans, invokes tools, and updates its beliefs over multiple steps Xie et al. (2024a); Liu et al. (2024); Wang et al. (2024b); Team (2023a); Contributors (2023). This paradigm has enabled strong assistants for web interaction and document understanding Nakano et al. (2021); Sun et al. (2025), visual analysis Surís et al. (2023); Lu et al. (2023), and software engineering Jimenez et al. (2023); Wu et al. (2023c); Yang et al. (2024). Yet recent agentic benchmarks show that strong final answers can coexist with brittle multi-step behavior: models often fail in tool routing, trajectory maintenance, or reasoning over intermediate evidence Ashraf et al. (2025); Wang et al. (2024b); Mialon et al. (2023). As models become more capable, the limiting factor is increasingly not raw perception, but reliable execution under tool interaction.

These challenges are especially acute in medicine. Clinical decision making is inherently iterative and multimodal: it requires interpreting images, extracting text, measuring findings, aggregating evidence across steps, consulting clinical knowledge, and producing structured conclusions Shen et al. (2017); Chow & Paramesran (2016); Cabral et al. (2024); Sun et al. (2022). Accordingly, agentic methods are beginning to appear in healthcare Duncan & Ayache (2000); Xu et al. (2025a); Kim et al. (2024; 2025b;a); Schmidgall et al. (2024); Jiang et al. (2025a); Li et al. (2024a). But existing evaluations Hu et al. (2024); Jiang et al. (2025a); He et al. (2020) still emphasize dialogue outcomes, EHR navigation, or perception accuracy, with limited support for *multimodal, tool-executable, and traceable* clinical workflows (Table 1). This gap matters in medical settings as a final answer is not enough: users also need process-level accountability about what evidence was used, what was measured, and how the conclusion was reached.

To address this gap, we introduce **`MedCTA`**, a trajectory-aware benchmark and executable evaluation platform for *medical tool agents*, inspired by agentic benchmarks like GAIA Mialon et al. (2023),

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.

Table 1: Comparison with representative medical VQA, medical reasoning, and medical agent benchmarks. MedCTA is the only benchmark in this comparison that combines step-implicit clinical queries with executable, clinician-verified tool trajectories across multimodal medical inputs.

Benchmark	# Modalities	Agentic Tasks	Real-world Assistive Tools	Real-world Clinical Queries	Multimodal Inputs	Deep Reasoning
VQA-RAD Lau et al. (2018)	3	✗	✗	✗	✓	✗
SLAKE Liu et al. (2021)	3	✗	✗	✗	✓	✗
Path-VQA He et al. (2020)	2	✗	✗	✗	✓	✗
VQA-Med Ben Abacha et al. (2019)	5	✗	✗	✗	✓	✗
PMC-VQA Zhang et al. (2024)	2	✗	✗	✗	✓	✗
MMMU-Med Yue et al. (2024)	2	✗	✗	✓	✓	✓
MedQA Jin et al. (2021)	1	✗	✗	✓	✗	✓
PubMedQA Jin et al. (2019)	1	✗	✗	✓	✗	✓
MedAgentBench Jiang et al. (2025a)	1	✓	✗	✓	✗	✓
OmniMedVQA Hu et al. (2024)	12	✗	✗	✗	✓	✗
MedCTA (Ours)	**16**	✓	✓	✓	✓	✓

GTA Wang et al. (2024b), and Agent-X Ashraf et al. (2025). MedCTA contains clinician-validated, step-implicit clinical queries grounded in authentic multimodal medical inputs, including radiology images, pathology slides, figures, reports, and scanned clinical content. Each task must be solved by autonomously selecting from deployed tools for perception, operation, reasoning, and reporting, without being told the tool sequence. We pair every task with a clinician-verified reference trajectory, enabling fine-grained evaluation of tool selection, argument validity, intermediate evidence consistency, and final task success.

Across **18** models, MedCTA reveals a gap between medical competence and reliable medical agency. The best outcome accuracy in the agentic (tool-using) setting is only **31.54%**, while the strongest open model reaches **27.80%**. Under stricter rollout diagnostics, no model achieves non-zero strict trajectory success. In contrast, when provided with gold-standard next-tool routing, while still generating arguments and final answers, diagnostic answer accuracy improves by up to **+35%**. This gap indicates that most performance degradation arises from controller failures rather than a lack of underlying clinical reasoning. The remaining errors are dominated by protocol instability, premature stopping, wrong tool recruitment, and failures in localized evidence grounding. These findings position MedCTA not just as a leaderboard benchmark, but as a diagnostic instrument for auditing and improving clinically grounded agents, helping to steer the field toward more reliable controller design and robust tool use. To summarize:

- We introduce MedCTA, a clinician-validated benchmark for *medical tool agents* built around realistic multimodal clinical tasks with step-implicit queries and executable tool use.
- We curate clinician-verified agentic tasks with 1,926 rollouts across 34 anatomical regions, and define process-aware metrics for tool routing, trajectory fidelity, evidence use, and outcomes.
- We benchmark 18 frontier models and show that current systems remain brittle on multi-step clinical tool use, with large controller gaps, severe under-calling, and persistent failures in long-horizon localized grounding.

2 Related Work

Tool-augmented and trajectory-aware agents. General-purpose agent systems have shown that large language models can act as controllers over external tools and environments, as in LangChain Chase (2022), Auto-GPT Gravitas (2023), and BabyAGI Nakajima (2023). Subsequent work extended this paradigm to web interaction with WebGPT Nakano et al. (2021), WebShop Yao et al. (2022), and WebCPM Qin et al. (2023); API and service orchestration with RESTGPT Song et al. (2023) and APIBank Li et al. (2023c); operating-system and app control with OSWorld Xie et al. (2024b) and AppAgent Zhang et al. (2023); and multimodal tool use with HuggingGPT Shen et al. (2024) and MSAgent Li et al. (2023a). Vision-centric agents such as VisualGPT Wu et al. (2023b), MM-REACT Yang et al. (2023), MLLM Wang et al. (2024a), and LLaVA-Plus Liu et al. (2024) further demonstrate structured reasoning over visual inputs. These works establish the promise of tool-augmented reasoning, but they are largely open-domain and are evaluated by end-task success.

Multimodal medical benchmarks and models. Progress in medical multimodal learning has been driven primarily by perception- and QA-centric benchmarks such as VQA-RAD Lau et al. (2018), SLAKE Liu et al. (2021), Path-VQA He et al. (2020), and VQA-Med Ben Abacha et al. (2019). More recent resources broaden coverage and reasoning scope, including PMC-VQA Zhang

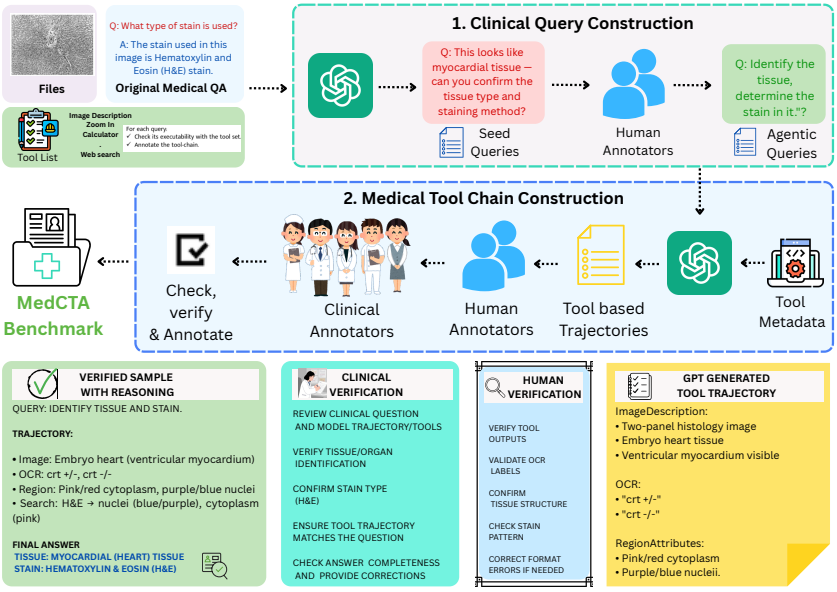


Figure 1: **MedCTA curation pipeline.** ❶ *Query construction:* Annotators expand expert exemplars into structured, executable, and tool-aware clinical queries, followed by clinical verification for correctness and relevance. ❷ *Tool chain construction:* GPT-generated tool trajectories are refined and corrected by annotators. Technical verification ensures executability and format compliance, while clinical experts validate medical correctness and reasoning soundness.

et al. (2024), MMMU-Med Yue et al. (2024), and OmniMedVQA Hu et al. (2024). These datasets have supported increasingly strong medical vision-language models, including BioViL Boecking et al. (2022), MedCLIP Wang et al. (2022), LLaVA-Med Li et al. (2023b), Med-Flamingo Moor et al. (2023), RadFM Wu et al. (2023a), Lingshu Xu et al. (2025c), Fleming-VL Shu et al. (2025), MedGemma Sellergren et al. (2025), and MedMO Deria et al. (2026). However, these benchmarks still mostly evaluate static understanding: report generation or single-turn question answering.

Medical agents and interactive clinical evaluation. Recent work has begun moving toward more interactive medical settings. Multi-agent and coordination-oriented systems include MDAgents Kim et al. (2024) and TAO Kim et al. (2025b). Simulated clinical-interaction benchmarks include AgentClinic Schmidgall et al. (2024) and AI Hospital Fan et al. (2025). Longitudinal record-centric environments are represented by Virtual EHR / MedAgentBench Jiang et al. (2025b), while proactive clarification and uncertainty-aware interaction are emphasized in BehaviorSFT Kim et al. (2025a) and MediQ Li et al. (2024b). Large-scale clinical decision evaluation is further explored in CliBench Ma et al. (2024). Additional closely related efforts include modality-specific reasoning agents such as MedRAX Fallahpour et al. (2025), contamination-free evaluation frameworks like LiveMedBench Yan et al. (2026), HealthAdminBench Bedi et al. (2026) and multimodal agentic diagnosis systems such as MedAgent-Pro Wang et al. (2025). Strong generalist and evaluation-oriented baselines include MedVersa Zhou et al. (2024) and MedQA-CS Yao et al. (2024), while tool-centric and adaptive agent frameworks such as MedAgentGYM Xu et al. (2025b) further explore multi-step orchestration and self-evolving behaviors. In this work, we introduce `MedCTA`, the first agentic benchmark specifically designed to evaluate tool-augmented `LLM` models on realistic clinical tasks with executable tool chains and trajectory-level reasoning.

3 MedCTA Benchmark

In this section, we describe the design and content of the `MedCTA` benchmark. We introduce the formulation of a dataset sample in Section 3.1. The construction procedures for clinical queries and medical tool chains are presented in Sections 3.2 and 3.3, respectively. The overall dataset construction pipeline is shown in Figure 1 with corresponding dataset statistics in Section 3.4.

Table 2: **Query transformation in MedCTA.** Blue highlights perception-only prompts. Red highlights procedural/tool-leaking phrasing. Green highlights the final clinically grounded design.

Domain	Original Dataset Query	LMM-based Agentic Query (Tool-Explicit)	Final Query (Human + Clinician Verified)
Radiology (CT)	What type of thrombosis is shown in the image?	Using the CT image, localize the portal vein and superior mesenteric vein regions and verify whether thrombosis is present in each; then determine the type of thrombosis.	Based on the CT image, what type of venous thrombosis is present?
Radiology (Mass Measurement)	What is the size of the mass?	Using the abdominal CT image, first check for measurement labels via OCR, then estimate the mass dimensions and summarize the size.	Are any measurement labels visible in the image? Also, find the mass size and identify its abdominal quadrant.
Histology (Organ Comparison)	What organs are being compared in the image?	Summarize the multi-panel figure, detect panel titles, and verify organ labels before identifying which organs are compared.	Identify the organs shown.
Histology (Tissue Identification)	What organ is being analyzed in the image?	Localize a representative tissue region with a bounding box and describe its morphology to determine which organ is analyzed.	Identify the tissue and determine the stain used.
Developmental Pathology	What is the main difference between the normal and Gpc3 -/ mouse kidneys at E12.0?	Compare E12.0 kidney sections labeled Gpc3 +/+ and Gpc3 -/-. Confirm labels and inspect ureteric bud branching differences using tools.	Compare Gpc3 +/+ and Gpc3 -/ embryonic kidneys across E12.0, E13.5, and E16.5, focusing on ureteric bud branching, kidney size, and cortical–medullary architecture.
Design Principle	Single-step perception question.	Tool-explicit, step-decomposed, instruction-driven phrasing.	Clinically grounded, goal-oriented query without explicit tool or procedural references.

3.1 Dataset Formulation

Let $\mathcal{D} = \{\tau_k\}_{k=1}^{N}$ denote a library of executable tools. We formulate each task in MedCTA as a structured tuple $(\mathcal{X}, \mathcal{Q}, \mathcal{U}, \pi, \mathcal{A})$ that specifies (i) the clinical context, (ii) an objective to be solved, (iii) the tools required, (iv) the tool-mediated interaction trace, and (v) the final clinical outcome. This design follows the general agentic benchmark template of defining a context, a query that necessitates multi-step tool use, a tool subset, and a stepwise trace of tool calls.

Clinical context. $\mathcal{X}$ is a collection of clinical images, including one or two images (e.g., CT, MRI, X-ray, ultrasound, pathological images) and/or scanned documents. The context may be heterogeneous and multimodal, reflecting realistic clinical setups where evidence is distributed across files.

Agentic query. $\mathcal{Q}$ is a clinically grounded request conditioned on $\mathcal{X}$. While $\mathcal{Q}$ describes a plausible medical goal, solving it requires decomposition into multiple subproblems and strategic invocation of tools from $\mathcal{D}$. Importantly, the query does not reveal which tools are needed, nor their execution order, and the agent must plan to execute the query.

Tool subset. $\mathcal{U} \subseteq \mathcal{D}$ denotes the (hidden) subset of tools that are sufficient to resolve the query: $\mathcal{U} = \bigcup_{i=1}^{m} \{\tau_i\}$, where τ_i is the tool used at step i.

Reference tool chain (interaction trace). We provide a reference tool-assisted reasoning trace $\pi = \{s_i\}_{i=1}^{m}$, where each step is a triplet $s_i = (\tau_i, \alpha_i, \rho_i)$ consisting of the selected tool, its input arguments, and the returned observation (tool output), respectively. This representation enables fine-grained evaluation of whether the agent makes coherent intermediate decisions and uses tools effectively throughout the chain.

Final outcome. $\mathcal{A}$ is the final clinical conclusion produced after completing the tool-mediated process (e.g., diagnosis, quantified assessment, classification, or a finding). For queries in which the medically correct response is not unique, we include multiple acceptable reference answers to reflect valid clinical variability. For queries with a single ground-truth target, $\mathcal{A}$ is unique.

3.2 Clinical Query Construction

A central goal of MedCTA is to move beyond *perception-only* medical VQA and evaluate *agentic* clinical problem solving. The model must interpret evidence in $\mathcal{X}$, infer which tools are useful, and execute a coherent multi-step workflow without being told the tool sequence. To build $(\mathcal{X}, \mathcal{Q}, \mathcal{U})$, we use a semi-automated, human-in-the-loop pipeline that converts shallow dataset-style questions into realistic, tool-requiring clinical objectives while explicitly removing procedural hints from the final query (Table 2). This follows the core principle used in recent agentic vision benchmarks Ashraf et al. (2025): tool use should be necessary, but implicit. Queries are first drafted with LMM assistance, then refined by humans for realism and clarity, while tool usage remains latent in the final task statement.

Stage 1: Seed collection (perception-level prompts). We start from existing medical VQA datasets and curated clinical education resources, extracting image–question pairs $(\mathcal{X}, q)$ that target localized findings, attributes, or recognition (e.g., abnormality presence or organ identification). These seeds provide diverse medical imagery and grounded intent, but they are usually solvable in a single step and therefore do not evaluate planning or tool-mediated reasoning.

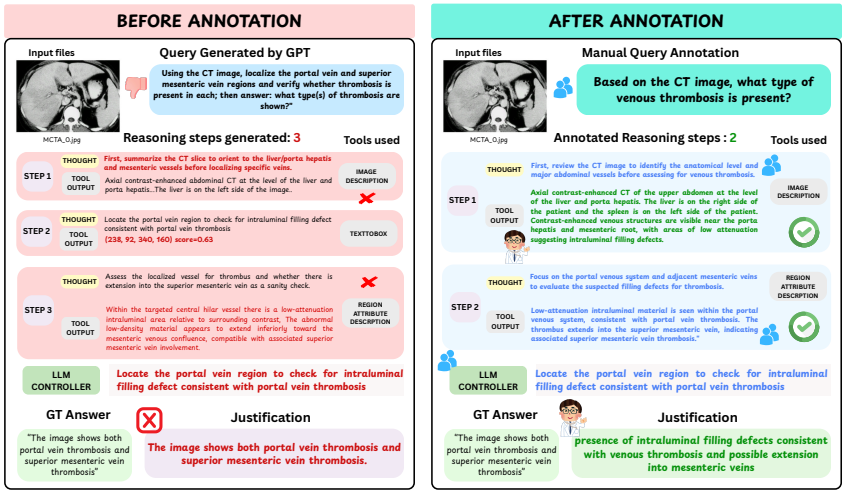


Figure 2: **Comparison of GPT-generated vs. human-annotated trajectories**. Blue: technical fixes (tool compliance, stability, redundancy removal); Green: clinical corrections (sound reasoning, evidence grounding). The annotated version is clinically coherent, avoiding unnecessary steps while reaching the same diagnosis.

Stage 2: LMM drafting (agentic lift). Given a seed pair, we use an LMM to propose an *agentic* reformulation of the question. The input includes the original question, the corresponding visual input $\mathcal{X}$, and the full tool library $\mathcal{D}$. The model is prompted to rewrite the question into a clinically meaningful objective that *requires* tool use and multi-hop reasoning, producing an initial set of agentic queries. Table 2 (column 3) shows examples of these LMM-generated drafts. In practice, such drafts often capture useful latent workflows, but may also over-specify steps, add unnecessary subgoals, or assume unrealistic clinical operations.

Stage 3: Human rewrite (goal-first, tool-agnostic). To ensure the benchmark tests autonomous planning rather than instruction following, expert annotators rewrite the LMM-generated drafts into a *goal-oriented clinical request*. This rewrite removes step-by-step phrasing and tool-suggestive wording (e.g., explicit mention of reading labels, drawing boxes, or extracting text), while preserving the intended reasoning depth. Concretely, the final query is revised so that 1) it is solvable through some subset $\mathcal{U} \subseteq \mathcal{D}$, but $\mathcal{U}$ and the invocation order are not disclosed; 2) tool usage remains *latent*, i.e., the query states the clinical goal rather than the procedure; 3) the wording matches real clinical communication and avoids raw pipeline-style scaffolding. This aligns the task with the benchmark principle that the agent must infer both *what to do next* and *which tools to use*, rather than being guided by explicit procedural instructions.

Stage 4: Clinical validation and final annotation. Finally, clinician reviewers verify each query for medical correctness, coherence, and executability under the tool library. They remove ambiguous or unstable cases and annotate reference outcomes $\mathcal{A}$ where needed, including multiple acceptable answers when clinical descriptions are inherently non-unique. The resulting tasks therefore require multi-step clinical reasoning with implicit tool interaction, enabling robust evaluation of agent planning and decision making in medical settings.

Outcome. For each sample $(\mathcal{X}, q)$, this pipeline transforms a single-hop perception query into a clinically grounded agentic query $\mathcal{Q}$ that is solvable by a latent tool subset $\mathcal{U}$ and evaluated against reference answer(s) $\mathcal{A}$. LMMs provide scalable drafts, humans enforce goal-first phrasing and remove tool leakage, and clinicians ensure correctness and reliability. The result is a benchmark of realistic, challenging tasks that diagnose agentic capability rather than prompt compliance. Table 2 shows examples of this process.

3.3 Medical Tool Chain Construction

Given a finalized task instance from previous step $(\mathcal{X}, \mathcal{Q}, \mathcal{U}, \mathcal{A})$ (Sec. 3.1, Sec. 3.2), we construct an executable reference trajectory π that operationalizes *how* the answer $\mathcal{A}$ can be reached using tools in $\mathcal{U} \subseteq \mathcal{D}$, while remaining faithful to clinical reasoning. Importantly, π is *not* revealed to the agent at

test time; rather, it serves as a structured reference for evaluating tool selection, argument validity, intermediate evidence consistency, and final outcome correctness. We emphasize two principles: **minimality** (avoid redundant steps) and **stability** (avoid brittle dependencies).

Stage 1: LMM-Based Trajectory Drafting (Prompt-Guided Initialization). We first generate a candidate tool trajectory using an `LMM` conditioned on the finalized query $\mathcal{Q}$, tool library $\mathcal{U}$, and answer $\mathcal{A}$. This step provides a scalable initial decomposition of the clinical objective into intermediate tool calls. However, `LLM`-generated trajectories frequently exhibit over-decomposition (e.g., unnecessary localization steps), unstable intermediate dependencies (e.g., bounding-box reliance), or loosely justified reasoning. For example, an `LMM` draft may insert a `TextToBox` localization step before evaluating venous filling defects, thereby creating an avoidable failure point without improving diagnostic fidelity (Figure 2).

Stage 2: Technical Annotation (Executability and Minimality Enforcement). Next, technical annotators refine the `LMM`-generated candidate trajectories to ensure strict tool-schema compliance and execution stability. Each step $s_i = (\tau_i, \alpha_i, \rho_i)$ is validated to confirm: (i) correct tool selection, (ii) valid and stable argument formatting, (iii) coherent interpretation of tool outputs, and (iv) absence of redundant or brittle steps. Unnecessary operations (e.g., intermediate bounding box extraction when region-level evaluation suffices) are removed. The refined chain is then executed and logged in structured JSON format to ensure deterministic reproducibility.

Stage 3: Clinical Verification (Medical Soundness and Workflow Alignment). Clinician reviewers evaluate the technically validated trajectory to ensure medical correctness and realistic workflow alignment. They verify that intermediate outputs correspond to clinically meaningful findings (e.g., identification of intraluminal filling defects), that essential reasoning steps are not missing, and that the final conclusion $\mathcal{A}$ is supported by evidence. Clinicians may revise reasoning chains if they detect unsafe assumptions, incomplete diagnostic justification, or medically implausible tool usage.

Stage 4: Quality Control and Finalization. Only trajectories that pass both technical and clinical review are included in `MedCTA`. This dual-layer validation ensures that reference trajectories are executable, minimal, clinically grounded, and robust to tool instability. The curated trajectories are typically shorter, more stable, and more medically coherent than naive `LMM`-generated drafts, while preserving the multi-step structure required for trajectory-aware evaluation (Figure 2).

More Details. Appendix A presents the data card, Appendix B details the executable tools, Appendix C covers clinician verification and agreement, Appendix D describes the annotation protocol, Appendix E explain the stepwise construction of executable clinical tool trajectories, and Appendix F provides `LMM`-based generation prompts.

3.4 Dataset Statistics

With a total of 321 human annotation hours, `MedCTA` contains 107 clinician-validated tasks with 1,926 autonomous rollouts grounded in multi-modal medical data, including images and 5 executable tools spanning perception, reasoning, and structured interaction. The benchmark covers 34 anatomical regions and body systems, such as the brain, chest, abdomen, extremities, pelvis, ocular structures, and cellular-level pathology, enabling broad cross-specialty reasoning. Tasks require 2–4 tool-execution steps (average 3.1), reflecting the benchmark's compositional nature. `MedCTA` also exhibits diverse modality coverage, led by CT (20.0%), report-based inputs (20.0%), histopathology (18.3%), X-ray (9.6%), MRI (8.7%), fundus photography (7.0%), and gross pathology (5.2%), with additional representation from microscopy/TEM (3.5%), dermoscopy (3.5%), PET (0.9%), mammography (0.9%), and general pathology (0.9%) etc. More details are in the Appendix G.

4 Experiments

4.1 Benchmark Evaluation

Setup. We evaluate `MedCTA` with **18** large language models, covering both proprietary APIs and open-source checkpoints from the GPT Achiam et al. (2023); Singh et al. (2026); OpenAI et al. (2025), Claude Anthropic (2024), Gemini Team et al. (2023), Qwen Bai et al. (2023), LLaMA Meta (2024), DeepSeek Lu et al. (2024), Mistral Jiang et al. (2023), and Phi Abdin et al. (2024) families,

Table 3: Evaluation metrics are grouped into step-by-step tool-use fidelity, clinical reasoning quality, and outcome accuracy. All scores are reported as percentages in $[0, 100]$ unless otherwise specified. More detailed definitions are provided in Appendix § G.

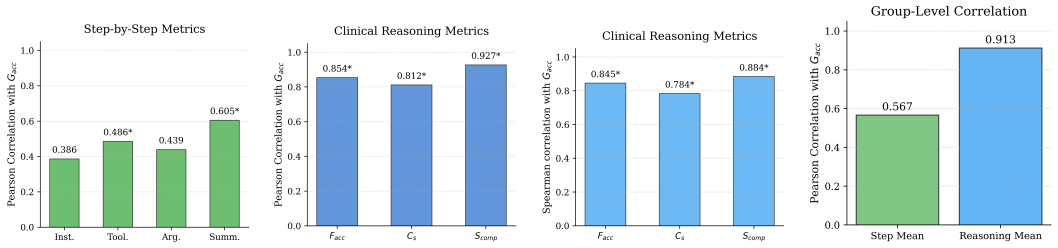
Metric (Symbol)	Description
Step-by-Step Mode	
`InstAcc` ($\mathbf{I_{acc}}$)	*Whether the model follows the required agent protocol and output format at each step.*
`ToolAcc` ($\mathbf{T_{acc}}$)	*Whether the model selects the correct tool for step $n{+}1$ given the first n reference steps.*
`ArgAcc` ($\mathbf{A_{acc}}$)	*Whether the predicted tool arguments are syntactically valid and clinically appropriate.*
`SummAcc` ($\mathbf{S_{acc}}$)	*Whether the model correctly summarizes intermediate evidence into the expected conclusion.*
Clinical Reasoning Mode	
`Clinical Faithfulness` ($\mathbf{F_{acc}}$)	*Logical consistency of intermediate reasoning with clinical evidence and workflow.*
`Context Integration Score` ($\mathbf{C_s}$)	*Effective integration of multimodal clinical context (images, reports, structured data).*
`Semantic Completeness` ($\mathbf{S_{comp}}$)	*Coverage of all clinically necessary findings required for a valid conclusion.*
Outcome Mode	
`Goal Accuracy` ($\mathbf{G_{acc}}$)	*Final answer accuracy for diagnostic and interpretive clinical queries.*

Table 4: **Main results of `MedCTA`.** $*$ refers to a closed-source model.

Family	Model	Step-by-Step				Clinical Reasoning			Outcome
		Inst.	Tool.	Arg.	Summ.	F_{acc}	C_s	S_{comp}	G_{acc}
OpenAI	GPT-5.4*	35.27	**23.46**	12.61	35.51	17.52	14.21	18.60	**31.54**
	GPT-5.4-mini*	5.36	6.74	3.23	0.93	16.47	10.56	17.29	28.31
	GPT-5.4-nano *	33.93	18.18	12.02	34.24	**18.43**	11.96	14.77	20.30
	GPT-oss-20B	1.79	0.00	0.00	0.00	1.31	0.56	1.68	3.18
Anthropic	Claude-opus-4-6*	24.78	8.80	0.59	39.25	14.11	14.86	**23.83**	31.32
	Claude-sonnet-4-6*	23.66	4.99	0.00	33.64	12.77	12.90	20.19	25.33
	Claude-haiku-4-5*	27.46	13.78	4.69	**43.93**	9.35	3.36	14.11	23.08
Google	Gemini-3-flash*	3.35	17.30	0.00	5.61	11.31	8.60	15.98	25.87
	Gemini-3-flash-lite*	2.90	8.21	0.00	1.87	10.75	6.82	14.58	23.64
Qwen	Qwen3.5-9B	**44.20**	14.37	**13.78**	29.91	10.37	**17.10**	13.36	21.64
	Qwen3-8B	33.93	10.56	7.04	32.71	8.50	10.09	11.50	27.80
DeepSeek	DeepSeek-R1-Distill-7B	10.49	3.52	0.00	7.48	2.62	0.84	3.36	10.61
	Deepseek-llm-7b-chat	11.61	6.45	0.00	4.67	4.30	2.62	4.02	11.00
	DeepSeek-V2-Lite-Chat	11.83	11.14	0.29	0.00	3.83	3.55	6.54	6.96
Meta	Llama-3.1-8B-Instruct	23.66	7.92	0.00	6.54	7.94	5.42	11.21	18.94
	Llama-3.2-3B-Instruct	18.53	1.76	0.00	4.67	3.08	1.68	5.14	11.29
Mistral	Mistral-7B	18.75	14.66	0.00	9.35	2.52	1.87	3.46	9.40
Microsoft	Phi-4	20.09	6.45	0.00	14.02	6.36	3.36	6.17	10.65

on clinician-verified tasks with **1926** rollouts in total. We use OpenCompass Contributors (2023) as the evaluation harness and Lagent Team (2023b) as the agent framework, adopting ReAct Yao et al. (2023)-style prompting and an Agentlego Team (2023a) tool interface. In this autonomous setting, the model is fully responsible for the entire control loop: it must decide *which tool* to call, produce *valid arguments*, interpret tool outputs, decide *when to stop*, and synthesize the final answer. See Appendix §B for the executable tool library and Appendix §F for prompts.

Evaluation metrics. We report performance using three complementary metric groups (Table 3). STEP-BY-STEP measures interaction fidelity (instruction following, tool selection, and argument validity), following the trajectory-level evaluation protocol of GTA Wang et al. (2024b). CLINICAL REASONING evaluates evidence usage and summary quality. OUTCOME evaluates task success via the official final outcome score G_{acc}. Together, these metrics disentangle controller competence from evidence-conditioned reasoning. More details are in Appendix § G. To further validate the automated clinical-reasoning evaluation, we conducted an additional clinician audit on 36 benchmark tasks across four representative models, yielding 144 evaluated rollouts and 432 scalar clinical judgments over F_{acc}, C_s, and S_{comp}. The audit confirms the same qualitative conclusion as the automated judge: clinical reasoning remains weak even for strong models, with an overall human clinical-reasoning score of 28.5/100 and moderate model-level agreement with the automated evaluator on the aggregate clinical score (Spearman $\rho = 0.60$; Appendix C.6).



(a) Step (Pearson) (b) Reasoning (Pearson) (c) Reasoning (Spearman) (d) Group-level

Figure 3: **Correlation with final accuracy (G_{acc}).** Step-level metrics show moderate correlation, while clinical reasoning metrics (especially S_{comp} and F_{acc}) strongly correlate with final performance. This indicates that controller instability prevents models from reaching stable reasoning.

Table 5: **Autonomous vs. Gold (G_{acc}).**

Model	Auto	Gold	Δ
GPT-5.4	31.54	49.50	**+17.96**
Claude-opus-4-6	31.32	66.40	**+35.08**
Qwen3.5-9B	21.64	49.50	+27.86

Table 6: **Global rollout diagnostics.**

Metric	Value	Insight
API error rate	**64.2**	protocol instability
Under-call rate	**99.2**	premature stopping
Protocol failure	**58.3**	rollout breakdown
Tool-selection failure	41.6	incorrect actions

Results. Table 4 shows that `MedCTA` remains far from saturated, with the best G_{acc} only **31.54**, indicating substantial headroom. Closed models generally lead, but strong open models (e.g., Qwen3-8B at **27.80**) are competitive, suggesting controller stability can offset scale. Performance is multi-dimensional, no model dominates across tool use, instruction following, argument construction, and summarization, and these do not consistently translate to outcome. Notably, outcome can overestimate reliability, as some models achieve high G_{acc} despite weak step-level fidelity, while others show strong interaction metrics without corresponding success. While scaling often helps, it is not monotonic, and interaction style (e.g., stopping and evidence synthesis) can matter as much as model size. Furthermore, Fig. 3 shows the relationship between step-level execution, clinical reasoning metrics, and final answer accuracy. The results indicate that reasoning-centric metrics, particularly S_{comp} and F_{acc}, exhibit substantially stronger correlations with G_{acc} than step-by-step signals, highlighting that final performance is primarily driven by reasoning quality rather than intermediate execution. Reasoning metrics correlate most with final answer quality, while rollout diagnostics show controller failures dominate trajectory collapse.

4.2 Failure Analysis and Discussion

Gold-standard routing exposes a large controller gap. Table 5 shows that replacing autonomous tool selection with a gold-standard tool trajectory produces large gains for representative frontier models. Claude-opus-4-6 rises from **31.32** to **66.40**, Qwen3.5-9B from **21.64** to **49.50**, and GPT-5.4 from **31.54** to **49.50**. This gap is the clearest evidence that current models possess substantial latent clinical reasoning ability, but fail to reliably access it under autonomous control. The bottleneck is therefore not simply medical knowledge or perception: it is the ability to maintain a valid tool-using loop, select the right evidence source, and stop only after sufficient evidence has been gathered.

Autonomous rollouts fail before clinical reasoning can fully begin. Table 6 summarizes the dominant rollout errors: **64.2%** of episodes encounter API/protocol errors, and **99.2%** under-call relative to the reference trajectory. The first-failure breakdown is equally sharp: **58.3%** of failures occur first at protocol/API handling and **41.6%** at tool selection. Most models do not primarily fail after carefully reasoning over wrong evidence; they fail earlier, by leaving the clinical tool-use process before the necessary evidence has been acquired.

Qualitative failures reveal evidence drift, not just wrong answers. Figure 4 illustrates the same diagnostic story at trajectory level. In the sentinel-node metastasis example, the gold trajectory grounds the answer by extracting visible chart labels and mapping them to histologic and molecular assays. GPT-5.4 bypasses this evidence chain and answers prematurely with an incomplete generic method description, while Qwen3.5-9B encounters unstable execution and drifts toward unrelated clinical procedures despite recoverable OCR evidence. This example is representative of a broader

Qualitative Comparison: Sentinel-node metastasis detection (semantic drift)

Query: *Identify the methods used to detect metastases in sentinel nodes.*

Gold	GPT-5.4	Qwen3.5-9B
• `ImageDescription` + `OCR` extract H&E, CK-18, CEA, hTRT, MUC-1. • `Search + Region` maps to histologic + molecular assays. • **Answer:** H&E staining + RT-PCR/Southern blot.	• **Skips tool reasoning** and answers directly. • **Over-generalizes** to generic "IHC + RT-PCR". • **Answer:** IHC + RT-PCR. • **Failure:** premature answer, incomplete grounding.	• **API errors** + weak initial grounding. • **Ignores OCR evidence**. • **Answer:** unrelated clinical procedures. • **Failure:** semantic drift from visual evidence.

Issues: *Errors arise from broken reasoning chains (GPT-5.4) and drift from extracted evidence (Qwen3.5-9B), showing that correct perception alone is insufficient without grounded execution.*

Figure 4: **Qualitative failure analysis on sentinel-node metastasis detection.** The gold trajectory grounds reasoning in extracted visual evidence via tool execution. GPT-5.4 fails due to premature answering, while Qwen3.5-9B exhibits semantic drift by ignoring OCR evidence and relying on unrelated priors. This shows that accurate perception alone does not ensure grounded reasoning.

pattern documented in Appendix H: failures often arise from broken evidence acquisition, semantic drift away from tool observations, or premature synthesis from prior knowledge. The central lesson is that correct perception alone is insufficient; reliable medical agents must remain evidence-obedient throughout the full trajectory.

Takeaway. Together, these diagnostics show that current medical tool agents fail in two separable layers. The first and dominant layer is *controller reliability*: protocol adherence, tool recruitment, and stop/continue calibration. The second layer, visible after gold-standard routing improves tool access, is *grounded long-horizon clinical reasoning*: using localized evidence correctly across multiple steps. This decomposition is the main diagnostic value of `MedCTA` and identifies where future progress is most likely to matter. To mitigate evaluation bias, we use a structured LLM-as-judge framework with disentangled prompts that independently assess workflow, evidence usage, factual correctness, and completeness while explicitly ignoring other factors. All evaluations are grounded in clinician-validated reference trajectories, ensuring alignment with medically plausible reasoning. This conclusion is also supported by a clinician audit of model rollouts (Appendix C.6): the best audited model reaches only 37.9/100 mean human clinical-reasoning score, and only 6.9% of audited rollouts achieve a strong mean score (≥ 0.7), indicating that the observed failures are visible to clinical reviewers and are not merely artifacts of LLM-based judging.

Additional Experiments. We also include additional experiments showing a large gold-standard tool routing gap (Appendix H.1), early protocol and tool-use failures with under-calling (Appendix H.2), and residual long-horizon grounding challenges after routing correction (Appendices H.4, H.5).

5 Conclusion

We introduce `MedCTA`, a benchmark for evaluating medical tool agents beyond static medical VQA by measuring planning, tool selection, localized evidence acquisition, and end-to-end outcome quality on clinician-verified multimodal tasks. Across 18 models, the benchmark shows that reliable clinical tool use remains unsolved: even strong systems achieve limited outcome accuracy, gold-standard tool routing exposes a large controller gap, and most failures arise from protocol instability, under-calling, and weak local grounding rather than final synthesis alone. These results position `MedCTA` not only as a challenging benchmark, but also as a diagnostic framework for developing more faithful, robust, and clinically grounded medical agents.

Limitations. `MedCTA` is currently monolingual and intentionally diagnostic rather than exhaustive. Its semi-automated curation pipeline may occasionally introduce lower-quality samples and require multiple verifications. Scaling and adding more diverse samples remain future work.

Ethics Statement

MedCTA uses medical inputs drawn from public research datasets that are de-identified to the best of our knowledge. No new patient data were collected, and no patients were recruited or subjected to an intervention. Source assets remain governed by their upstream licenses; the release documents provenance and requires users to comply with the applicable source terms. MedCTA is a research evaluation resource, not a medical device or a clinical decision-support system. Benchmark performance must not be interpreted as evidence of patient-level safety, fairness, or deployment readiness. The benchmark is not representative of the full distribution of patients, institutions, modalities, or diseases, and demographic auditing is limited by missing attributes in public source data. Models evaluated on MedCTA may produce plausible but incorrect conclusions, omit necessary evidence, or rely on unsupported priors. Any clinical use would require independent clinician validation, institutional approval, and applicable regulatory review.

Reproducibility Statement

Section 3 specifies the task representation and the benchmark construction and validation pipeline, while Section 4 defines the evaluation settings and metrics. Appendices B, D, F, G, and J provide the tool interfaces, annotation protocols, prompts, metric definitions, and automated-judge rubrics, and Appendix C documents clinician validation and the clinician audit of model rollouts. The anonymized repository contains the released benchmark and evaluation suite. External retrieval is frozen by caching the snippets, source identifiers, and access dates used by the reference trajectories, reducing temporal drift. Exact replication of proprietary-API results may nevertheless be affected by provider-side model updates.

AI Use Statement

Generative AI tools were used to draft candidate reformulations of source questions into step-implicit clinical tasks, initialize candidate tool trajectories, standardize annotation formatting, assist with data processing and filtering, and improve manuscript wording. AI-generated outputs were not accepted as ground truth without review: technical annotators checked tool schemas, arguments, and executability, and clinician reviewers revised and validated the clinical queries, evidence, trajectories, and final outcomes. Separately, language models are the systems evaluated in this benchmark, and an LLM is used for rubric-based automated evaluation as described in Section 4 and Appendix J. All AI-assisted material was reviewed by the authors, who take responsibility for the manuscript, claims, annotations, and released artifacts.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 2024.

Tajamul Ashraf, Amal Saqib, Hanan Ghani, Muhra AlMahri, Yuhao Li, Noor Ahsan, Umair Nawaz, Jean Lahoud, Hisham Cholakkal, Mubarak Shah, et al. Agent-x: Evaluating deep multimodal reasoning in vision-centric agentic tasks. *arXiv preprint arXiv:2505.24876*, 2025.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Suhana Bedi, Ryan Welch, Ethan Steinberg, Michael Wornow, Taeil Matthew Kim, Haroun Ahmed, Peter Sterling, Bravim Purohit, Qurat Akram, Angelic Acosta, et al. Healthadminbench: Evaluating computer-use agents on healthcare administration tasks. *arXiv preprint arXiv:2604.09937*, 2026.

Asma Ben Abacha, Sadid A Hasan, Vivek V Datla, Dina Demner-Fushman, and Henning Müller. Vqa-med: Overview of the medical visual question answering task at imageclef 2019. In *Proceedings of CLEF (Conference and Labs of the Evaluation Forum) 2019 Working Notes*. 9-12 September 2019, 2019.

Benedikt Boecking, Naoto Usuyama, Shruthi Bannur, Daniel C. Castro, Anton Schwaighofer, Stephanie L. Hyland, Maria T. Wetscherek, Tristan Naumann, Harsha Nori, Neeraj Ahuja, et al. Making the most of text semantics to improve biomedical vision–language processing, 2022. URL https://arxiv.org/abs/2204.09817.

Stephanie Cabral, Daniel Restrepo, Zahir Kanjee, Philip Wilson, Byron Crowe, Raja-Elie Abdulnour, and Adam Rodman. Clinical reasoning of a generative artificial intelligence model compared with physicians. *JAMA internal medicine*, 184(5):581–583, 2024.

Harrison Chase. Langchain, October 2022. URL https://github.com/langchain-ai/langchain.

Li Sze Chow and Raveendran Paramesran. Review of medical image quality assessment. *Biomedical signal processing and control*, 27:145–154, 2016.

OpenCompass Contributors. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass, 2023.

Ankan Deria, Komal Kumar, Adinath Madhavrao Dukre, Eran Segal, Salman Khan, and Imran Razzak. Medmo: Grounding and understanding multimodal large language model for medical images. *arXiv preprint arXiv:2602.06965*, 2026.

James S. Duncan and Nicholas Ayache. Medical image analysis: Progress over two decades and the challenges ahead. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):85–106, 2000. doi: 10.1109/34.824822.

Adibvafa Fallahpour, Jun Ma, Alif Munim, Hongwei Lyu, and Bo Wang. Medrax: Medical reasoning agent for chest x-ray. *arXiv preprint arXiv:2502.02673*, 2025.

Zhihao Fan, Lai Wei, Jialong Tang, Wei Chen, Siyuan Wang, Zhongyu Wei, Jun Xie, Fei Huang, and Jingren Zhou. Ai hospital: Benchmarking large language models in a multi-agent medical interaction simulator. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 10183–10213. Association for Computational Linguistics, January 2025. URL https://aclanthology.org/2025.coling-main.680/.

Significant Gravitas. Autogpt, 2023. URL https://github.com/Significant-Gravitas/AutoGPT.

Xuehai He, Yichen Zhang, Luntian Mou, Eric Xing, and Pengtao Xie. Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*, 2020.

Yutao Hu, Tianbin Li, Quanfeng Lu, Wenqi Shao, Junjun He, Yu Qiao, and Ping Luo. Omnimedvqa: A new large-scale comprehensive evaluation benchmark for medical lvlm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22170–22183, 2024.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

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, 2025a.

Yixing Jiang, Kameron C. Black, Gloria Geng, Danny Park, James Zou, Andrew Y. Ng, and Jonathan H. Chen. A virtual EHR environment to benchmark medical LLM agents. *NEJM AI*, 2025b. doi: 10.1056/AIdbp2500144. URL https://ai.nejm.org/doi/full/10.1056/AIdbp2500144.

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?, 2023. URL https://arxiv.org/abs/2310.06770.

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.

Yubin Kim, Chanwoo Park, Hyewon Jeong, Yik S Chan, Xuhai Xu, Daniel McDuff, Hyeonhoon Lee, Marzyeh Ghassemi, Cynthia Breazeal, and Hae W Park. Mdagents: An adaptive collaboration of llms for medical decision-making. *Advances in Neural Information Processing Systems*, 37: 79410–79452, 2024.

Yubin Kim, Zhiyuan Hu, Hyewon Jeong, Eugene W Park, Shuyue Stella Li, Chanwoo Park, Shiyun Xiong, MingYu Lu, Hyeonhoon Lee, Xin Liu, et al. Behaviorsft: Behavioral token conditioning for health agents across the proactivity spectrum. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 9789–9817, 2025a.

Yubin Kim, Hyewon Jeong, Chanwoo Park, Eugene Park, Haipeng Zhang, Xin Liu, Hyeonhoon Lee, Daniel McDuff, Marzyeh Ghassemi, Cynthia Breazeal, et al. Tiered agentic oversight: A hierarchical multi-agent system for healthcare safety. *arXiv preprint arXiv:2506.12482*, 2025b.

Jason J Lau, Soumya Gayen, Asma Ben Abacha, and Dina Demner-Fushman. A dataset of clinically generated visual questions and answers about radiology images. *Scientific data*, 5(1):180251, 2018.

Binxu Li, Tiankai Yan, Yuanting Pan, Jie Luo, Ruiyang Ji, Jiayuan Ding, Zhe Xu, Shilong Liu, Haoyu Dong, Zihao Lin, et al. Mmedagent: Learning to use medical tools with multi-modal agent. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 8745–8760, 2024a.

Chenliang Li, He Chen, Ming Yan, Weizhou Shen, Haiyang Xu, Zhikai Wu, Zhicheng Zhang, Wenmeng Zhou, Yingda Chen, Chen Cheng, et al. Modelscope-agent: Building your customizable agent system with open-source large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 566–578, 2023a.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day, 2023b. URL https://arxiv.org/abs/2306.00890.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li API-bank. A comprehensive benchmark for tool-augmented llms. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3102–3116, 2023c.

Shuyue Stella Li, Vidhisha Balachandran, Shangbin Feng, Jonathan S. Ilgen, Emma Pierson, Pang Wei Koh, and Yulia Tsvetkov. Mediq: Question-asking LLMs and a benchmark for reliable interactive clinical reasoning, 2024b. URL https://arxiv.org/abs/2406.00922.

Bo Liu, Li-Ming Zhan, Li Xu, Lin Ma, Yan Yang, and Xiao-Ming Wu. Slake: A semantically-labeled knowledge-enhanced dataset for medical visual question answering. In *2021 IEEE 18th international symposium on biomedical imaging (ISBI)*, pp. 1650–1654. IEEE, 2021.

Shilong Liu, Hao Cheng, Haotian Liu, Hao Zhang, Feng Li, Tianhe Ren, Xueyan Zou, Jianwei Yang, Hang Su, Jun Zhu, et al. Llava-plus: Learning to use tools for creating multimodal agents. In *European conference on computer vision*, pp. 126–142. Springer, 2024.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Yaofeng Sun, et al. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*, 2024.

Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. Chameleon: Plug-and-play compositional reasoning with large language models, 2023. URL https://arxiv.org/abs/2304.09842.

Mingyu Derek Ma, Chenchen Ye, et al. Clibench: A multifaceted and multigranular evaluation of large language models in clinical decisions on diagnoses, procedures, lab tests orders and prescriptions, 2024. URL https://arxiv.org/abs/2406.09923.

AI Meta. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI Blog (accessed 2024–04–20). There is no corresponding record for this reference*, 2024.

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*, 2023.

Michael Moor, Shraey B. Huang, Yadong Wu, T. Kapur, et al. Med-flamingo: a multimodal medical few-shot learner. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 225 of *Proceedings of Machine Learning Research*, 2023. URL https://proceedings.mlr.press/v225/moor23a.html.

Yohei Nakajima. Babyagi, 2023. URL https://github.com/yoheinakajima/babyagi.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. Webgpt: Browser-assisted question-answering with human feedback, 2021. URL https://arxiv.org/abs/2112.09332.

OpenAI, :, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, Kai Chen, Mark Chen, Enoch Cheung, Aidan Clark, Dan Cook, Marat Dukhan, Casey Dvorak, Kevin Fives, Vlad Fomenko, Timur Garipov, Kristian Georgiev, Mia Glaese, Tarun Gogineni, Adam Goucher, Lukas Gross, Katia Gil Guzman, John Hallman, Jackie Hehir, Johannes Heidecke, Alec Helyar, Haitang Hu, Romain Huet, Jacob Huh, Saachi Jain, Zach Johnson, Chris Koch, Irina Kofman, Dominik Kundel, Jason Kwon, Volodymyr Kyrylov, Elaine Ya Le, Guillaume Leclerc, James Park Lennon, Scott Lessans, Mario Lezcano-Casado, Yuanzhi Li, Zhuohan Li, Ji Lin, Jordan Liss, Lily, Liu, Jiancheng Liu, Kevin Lu, Chris Lu, Zoran Martinovic, Lindsay McCallum, Josh McGrath, Scott McKinney, Aidan McLaughlin, Song Mei, Steve Mostovoy, Tong Mu, Gideon Myles, Alexander Neitz, Alex Nichol, Jakub Pachocki, Alex Paino, Dana Palmie, Ashley Pantuliano, Giambattista Parascandolo, Jongsoo Park, Leher Pathak, Carolina Paz, Ludovic Peran, Dmitry Pimenov, Michelle Pokrass, Elizabeth Proehl, Huida Qiu, Gaby Raila, Filippo Raso, Hongyu Ren, Kimmy Richardson, David Robinson, Bob Rotsted, Hadi Salman, Suvansh Sanjeev, Max Schwarzer, D. Sculley, Harshit Sikchi, Kendal Simon, Karan Singhal, Yang Song, Dane Stuckey, Zhiqing Sun, Philippe Tillet, Sam Toizer, Foivos Tsimpourlas, Nikhil Vyas, Eric Wallace, Xin Wang, Miles Wang, Olivia Watkins, Kevin Weil, Amy Wendling, Kevin Whinnery, Cedric Whitney, Hannah Wong, Lin Yang, Yu Yang, Michihiro Yasunaga, Kristen Ying, Wojciech Zaremba, Wenting Zhan, Cyril Zhang, Brian Zhang, Eddie Zhang, and Shengjia Zhao. gpt-oss-120b and gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Yujia Qin, Zihan Cai, Dian Jin, Lan Yan, Shihao Liang, Kunlun Zhu, Yankai Lin, Xu Han, Ning Ding, Huadong Wang, et al. Webcpm: Interactive web search for chinese long-form question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8968–8988, 2023.

Samuel Schmidgall, Rojin Ziaei, Carl Harris, Eduardo Reis, Jeffrey Jopling, and Michael Moor. Agentclinic: A multimodal agent benchmark to evaluate ai in simulated clinical environments, 2024. URL https://arxiv.org/abs/2405.07960.

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, et al. Medgemma technical report. *arXiv preprint arXiv:2507.05201*, 2025.

Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19(1):221–248, 2017.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Advances in Neural Information Processing Systems*, 36, 2024.

Yan Shu, Chi Liu, Robin Chen, Derek Li, and Bryan Dai. Fleming-vl: Towards universal medical visual reasoning with multimodal llms. *arXiv preprint arXiv:2511.00916*, 2025.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2026.

Yifan Song, Weimin Xiong, Dawei Zhu, Cheng Li, Ke Wang, Ye Tian, and Sujian Li. Restgpt: Connecting large language models with real-world applications via restful apis. *arXiv preprint arXiv:2306.06624*, 2023.

Jinghan Sun, Dong Wei, Liansheng Wang, and Yefeng Zheng. Lesion guided explainable few weak-shot medical report generation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 615–625. Springer, 2022.

Lin Sun et al. Docagent: An agentic framework for multi-modal long-context document understanding. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025. URL https://aclanthology.org/2025.emnlp-main.893/.

Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/papers/Suris_ViperGPT_Visual_Inference_via_Python_Execution_for_Reasoning_ICCV_2023_paper.pdf.

AgentLego Developer Team. Enhance llm agents with versatile tool apis. https://github.com/InternLM/agentlego, 2023a.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Lagent Developer Team. Lagent: InternLM a lightweight open-source framework that allows users to efficiently build large language model(llm)-based agents. https://github.com/InternLM/lagent, 2023b.

C Wang, W Luo, Q Chen, H Mai, J Guo, S Dong, XM Xuan, Z Li, L Ma, and S Gao. Mllm-tool: A multimodal large language model for tool agent learning. *arXiv preprint arXiv:2401.10727*, 4, 2024a.

Jize Wang, Zerun Ma, Yining Li, Songyang Zhang, Cailian Chen, Kai Chen, and Xinyi Le. Gta: A benchmark for general tool agents. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 75749–75790. Curran Associates, Inc., 2024b. doi: 10.52202/079017-2412. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/8a75ee6d4b2eb0b777f549a32a5a5c28-Paper-Datasets_and_Benchmarks_Track.pdf.

Zifeng Wang, Zhenbang Wu, Dinesh Agarwal, and Jimeng Sun. Medclip: Contrastive learning from unpaired medical images and text, 2022. URL https://arxiv.org/abs/2210.10163.

Ziyue Wang, Junde Wu, Linghan Cai, Chang Han Low, Xihong Yang, Qiaxuan Li, and Yueming Jin. Medagent-pro: Towards evidence-based multi-modal medical diagnosis via reasoning agentic workflow. *arXiv preprint arXiv:2503.18968*, 2025.

Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. Towards generalist foundation model for radiology by leveraging web-scale 2d&3d medical data, 2023a. URL https://arxiv.org/abs/2308.02463.

Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023b.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Shaokun Zhang, Erkang Zhu, Beibin Li, Li Jiang, Xiaoyun Zhang, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation framework, 2023c. URL https://arxiv.org/abs/2308.08155.

Junlin Xie, Zhihong Chen, Ruifei Zhang, Xiang Wan, and Guanbin Li. Large multimodal agents: A survey. *arXiv preprint arXiv:2402.15116*, 2024a.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *arXiv preprint arXiv:2404.07972*, 2024b.

Gelei Xu, Xueyang Li, Yixiong Chen, Yuying Duan, Shuqing Wu, Alexander Yu, Ching-Hao Chiu, Juntong Ni, Ningzhi Tang, Toby Jia-Jun Li, et al. A comprehensive survey of ai agents in healthcare. *TechRxiv*, 2025a. doi: 10.36227/techrxiv.176240542.22279040/v1. URL https://www.techrxiv.org/doi/full/10.36227/techrxiv.176240542.22279040/v1.

Ran Xu, Yuchen Zhuang, Yishan Zhong, Yue Yu, Xiangru Tang, Hang Wu, May Dongmei Wang, Peifeng Ruan, Donghan Yang, Tao Wang, et al. Medagentgym: Training llm agents for code-based medical reasoning at scale. In *The Second Workshop on GenAI for Health: Potential, Trust, and Policy Compliance*, 2025b.

Weiwen Xu, Hou Pong Chan, Long Li, Mahani Aljunied, Ruifeng Yuan, Jianyu Wang, Chenghao Xiao, Guizhen Chen, Chaoqun Liu, Zhaodonghui Li, et al. Lingshu: A generalist foundation model for unified multimodal medical understanding and reasoning. *arXiv preprint arXiv:2506.07044*, 2025c.

Zhiling Yan, Dingjie Song, Zhe Fang, Yisheng Ji, Xiang Li, Quanzheng Li, and Lichao Sun. Livemedbench: A contamination-free medical benchmark for llms with automated rubric evaluation. *arXiv preprint arXiv:2602.10367*, 2026.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. Swe-agent: Agent-computer interfaces enable automated software engineering, 2024. URL https://arxiv.org/abs/2405.15793.

Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*, 2023.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022.

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

Zonghai Yao, Zihao Zhang, Chaolong Tang, Xingyu Bian, Youxia Zhao, Zhichao Yang, Junda Wang, Huixue Zhou, Won Seok Jang, Feiyun Ouyang, et al. Medqa-cs: Benchmarking large language models clinical skills using an ai-sce framework. *arXiv preprint arXiv:2410.01553*, 2024.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of CVPR*, 2024.

Chi Zhang, Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. Appagent: Multimodal agents as smartphone users, 2023.

Xiaoman Zhang, Chaoyi Wu, Ziheng Zhao, Weixiong Lin, Ya Zhang, Yanfeng Wang, and Weidi Xie. Pmc-vqa: Visual instruction tuning for medical visual question answering, 2024. *URL https://arxiv.org/abs/2305.10415*, 40, 2024.

Hong-Yu Zhou, Julián Nicolás Acosta, Subathra Adithan, Suvrankar Datta, Eric J Topol, and Pranav Rajpurkar. Medversa: a generalist foundation model for medical image interpretation. *arXiv preprint arXiv:2405.07988*, 2024.

Appendix for MedCTA: A benchmark for Clinical Tool Agents

Section A	**Datacard for MedCTA Benchmark**
Section B	**Executable Tool Library**
Section C	**Inter Annotator Agreement**
Section D	**Instruction for Annotators**
Section E	**Stepwise Construction of Executable Clinical Tool Trajectories**
Section F	**Prompting Templates for Agent Trajectory Construction**
Section G	**Metrics and Implementation Details**
Section H	**Additional Experimental Results and Diagnostic Analyses**
Section I	**MedCTA Task Examples**
Section J	**LLM-as-Judge Prompts**

A Datacard for MedCTA

A.1 Motivation

- **For what purpose was the dataset created?**
  MedCTA is designed to evaluate the *multi-step clinical reasoning and tool-orchestration capabilities* of multimodal medical agents in realistic healthcare scenarios. Unlike traditional medical VQA benchmarks, it features clinician-validated, step-implicit clinical queries that require structured reasoning and implicit tool use. The benchmark integrates executable tools spanning perception, operation, clinical reasoning, and reporting, grounded in authentic multimodal clinical inputs. MedCTA bridges the gap between static medical QA and real-world tool-mediated clinical workflows.
- **Who created the dataset?**
  The dataset was created by the authors of this paper in collaboration with clinician annotators and technical researchers.
- **Who funded the creation of the dataset?**
  Funding details will be disclosed after the review process.

A.2 Composition

- **What do the instances represent?**
  Each instance represents a clinically grounded task stored in JSON format. It includes a natural-language clinical query, multimodal medical inputs (e.g., CT, MRI, X-ray, pathology images, reports), a set of executable tools, and a reference tool trajectory. The trajectory consists of step-wise tool calls with arguments and outputs, ending with a clinically validated final answer.
- **How many instances are there?**
  MedCTA contains **107 clinician-validated tasks**, each grounded in real medical data and paired with executable tool trajectories.
- **Is the dataset a sample or complete?**
  The dataset is a purpose-built benchmark and contains all curated and validated instances constructed for MedCTA.
- **What data does each instance include?**
  Each instance includes a clinical query, multimodal inputs, tool descriptions, a reference execution trajectory, and a final validated answer.
- **Is there a label or target?**
  Yes. Each instance includes a reference trajectory and final clinical outcome. For ambiguous tasks, multiple acceptable answers may be provided.

- **Is any information missing?**
  No.
- **Are relationships between instances explicit?**
  No.
- **Are there recommended splits?**
  MedCTA is designed as an evaluation benchmark and does not define training/validation splits.
- **Are there errors or noise?**
  The dataset is constructed through a semi-automated pipeline and validated by clinicians. Minor annotation errors may still exist.
- **Is the dataset self-contained?**
  Yes. While images originate from public datasets, all queries, trajectories, and annotations are newly constructed and included.
- **Does it contain confidential data?**
  No. All data is publicly sourced and de-identified.
- **Does it contain harmful content?**
  No.

A.3 Collection Process

- **How was the data acquired?**
  Clinical queries and draft trajectories were generated using large multimodal models (e.g., GPT-4o) and refined through human annotation and clinician validation. Medical images were sourced from publicly available academic datasets.
- **What procedures were used?**
  A semi-automated pipeline was used: LLM-based drafting, technical trajectory construction, and clinician validation. Tool executions were logged and verified using structured interfaces.
- **Who was involved and how were they compensated?**
  The dataset was constructed by researchers, graduate students, and clinician collaborators.
- **Timeframe?**
  Data collection and curation were performed during **2025–2026**.
- **Ethical review?**
  All inputs come from public datasets approved for research use. No identifiable patient data is included.

A.4 Preprocessing / Cleaning / Labeling

- **Was preprocessing done?**
  Yes. Queries and trajectories were refined through technical verification and clinician validation. Redundant or unstable steps were removed, and tool arguments were standardized.
- **Was raw data preserved?**
  There is no separate raw trajectory dataset; trajectories are purpose-built. Medical images originate from existing datasets.
- **Is preprocessing software available?**
  Annotation and logging were performed using internal tools and standard research environments.

A.5 Uses

- **Has the dataset been used?**
  No prior use beyond this work.

- **Is there a repository?**
  An anonymized repository is released with the submission
- **Other potential uses?**
  MedCTA can be used for evaluating multimodal reasoning, tool-augmented agents, trajectory-level evaluation, and safe clinical decision-support research.
- **Any limitations affecting use?**
  Evaluation assumes access to compatible tool interfaces and structured reasoning outputs.
- **Potential negative impacts?**
  Misuse may lead to overestimating model readiness for clinical deployment. The dataset is intended strictly for research.

A.6 DISTRIBUTION

- **Will it be shared?**
  Yes, for academic research use.
- **How?**
  Via a public GitHub repository.
- **License?**
  Academic research license.
- **IP restrictions?**
  No.

A.7 MAINTENANCE

- **Who maintains it?**
  The authors.
- **Contact?**
  Via the corresponding author.
- **Updates?**
  Yes, including corrections, new tasks, and tool improvements.
- **External contributions?**
  Supported via GitHub issues and pull requests.

Table 7: Detailed Definitions of Tools in MedCTA

Name	Description	Input	Output
OCR	Extracts all visible text from an image along with spatial grounding when applicable.	Image	Text (optionally with bounding boxes)
ImageDescription	Generates a holistic natural language description summarizing the visual content.	Image	Image caption / summary
RegionAttributeDescription	Produces fine-grained descriptions of specified attributes within a given image region.	Image, bounding box, attribute (optional)	Region-level attribute description
GoogleSearch	Retrieves relevant external knowledge for a given query using a web search interface.	Query, top-k (optional)	Ranked search results
Calculator	Evaluates mathematical expressions for symbolic and numerical reasoning (restricted execution environment).	Text expression	Computed result

B Executable Tool Library.

MedCTA integrates a compact library of executable multimodal tools that enable structured, step-wise reasoning over clinical inputs. As summarized in Table 7, the toolset consists of five core components spanning perception, grounding, retrieval, and computation: `OCR`, `ImageDescription`, `RegionAttributeDescription`, `GoogleSearch`, and `Calculator`.

The perception module includes `OCR` for extracting textual content from medical images and documents, and `ImageDescription` for generating holistic semantic summaries of visual inputs. For fine-grained spatial reasoning, `RegionAttributeDescription` enables localized attribute extraction conditioned on specified image regions, supporting tasks such as lesion characterization or anatomical detail analysis. External knowledge retrieval is handled by `GoogleSearch`, which provides access to up-to-date medical and general-domain information when required for reasoning beyond the input context. Finally, `Calculator` supports symbolic and numerical computation, enabling quantitative reasoning (e.g., dosage calculation, measurement comparison).

Each tool is defined by a standardized input–output interface, allowing tool invocations to be executed deterministically and logged in structured JSON format. This explicit interface design facilitates fine-grained evaluation along multiple dimensions, including tool selection accuracy, argument validity, intermediate evidence consistency, and overall task correctness.

Importantly, within the MedCTA evaluation framework, tool usage is mediated through a Lagent Team (2023b)-based agent wrapper integrated with OpenCompass Contributors (2023). Depending on the evaluation setting, tools are either (i) exposed as a static tool schema via `tool_meta` for step-by-step evaluation without execution (`every_with_gt`), or (ii) accessed through a deployed tool server via `tool_server` for end-to-end execution (`every`). While the reference tool chain π specifies a sufficient set of tools to solve each task, the available subset $\mathcal{U} \subseteq \mathcal{D}$ and the invocation order remain hidden from the agent at test time. This design requires the agent to perform autonomous planning, implicit tool selection, and multi-step reasoning under partial observability.

C Clinician Verification Protocol and Inter-Annotator Agreement

This appendix describes the clinician verification protocol used to construct and validate `MedCTA`. Because the benchmark evaluates medical tool agents, correctness is not limited to the final answer: the intermediate reasoning trajectory must also be clinically plausible, grounded in the available evidence, and consistent with real diagnostic workflows. We therefore use clinician review to verify both outcome correctness and process-level validity.

C.1 Benchmark sample structure

Each benchmark instance contains:

- a de-identified medical input, such as a radiology image, pathology image, clinical figure, report, or multimodal medical file;
- a clinically realistic, step-implicit task query;
- a structured executable trajectory consisting of tool calls, arguments, observations, and intermediate reasoning;
- a final clinically validated answer, with multiple acceptable answers when appropriate.

The goal of this design is to evaluate whether a model can solve a clinical task through a plausible tool-mediated reasoning process, rather than merely guessing a correct final response.

Clinician reviewers were blinded to model predictions during dataset verification. This prevents model outputs from influencing the gold trajectory or final answer annotations.

C.2 Verification protocol

Each benchmark instance was reviewed along four dimensions.

Image and input validity. Clinicians verified that the input was medically interpretable, that the modality and anatomical region were correctly specified, and that image quality was sufficient for the intended task. Inputs with severe ambiguity, insufficient quality, or misleading artifacts were corrected or excluded.

Task goal correctness. Clinicians assessed whether the task query was clinically meaningful, realistic, and answerable from the provided input and available tool library. Queries that exposed the tool sequence explicitly, contained unsupported assumptions, or asked clinically unstable questions were revised.

Trajectory plausibility. Clinicians examined whether the reference tool trajectory followed a plausible diagnostic workflow. This included verifying that the trajectory used appropriate anatomical landmarks, visual findings, OCR evidence, external knowledge when needed, and localized region descriptions where clinically relevant. Trajectories containing hallucinated observations, unsupported shortcuts, redundant steps, or non-clinical reasoning were corrected.

Final answer accuracy. The final answer was checked against the visible evidence, tool observations, and accepted clinical interpretation. For tasks with non-unique valid descriptions, clinicians allowed multiple acceptable reference answers as long as they preserved the same clinical meaning.

Samples failing any verification criterion were either corrected through adjudication or removed from the benchmark.

C.3 Agreement protocol

To quantify annotation reliability, a stratified subset of samples was independently reviewed by multiple clinicians. The subset was selected to cover different modalities, anatomical regions, tool types, and trajectory lengths. Each clinician independently judged:

- whether the medical input was valid and interpretable;
- whether the task goal was clinically realistic and answerable;
- whether the reference trajectory was clinically plausible;
- whether the final answer was clinically correct.

Disagreements were resolved only after computing agreement statistics, so the reported agreement reflects independent judgments rather than post-hoc consensus. In addition, we perform a separate clinician audit of model rollouts in Appendix C.6, which evaluates whether the automated clinical-reasoning judge aligns with human clinical assessment on F_{acc}, C_s, and S_{comp}.

C.4 ETHICAL CONSIDERATIONS

All medical inputs used in MedCTA are de-identified and contain no protected health information. The benchmark is intended for research evaluation of medical AI systems and is not intended for clinical deployment. The dataset should not be used to make patient-level decisions. Any institutional or ethics review requirements were handled according to local policy for de-identified retrospective or publicly available medical data.

C.5 HUMAN ANNOTATION EFFORT

The construction of MedCTA required approximately **321 human annotation hours**, combining technical annotation, trajectory construction, and clinician verification. This effort was distributed across three stages.

Sample selection and query drafting (72 hours). Annotators first selected medically interpretable samples from public clinical and medical VQA sources, filtered ambiguous or low-quality cases, and converted perception-level questions into clinically grounded, step-implicit task goals. This stage also included removing tool-leaking phrasing and drafting initial task descriptions suitable for agentic evaluation.

Tool-chain construction and technical annotation (164 hours). Annotators then constructed executable tool trajectories for each task. This involved inspecting the medical inputs, selecting the required tools, specifying tool arguments, executing tool calls, recording observations, and revising trajectories for schema validity, minimality, and stability. LMM-generated drafts were used only as initialization; humans corrected redundant steps, invalid arguments, brittle localization, and unsupported reasoning.

Clinical verification and finalization (85 hours). Clinician reviewers verified the medical correctness of the final queries, reference trajectories, tool outputs, and final answers. This stage ensured that each trajectory followed a plausible clinical workflow, that intermediate evidence supported the conclusion, and that ambiguous or clinically unstable samples were corrected or removed. Finalized samples were then checked for formatting consistency and JSON executability.

C.6 CLINICIAN AUDIT OF MODEL ROLLOUTS

Motivation. MedCTA uses automated rubric-based evaluation to scale clinical reasoning assessment over many model rollouts. Because the benchmark is intended to diagnose clinically grounded tool use rather than only final-answer matching, we additionally performed a clinician audit on a subset of model trajectories. The goal of this audit is to test whether the automated clinical-reasoning metrics reflect clinically meaningful judgments made by a human reviewer.

Protocol. We evaluated 36 benchmark tasks for four representative models: GPT-5.4, Claude-opus-4-6, Gemini-3-flash, and Qwen3.5-9B. This yields 144 model–task rollouts. For each rollout, the clinician assigned scores for the same three clinical-reasoning dimensions used in the main evaluation: F_{acc} for clinical faithfulness, C_s for context integration, and S_{comp} for semantic completeness. Each dimension was scored on the same ordinal rubric $\{0, 0.1, 0.4, 0.7, 1.0\}$, where higher values indicate stronger clinical validity, evidence use, or completeness. We report all scores on a 0–100 scale for consistency with the main paper.

Human audit results. Table 8 summarizes the clinician scores. The audit confirms that the clinical-reasoning bottleneck remains substantial: the overall mean score across all 144 evaluated rollouts is only 28.5/100. GPT-5.4 obtains the highest human clinical-reasoning score, but still reaches only 37.9/100 on average. Semantic completeness is the weakest dimension overall (25.1/100), suggesting that models often omit required clinical findings even when parts of the reasoning chain are plausible.

Agreement with the automated clinical judge. We next compare the clinician audit with the automated LLM-as-judge scores reported in the main benchmark table for the same four model families. Since the uploaded audit contains human scores but not per-sample automated scores for the

Table 8: Clinician audit of model rollouts on 36 MedCTA tasks. Scores are percentages. "Mean" is the average of F_{acc}, C_s, and S_{comp} per rollout. The bracketed interval is a normal-approximation 95% confidence interval for the per-task mean score. The last two columns report the percentage of rollouts whose mean score is at least 0.4 or 0.7, respectively.

Model	N	F_{acc}	C_s	S_{comp}	Mean	Mean ≥ 0.4	Mean ≥ 0.7
GPT-5.4	36	38.3	39.7	35.6	37.9 [29.2, 46.5]	41.7	11.1
Claude-opus-4-6	36	32.2	30.3	23.1	28.5 [21.6, 35.4]	33.3	2.8
Gemini-3-flash	36	29.2	21.9	26.7	25.9 [17.0, 34.9]	25.0	11.1
Qwen3.5-9B	36	23.6	26.1	15.0	21.6 [16.2, 27.0]	11.1	2.8
Average	144	30.8	29.5	25.1	28.5	27.8	6.9

Table 9: Model-level agreement between the clinician audit and the automated clinical-reasoning judge. MAE is reported in percentage points.

Dimension	Pearson r	Spearman ρ	MAE
Clinical Faithfulness (F_{acc})	0.97	1.00	17.5
Context Integration (C_s)	0.40	0.20	15.8
Semantic Completeness (S_{comp})	0.33	0.40	7.5
Aggregate clinical score	0.62	0.60	13.5

same 36 examples, we report model-level agreement rather than per-instance inter-rater agreement. Table 9 shows that the aggregate clinical score has moderate model-level alignment between human and automated evaluation (Pearson $r = 0.62$, Spearman $\rho = 0.60$). Agreement is strongest for clinical faithfulness, where the human and automated judge induce the same model ranking (Spearman $\rho = 1.00$). Context integration and semantic completeness show weaker alignment, indicating that evidence-use and missing-finding judgments are more sensitive to judge calibration and subset composition.

Consistency among human clinical dimensions. The three human-scored dimensions are related but not redundant. Across all 144 rollouts, the three-dimensional clinical rubric has Cronbach's $\alpha = 0.79$, indicating good internal consistency. Pairwise correlations show that clinical faithfulness and semantic completeness are strongly related (Pearson $r = 0.78$, Spearman $\rho = 0.74$), while context integration is less tightly coupled with completeness (Pearson $r = 0.39$, Spearman $\rho = 0.34$). This supports the design choice of reporting separate clinical-reasoning dimensions: a rollout can follow a plausible workflow while still failing to use multimodal evidence, or it can use some evidence while omitting required clinical findings.

Table 10: Internal consistency of the clinician-scored clinical-reasoning dimensions across 144 audited rollouts. "Within one step" means that two scores differ by at most one adjacent rubric level.

Metric pair	Pearson r	Spearman ρ	Exact match	Within one step
F_{acc} vs. C_s	0.52	0.42	44.4	89.6
F_{acc} vs. S_{comp}	0.78	0.74	27.8	91.7
C_s vs. S_{comp}	0.39	0.34	34.7	68.8

Interpretation. The clinician audit strengthens the main finding of MedCTA. Even when judged by a human clinical reviewer, current tool agents rarely produce trajectories that are faithful, well-grounded, and complete. The best audited model remains far from saturation, and only 6.9% of all audited rollouts reach a strong mean clinical score. The audit also shows that automated evaluation is directionally useful but imperfectly calibrated: it broadly agrees with human judgment at the model level, especially for clinical faithfulness, while context integration and semantic completeness remain more difficult to judge automatically. We therefore use the automated clinical-reasoning metrics as scalable diagnostics, not as a replacement for expert clinical review.

D INSTRUCTION FOR ANNOTATORS

Technical Annotation Guidelines for Executable Tool-Chain Verification

Objective: Ensure that each annotated reasoning trajectory is *executable, valid, and minimal* under the deployed tool library, while preserving correctness with respect to the clinical objective.

Verification Protocol:

- **Schema Compliance:** Verify that every tool invocation strictly adheres to the predefined interface, including correct argument names, data types, and formatting consistent with the tool specification.
- **Tool Validity:** Ensure that each selected tool belongs to the available tool set $\mathcal{D} = \{\texttt{OCR}, \texttt{ImageDescription}, \texttt{RegionAttributeDescription}, \texttt{GoogleSearch}, \texttt{Calculator}\}$ and is appropriate for the intended subgoal.
- **Argument Correctness:** Check that all inputs are both syntactically valid and semantically grounded in the available evidence (e.g., image content, prior tool outputs, or retrieved knowledge).
- **Execution Robustness:** Identify and eliminate brittle or failure-prone steps (e.g., unnecessary region specifications, ambiguous OCR dependencies, or redundant intermediate queries) that may reduce reproducibility.
- **Minimality and Efficiency:** Remove redundant, circular, or over-decomposed steps. Each tool call should contribute directly to advancing the reasoning toward the final answer.
- **Trace Consistency:** Ensure that the trajectory forms a coherent chain of reasoning, where each step s_i logically follows from preceding outputs and maintains consistency of intermediate evidence.
- **Deterministic Executability:** Confirm that the full trajectory can be executed sequentially without ambiguity, missing dependencies, or undefined intermediate variables.

Constraints:

- Do not introduce new tools beyond the defined library.
- Do not alter the original clinical question or intended outcome.
- Avoid artificial expansion of reasoning steps that do not improve correctness or clarity.

Deliverable: Produce a corrected executable trajectory $\pi = \{s_i\}_{i=1}^{m}$, consisting of a minimal sequence of valid tool calls with well-formed arguments and consistent intermediate outputs.

Figure 5: Annotation protocol for validating executable tool-based reasoning trajectories in MedCTA.

Clinical Annotation Guidelines for Medical Reasoning Validation

Objective: Ensure that each reasoning trajectory reflects clinically sound, evidence-based decision-making and adheres to realistic medical workflows.

Validation Protocol:

- **Clinical Soundness:** Verify that intermediate tool outputs correspond to medically meaningful and plausible findings (e.g., anatomically consistent observations, valid clinical interpretations).
- **Reasoning Completeness:** Ensure that all critical diagnostic or interpretive steps required for the task are present, without omission of key clinical reasoning stages.
- **Logical Coherence:** Confirm that the trajectory follows a coherent clinical reasoning pattern (e.g., observation $\rightarrow$ interpretation $\rightarrow$ conclusion), consistent with standard diagnostic workflows.
- **Evidence Attribution:** Check that each conclusion is explicitly grounded in prior evidence, including visual findings, extracted text, or retrieved knowledge, and that the final answer $\mathcal{A}$ is fully supported by intermediate steps.
- **Safety and Reliability:** Identify and eliminate unsupported assumptions, hallucinated findings, or clinically unsafe inferences that could lead to incorrect or misleading conclusions.
- **Uncertainty Handling:** Ensure that ambiguous or insufficient evidence is appropriately reflected in the reasoning, avoiding overconfident conclusions when clinical certainty is not justified.

Correction Guidelines:

- Revise incorrect or clinically implausible interpretations.
- Insert missing but necessary diagnostic reasoning steps.
- Remove steps that contradict established medical knowledge or evidence.
- Flag trajectories with unresolved ambiguity or potential clinical risk.

Deliverable: Produce a clinically validated trajectory and assign an outcome label indicating whether the final answer is (i) correct, (ii) partially correct, or (iii) requires revision, based on consistency with clinical evidence and reasoning.

Figure 6: Clinical validation protocol for assessing medical correctness and reasoning quality in `MedCTA` trajectories.

E CASE STUDIES: STEPWISE CONSTRUCTION OF EXECUTABLE CLINICAL TOOL TRAJECTORIES

To illustrate how `MedCTA` examples are constructed, corrected, and validated, Figures 7 and 8 present two representative case studies that trace the full evolution of a benchmark sample from its original dataset form to its final executable trajectory. In both cases, the construction process is organized into five stages: (1) the original dataset instance and reference question-answer pair, (2) the initial GPT-generated reasoning and tool sequence, (3) technical verification of tool validity and execution logic, (4) clinical review and medical validation, and (5) the final executable trajectory retained in the benchmark.

E.1 TRAJECTORY FOR HISTOLOGICAL PATTERN INFERENCE

Figure 7 shows a histology example in which the original sample contained multiple questions, including one asking what type of cells are depicted across the panels. During review, the clinical team identified that the earlier answer introduced biological context that was not directly inferable from the image alone, such as proximal colon specificity, colonization status, and experimental reconstitution details. This motivated a revision of both the question-answer interpretation and the generated trajectory. The updated pipeline was redesigned to ground the response only in visible evidence from the figure itself, namely the panel labels *Naive*, *Th1*, and *Th2*, together with the observed gut mucosal histology. Technically, the revised trajectory uses `OCR` to extract the panel labels, `ImageDescription` to characterize the overall histologic context, and `RegionAttributeDescription` to confirm the presence of structural and inflammatory differences across the labeled regions. After technical verification, the new GPT-generated trajectory was explicitly handed off for clinical review, where unsupported claims were removed and the final answer was approved as image-grounded. This example illustrates how the pipeline corrects over-interpretation by forcing both the reasoning trace and the final answer to remain tied to directly observable evidence.

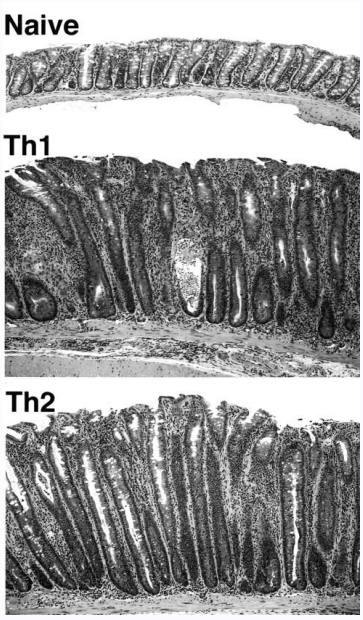
E.2 TRAJECTORY FOR SENTINAL NODE METASTASIS DETECTION

Figure 8 highlights a different but equally important failure mode: under-justified inference from figure labels to clinical methods. In this case, the original question asked which methods were used to detect metastases in sentinel nodes. The figure visibly contained assay labels such as *H&E*, *CK-18*, *CEA*, *hTRT*, and *MUC-1*, and the initial GPT-generated trajectory attempted to infer the underlying detection methods from these labels using only image-based tools. During technical and clinical review, it was determined that this inference could not be justified from the image alone, because mapping these labels to histopathology and molecular detection workflows requires external biomedical knowledge. To address this, the trajectory was revised by inserting an explicit `GoogleSearch` step after OCR extraction of the assay labels. This search was used to establish that H&E corresponds to conventional histopathology, while CK-18, CEA, hTRT, and MUC-1 are biomarkers used in molecular assays such as RT-PCR and related nucleic acid-based methods. The placement of this search step was then further refined so that the trajectory first extracts visible evidence, then retrieves the necessary domain knowledge, and finally verifies that the figure indeed compares histologic and molecular detection approaches. The clinically validated final answer therefore reflects both what is shown in the chart and what must be supplied through justified external knowledge.

These examples illustrate the purpose of the staged `MedCTA` construction process. The initial GPT trajectory is treated as a draft hypothesis, not a final solution. It is audited from two perspectives: a technical review ensures tool validity, executable inputs, and coherent sequencing, while a clinical review checks that reasoning and answers are medically sound and grounded in visual evidence. When issues arise, the question, answer, and tool chain are revised and re-validated. This workflow ensures high-quality examples with not just correct answers, but executable, clinically grounded reasoning paths, allowing the benchmark to evaluate both task accuracy and the ability to produce valid multi-step tool trajectories.

Example 1: Anatomical Variant Identification

Stage 1: Original Dataset Sample



11781367_fig4.jpg

Question 1: *What type of cells are shown in the image?*
Reference Answer: Proximal colon tissue with naïve T cells, Th1 cells, and Th2 cells

Question 2: *What is the magnification of the image?*
Reference Answer: 100×

Question 3: *What are the characteristics of the colon tissue in the Th1 and Th2 groups?*
Reference Answer: Inflammatory infiltrates, epithelial hyperplasia, mucin depletion

Stage 2: Initial GPT-Generated Trajectory

Agentic Query: For the provided histology figure, determine the stated original magnification (e.g., 40×/100×/400×) by inspecting any scale/magnification annotation on the panel. Report the magnification exactly as written?
Predicted Tools: `ImageDescription, OCR, RegionAttributeDescription`
Draft Reasoning Plan:

1. Inspect the histology panel to identify any visible magnification or scale annotation.
2. Extract text from the image to locate magnification information.
3. Verify potential annotation regions if magnification is not directly visible.

Draft Tool Trace:

- `ImageDescription(image)` → "Histology figure with panels labeled Naive, Th1, and Th2; no clear magnification visible."
- `OCR(image)` → "Naive, Th1, Th2" (no magnification text found)
- `RegionAttributeDescription(upper region)` → "No magnification text present"
- `RegionAttributeDescription(lower region)` → "No magnification text present"

Draft Answer: 100×

Stage 3: Technical Annotation and Tool-Chain Verification

Verification Focus:

- Confirm that only tools from the deployed library are used.
- Ensure that tool calls are minimal and executable.
- Check whether any redundant or unsupported intermediate steps are present.

Technical Assessment:

- **Tool validity:** Valid. `OCR`, `ImageDescription`, and `RegionAttributeDescription` are part of the deployed tool set.
- **Schema compliance:** Valid. All tool inputs are well-formed and correctly specified.
- **Minimality:** Valid. OCR is required to extract panel labels, and additional tools support contextual verification.
- **Execution robustness:** Valid. The trajectory avoids unsupported steps and ensures reliable extraction and validation.

Correction Applied: The original question and answer were revised based on clinical feedback to remove unsupported contextual details. A new trajectory was generated that grounds the answer strictly in observable image evidence (panel labels and histological patterns), resulting in a corrected and executable pipeline.

Stage 3.5: Clinical Review Handoff

Post-Technical Step: Clinical Review Handoff

- The updated GPT-generated trajectory is forwarded to clinical experts for validation.
- Clinical reviewers assess whether the reasoning and final answer are strictly grounded in visible image evidence.
- Any remaining inconsistencies or unsupported assumptions are flagged for correction.
- Upon successful validation, the trajectory is approved for final use.

Stage 4: Clinical Annotation and Medical Validation

Clinical Review Focus: Clinical Review Focus:

- Verify that the answer is grounded only in information directly visible in the histology figure.
- Confirm that the panel labels are sufficient to identify the depicted cell conditions.
- Ensure that no unsupported biological context is introduced beyond the image evidence.

Clinical Assessment:

- The panel labels clearly indicate the conditions **Naive**, **Th1**, and **Th2**.
- The histology image supports describing the tissue as gut mucosal tissue with these labeled conditions across panels.
- The earlier answer included unsupported details such as proximal colon, *pnir15.OVA-E. coli* colonization, and reconstitution context, which cannot be directly inferred from the image alone.
- The corrected answer, **"The sections display the naive T cells, Th1 cells, and Th2 cells on the gut mucosal tissue."**, is clinically appropriate and image-grounded.

Clinical Verdict: Correct after answer revision.

Stage 5: Final Executable Trajectory

Final Query: Using the provided histology figure, identify what type of cells/conditions are depicted across the panels?
Final Tool Set: `OCR, ImageDescription, RegionAttributeDescription`
Executable Trajectory π:

1. **Tool:** `OCR`
   Input: histology image `(image/image_29.jpg)`
   Output: "Naive, Th1, Th2"

2. **Tool:** `ImageDescription`
   Input: histology image `(image/image_29.jpg)`
   Output: "Histology micrographs of gut mucosal tissue arranged in panels labeled Naive, Th1, and Th2, showing variations in cellular density."

3. **Tool:** `RegionAttributeDescription`
   Input: `bbox = (40, 360, 470, 610)`
   Output: "Differences in cellular infiltration corresponding to Naive, Th1, and Th2 conditions."

Final Answer

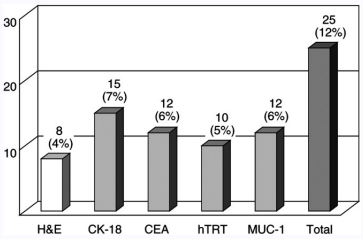
Naive T cells, Th1 cells, and Th2 cells

Final Justification: OCR identifies the panel labels (Naive, Th1, Th2), and visual inspection confirms corresponding histological differences in gut mucosal tissue, supporting that the sections depict these T cell conditions.

Figure 7: Stepwise construction of an executable and clinically validated trajectory for anatomical variant identification.

Example 2: Sentinel Node Metastasis Detection Methods

Stage 1: Original Dataset Sample



`12915877_fig2.jpg`

Question 1: *What are the methods used to detect metastases in the image?*
Reference Answer: The image shows three methods used to detect metastases in sentinel nodes: conventional histology, hematoxylin and eosin (H&E) staining, and RT-PCR and Southern blot assay.

Question 2: *What are the markers mentioned in the image?*
Reference Answer: CK-18, CEA, and hTRT

Question 3: *How many nodes were harvested from the patients?*
Reference Answer: 25 nodes

Stage 2: Initial GPT-Generated Trajectory

Predicted Tools: `ImageDescription, OCR, GoogleSearch, RegionAttributeDescription`

Draft Reasoning Plan:

1. Inspect the figure to determine whether it presents histologic and molecular detection methods.
2. Extract the biomarker and assay labels shown in the bar chart.
3. Use external medical knowledge to map the labels to the corresponding laboratory methods.
4. Verify from the figure that the chart compares histologic and molecular detection approaches.

Draft Tool Trace:

- `ImageDescription(image)` → "A bar chart comparing detection rates using H&E, CK-18, CEA, hTRT, and MUC-1 markers."
- `OCR(image)` → "H&E, CK-18, CEA, hTRT, MUC-1"
- `GoogleSearch(''What methods are used to detect sentinel lymph node metastases using H&E CK-18 CEA hTRT MUC-1'')` → "H&E is used for conventional histopathology. CK-18, CEA, hTRT, and MUC-1 are commonly detected using molecular techniques such as RT-PCR or related nucleic acid-based assays to identify micrometastases."
- `RegionAttributeDescription(region)` → "The figure compares H&E with biomarker-based detection methods, indicating both histological and molecular approaches."

Draft Answer: The figure shows that metastases in sentinel lymph nodes are detected using conventional histopathology (hematoxylin and eosin, H&E) and molecular techniques such as RT-PCR using biomarkers including CK-18, CEA, hTRT, and MUC-1.

Stage 3: Technical Annotation and Tool-Chain Verification

Verification Focus:

- Confirm that only tools from the deployed library are used.
- Ensure that tool calls are executable and logically ordered.
- Check whether external knowledge is required to infer laboratory methods from the biomarker labels.
- Remove unsupported assumptions and improve grounding between image evidence and final answer.

Technical Assessment:

- **Tool validity:** Valid. `ImageDescription`, `OCR`, `GoogleSearch`, and `RegionAttributeDescription` are all part of the deployed tool set.
- **Schema compliance:** Valid. All tool inputs are correctly formed and executable.
- **Minimality:** Revised. The original trajectory inferred laboratory methods directly from the chart labels, which was insufficiently grounded. `GoogleSearch` was added to support mapping H&E and biomarker labels to detection methods.
- **Execution robustness:** Valid after revision. The updated trajectory first extracts visible labels, then uses external knowledge, and finally verifies that the figure compares histologic and molecular approaches.

Correction Applied: Based on clinical feedback, a `GoogleSearch` step was inserted after OCR to justify the inference from biomarker labels to the underlying detection methods. The updated trajectory uses the clinician-provided query and revised reasoning step, and the placement of the search step was further corrected to maintain a logically grounded tool chain.

Stage 3.5: Clinical Review Handoff

Post-Technical Step: Clinical Review Handoff

- The revised GPT-generated trajectory is forwarded to clinical experts for validation.
- Clinical reviewers assess whether the inferred detection methods are adequately supported by the visible chart labels and the external search evidence.
- The placement and necessity of the `GoogleSearch` step are specifically reviewed.
- Any remaining unsupported assumptions are flagged and corrected before final approval.

Stage 4: Clinical Annotation and Medical Validation

Clinical Review Focus:

- Verify that the figure alone shows assay labels rather than explicitly naming all laboratory methods.
- Confirm that external medical knowledge is needed to connect H&E and the listed biomarkers to histopathology and molecular assays.
- Ensure that the final answer remains clinically accurate without overstating what is directly visible in the image.

Clinical Assessment:

- The figure visibly presents the labels **H&E**, **CK-18**, **CEA**, **hTRT**, and **MUC-1**.
- These labels alone do not fully specify the laboratory detection methods without domain knowledge.
- Adding an external search step is clinically appropriate because H&E corresponds to conventional histopathology, whereas CK-18, CEA, hTRT, and MUC-1 are biomarkers commonly used in molecular detection workflows such as RT-PCR.
- The revised answer is clinically acceptable because it distinguishes between histopathologic detection and molecular biomarker-based detection while remaining consistent with the figure content.

Clinical Verdict: Correct after trajectory revision

Stage 5: Final Executable Trajectory

Final Tool Set: `ImageDescription, OCR, GoogleSearch, RegionAttributeDescription`
Executable Trajectory π:

1. **Tool:** `ImageDescription`
   Input: figure image `(image/image_55.jpg)`
   Output: "A bar chart comparing detection rates using H&E, CK-18, CEA, hTRT, and MUC-1 markers."

2. **Tool:** `OCR`
   Input: figure image `(image/image_55.jpg)`
   Output: "H&E, CK-18, CEA, hTRT, MUC-1"

3. **Tool:** `GoogleSearch`
   Input: "What methods are used to detect sentinel lymph node metastases using H&E CK-18 CEA hTRT MUC-1"
   Output: "H&E is used for conventional histopathology. CK-18, CEA, hTRT, and MUC-1 are commonly detected using molecular techniques such as RT-PCR or related nucleic acid-based assays to identify micrometastases."

4. **Tool:** `RegionAttributeDescription`
   Input: `bbox = (40, 220, 580, 280)`
   Output: "The figure compares H&E with biomarker-based detection methods, indicating both histological and molecular approaches."

Final Answer

conventional histology, hematoxylin and eosin (H&E) staining, and RT-PCR and Southern blot assay

Final Justification: The chart explicitly shows H&E and the biomarker labels CK-18, CEA, hTRT, and MUC-1. External medical knowledge is required to map these labels to their corresponding detection methods. Together, the figure and the search evidence support that the figure compares conventional histopathologic detection with molecular biomarker-based detection of sentinel node metastases.

Figure 8: Stepwise construction of an executable and clinically validated trajectory for identifying metastasis detection methods in sentinel nodes.

ReAct-Style Execution Prompt Used in the Agent System

```
CALL_PROTOCOL_EN = """You are a assistant who can utilize
external tools. {tool_description}
To use a tool, please use the following format:
```
{thought}Think what you need to solve, do you need to use
tools?
{action}the tool name, should be one of [{action_names}]
{action_input}the input to the action
```
The response after utilizing tools should using the following
format:
```
{response}the results after call the tool.
```
If you already know the answer, or you do not need to use
tools, please using the following format to reply:
```
{thought}the thought process to get the final answer
{finish}final answer
```
Begin!"""
```

Figure 9: ReAct-style execution prompt used in the MedCTA agent system, adapted from the GTA Wang et al. (2024b)-style tool-use protocol. The prompt specifies the action format, tool input schema requirement, observation format, and final-answer format.

F Prompting Templates for Agent Trajectory Construction

To ensure reproducible agent behavior, we employ two complementary prompting components: (1) a *ReAct-style execution prompt* that governs how the agent reasons, invokes tools, and processes observations during inference, and (2) a *trajectory generation prompt* used to synthesize a single executable ground-truth trajectory for each medical VQA sample. Following the GTA framework Wang et al. (2024b), the agent operates in a structured reasoning-and-acting loop, interleaving concise reasoning steps, explicit tool calls, tool responses, and a final answer, ensuring that tool usage is integrated into a coherent and interpretable decision process. For benchmark construction, the generation prompt enforces strict constraints on output format, requiring a valid JSON trajectory with minimal and executable tool usage, realistic tool outputs, and exact preservation of the original answer, while encouraging clinically grounded strategies such as early OCR for labels, image-based contextualization, and region-level verification. Figures 9 and 10 illustrate these two prompts, which together define both the runtime behavior of the agent and the standardized procedure for constructing clinically valid, executable trajectories.

The ReAct-style execution prompt governs how the agent interacts with external tools at inference time by structuring the process into explicit reasoning, tool selection, tool input, and observation steps, making the decision process modular, interpretable, and auditable, an essential requirement in clinical settings. In contrast, the trajectory generation prompt is used offline to construct benchmark examples, enforcing strict JSON structure, executable tool usage, realistic outputs, and exact preservation of the original answer, ensuring that each trajectory serves as a valid reference. Both prompts are necessary: the execution prompt enables consistent runtime reasoning and tool interaction, while the generation prompt standardizes benchmark construction and comparability across samples. This dual design is particularly important in medical applications, where prompt quality directly impacts trajectory reliability, encouraging grounded reasoning, justified tool usage, and clinically valid, verifiable outcomes.

Prompt for Generating Ground-Truth Agent Trajectories

```
SYSTEM_PROMPT = r"""
You generate ONE agentic ground-truth trajectory for a medical VQA
sample.

You will receive:
- toolmetadata (dict keyed by tool name),
- one medical image,
- original_question,
- original_answer,
- sample_key.

You MUST output ONE valid JSON object and nothing else.
The JSON must have EXACTLY ONE top-level key = sample_key.
That key maps to an object with EXACT keys:
- tools: list[tool_spec] (subset of provided toolmetadata; include
ONLY tools actually used)
- files: list of file objects (must include the given image path)
- dialogs: list of messages (role=user/assistant/tool)
- gt_answer: {"whitelist": [[original_answer_exact]], "blacklist": null}

HARD FORMAT RULES
A) Tool calls MUST look exactly like:
  {"role":"assistant","tool_calls":[{"type":"function","function":
  {"name":"ToolName","arguments":{...}}}, ...],"thought":"..."}
B) After each tool call, add a tool message:
  {"role":"tool","name":"ToolName","content":{"type":"text","content":"
  <string>"}}
   - tool content.type MUST be "text" always
   - tool content.content MUST be a string (even for ints)
C) The FINAL message in dialogs MUST be:
  {"role":"assistant","content":"<EXACT original_answer>"}
  (character-for-character)

CRITICAL CONSTRAINTS
- Minimum tool calls = 2 (use more only if truly needed; 2{6 typical).
- Keep a clinician/radiologist/pathologist style of reasoning: brief,
grounded, stepwise.
- The rewritten question should be answerable with the tools you use
(agentic, but not silly).
- Tool outputs must be realistic:
   - OCR: "(x1, y1, x2, y2) TEXT" lines separated by "\n" (empty
   allowed if no text)
   - RegionAttributeDescription: short attribute grounded on provided
   bbox  (numeric coordinates)
   - ImageDescription: brief factual description
   - GoogleSearch: 3{6 short result lines
   - Calculator: numeric string only
   - exact preservation of the original answer (answer whitelist)

ADAPTIVE BEST-CLINICAL STRATEGY (choose per sample)
1) If there might be labels/markers/measurements: use OCR early.
2) Use ImageDescription to orient (modality, anatomy, figure type).
3) Use RegionAttributeDescription with a plausible numeric bbox to
verify akey structure/finding.
4) Optionally add a sanity-check step (e.g., another
RegionAttributeDescription or GoogleSearch if the question is
definitional).
5) End with the exact original answer string.

Return valid JSON only.
"""
```

Figure 10: Prompt template used to synthesize a single executable ground-truth trajectory for each medical VQA sample. The prompt enforces strict JSON structure, executable tool usage, realistic tool outputs, and exact preservation of the reference answer.

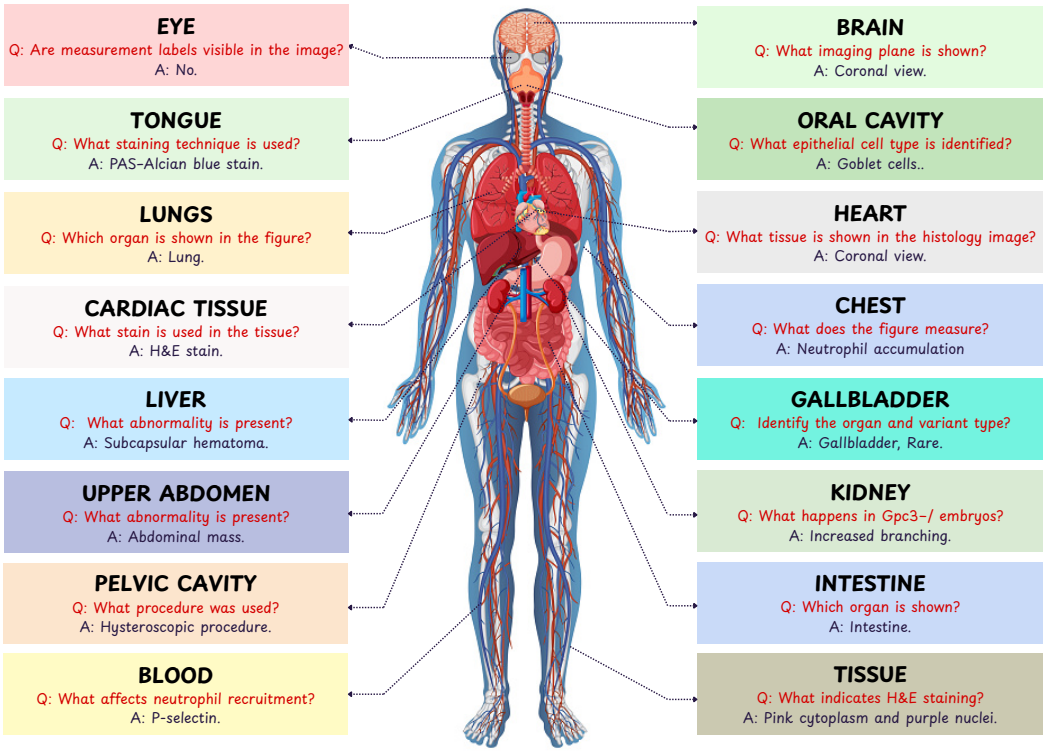


Figure 11: **Anatomical and modality coverage of MedCTA.** The benchmark spans diverse body regions and clinical inputs, including eye, brain, oral cavity, lungs, heart, liver, kidney, intestine, gallbladder, pelvis, blood, and tissue-level pathology. Each example shows a step-implicit clinical query and its target answer, illustrating that MedCTA evaluates broad cross-specialty medical tool use rather than a single organ system or imaging modality.

G Metrics and Implementation Details

Hardware and Setup. All experiments were conducted on NVIDIA A100 GPUs (80GB), ensuring consistent hardware conditions across models. Each evaluated `LMM` agent receives the clinical context $\mathcal{X}$ and query $\mathcal{Q}$, and interacts with the executable tool library $\mathcal{D}$ under a standardized agent framework. Tool invocation follows a structured ReAct-style prompting protocol, enabling step-wise reasoning and deterministic execution.

Tool Call Execution. All tools are implemented as callable Python functions with strictly defined input–output schemas. Model-generated reasoning traces are parsed as structured JSON, and tools are executed sequentially according to the predicted actions. A tool call is considered *valid* if (i) the selected tool exists in $\mathcal{D}$, (ii) the arguments conform to the required schema, and (iii) execution produces a non-empty output. Invalid formatting, incorrect tool selection, or argument mismatches are recorded as execution failures and incorporated into evaluation statistics.

Figure 11 summarizes the clinical breadth of `MedCTA`. The benchmark covers whole-body anatomy and tissue-level pathology, with tasks drawn from radiology, histology, gross pathology, ophthalmology, and report-based clinical reasoning. Importantly, the queries are not simple recognition prompts: they ask agents to identify findings, interpret stains, recognize variants, reason over measurements, and connect localized evidence to clinically meaningful answers. This diversity ensures that `MedCTA` evaluates general medical tool-agent behavior rather than overfitting to one modality, organ system, or question type.

Metrics Overview. As summarized in Table 3, MedCTA employs a multi-level evaluation framework consisting of three modes:

- **Step-by-Step Mode:** Evaluates intermediate tool-use behavior using *InstAcc*, *ToolAcc*, *ArgAcc*, and *SummAcc*, measuring protocol adherence, tool selection correctness, argument validity, and step-level summarization quality.
- **Clinical Reasoning Mode:** Assesses trajectory-level coherence and medical grounding via *Clinical Faithfulness* (F_{acc}), *Context Integration Score* (C_s), and *Semantic Completeness* (S_{comp}).
- **Outcome Mode:** Measures end-to-end correctness using *Goal Accuracy* (G_{acc}).

Metric scaling and aggregation. All reported metrics are normalized to the range $[0, 100]$, where higher is better. Scores are macro-averaged over tasks unless otherwise stated. Premature final answers before the reference stop point are counted as under-calls and receive zero credit for the missing remaining step-level decisions.

Step-by-step metrics. Let $s_i = (t_i, \alpha_i, \rho_i)$ be the i-th reference step and let $\hat{s}_i = (\hat{t}_i, \hat{\alpha}_i, \hat{\rho}_i)$ be the model-predicted step.

- **InstAcc** measures whether the model emits a parseable action or final-answer message under the required protocol.
- **ToolAcc** is one if $\hat{t}_i = t_i$ and zero otherwise. If multiple tools are clinically equivalent for a task, the accepted set is specified in the task metadata.
- **ArgAcc** measures whether $\hat{\alpha}_i$ is executable and clinically appropriate. We assign full credit when all required keys, data types, and clinically relevant values match after canonicalization. We assign half credit when the call is executable and semantically usable but contains harmless extra keys, minor formatting drift, or spatial arguments that overlap the reference region sufficiently. Non-parseable, non-executable, wrong-file, or clinically inconsistent arguments receive zero credit.
- **SummAcc** measures whether the model correctly integrates the current observation into the expected intermediate conclusion. It is graded using the same rubric as clinical reasoning but restricted to the current step.

Clinical reasoning metrics. Reasoning metrics are graded from the executed trajectory only using a rubric calibrated with clinician annotations:

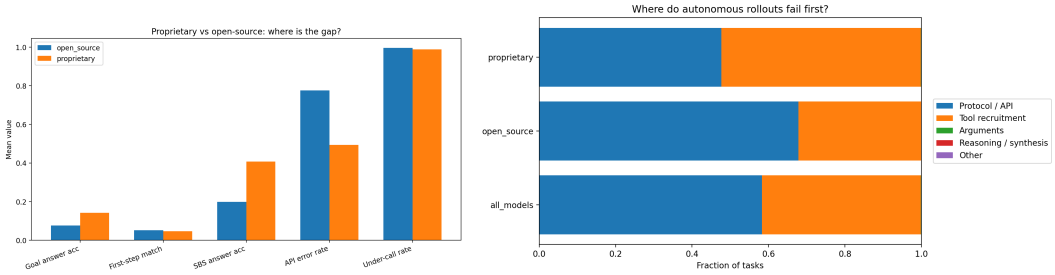
- **Clinical Faithfulness** (F_{acc}): whether reasoning claims are supported by tool observations and do not contradict the evidence.
- **Context Integration** (C_s): whether the trajectory integrates all relevant modalities and files rather than relying on a single cue or prior.
- **Semantic Completeness** (S_{comp}): whether all clinically necessary findings for the answer are covered.

Each dimension is scored on a 0–1 rubric and linearly mapped to $[0, 100]$. Scores are clipped to $[0, 100]$; negative values are not permitted.

Outcome metric. Goal accuracy (G_{acc}) measures final answer correctness against the clinician-validated answer whitelist. For closed-form answers, we use exact or canonicalized matching. For open-ended clinical answers, we use rubric-based semantic equivalence with clinician-calibrated adjudication.

H Additional Experimental Results and Diagnostic Analyses

This appendix keeps only the supplementary diagnostics referenced in the main text. We focus on the views that are most useful for interpreting benchmark behavior: a backbone-only zero-shot sanity check, the gold-standard tool routing gap, rollout failure dynamics, descriptive conditional diagnostics, horizon effects, and tool-level bottlenecks.



(a) Open-source vs. proprietary models under rollout diagnostics.

(b) Where autonomous rollouts fail first.

Figure 12: **Rollout diagnostics across models.** (Left) Proprietary models show lower API/protocol failures and stronger oracle answerability. (Right) Failures are dominated by protocol/API handling and tool recruitment.

Table 11: Autonomous call-count behavior (%). Only informative deviations from the dominant pattern are shown.

Model	Under-call	Exact match	Over-call
GPT_4o	91.6	**3.74**	4.67
Qwen3.5-9B	95.3	**4.67**	0.00
claude-haiku-4-5	99.1	0.00	0.93
gpt-5.4-mini	99.1	0.00	0.93
gemini-3.1-flash-lite-preview	99.1	0.00	0.93
all remaining models	100.0	0.00	0.00

H.1 GOLD-STANDARD TOOL ROUTING GAP AND CONTROLLER/READER SEPARATION

Figure 12a provides the same story at group level. Proprietary models outperform open-source models not mainly because of dramatically better first actions, but because they are more stable under the interaction protocol and better at exploiting the evidence once correct routing is provided.

H.2 AUTONOMOUS ROLLOUT DYNAMICS AND EARLY-FAILURE BEHAVIOR

Table 11 shows that autonomous rollouts are overwhelmingly dominated by *under-calling*. The only informative deviations from the default pattern come from a very small subset of models: GPT-4o and Qwen3.5-9B are the only models with non-trivial exact tool-count match, and over-calling is confined to GPT-4o and a few models at negligible rates. All remaining models under-call on essentially every task.

Figure 12b sharpens the diagnosis. First failures are concentrated almost entirely in protocol/API handling or tool recruitment, while downstream reasoning and synthesis contribute very little as *first* failure modes. This is the key reason the benchmark remains difficult: many agents fail before stable evidence-conditioned reasoning can even begin.

H.3 CONDITIONAL DIAGNOSTICS AND DESCRIPTIVE PARTIAL-ORACLE VIEWS

Table 12 and Figure 13 summarize which properties of autonomous rollouts tend to co-occur with higher final accuracy. The strongest positive descriptive signal is the subset of rollouts with schema-clean matched arguments, followed by the subset with no API/format error. By contrast, exact-sequence and long-prefix conditions have very small support and do not provide stable evidence of improvement.

These results should be read as *descriptive diagnostics*, not causal claims. They show which rollout properties are associated with success, and they motivate future partial-oracle interventions that provide only limited gold supervision instead of full teacher forcing.

Table 12: Observed (descriptive) answer uplift under different rollout conditions. These are associative (not causal).

Condition	Support	Rate	Baseline	If condition	Uplift
All models					
All matched arguments schema-clean	145	7.13	10.7	13.1	**+2.4**
No API / format error	728	35.81	10.7	12.1	**+1.4**
Exact tool-step count match	9	0.44	10.7	11.1	+0.4
First tool correct	128	6.30	10.7	8.6	**-2.1**
Matched gold prefix $\geq$ 2 steps	36	1.77	10.7	2.8	**-7.9**
Exact tool-sequence match	5	0.25	10.7	0.0	**-10.7**
Open-source models					
All matched arguments schema-clean	70	6.54	7.6	**14.3**	**+6.7**
First tool correct	69	6.45	7.6	10.1	**+2.6**
No API / format error	240	22.43	7.6	7.5	-0.1
Matched gold prefix $\geq$ 2 steps	13	1.21	7.6	0.0	**-7.6**
Exact tool-sequence match	3	0.28	7.6	0.0	**-7.6**
Proprietary models					
Exact tool-step count match	4	0.42	14.1	**25.0**	**+10.9**
No API / format error	488	50.67	14.1	14.3	+0.2
All matched arguments schema-clean	75	7.79	14.1	12.0	**-2.1**
First tool correct	59	6.13	14.1	6.8	**-7.3**
Matched gold prefix $\geq$ 2 steps	23	2.39	14.1	4.3	**-9.8**
Exact tool-sequence match	2	0.21	14.1	0.0	**-14.1**

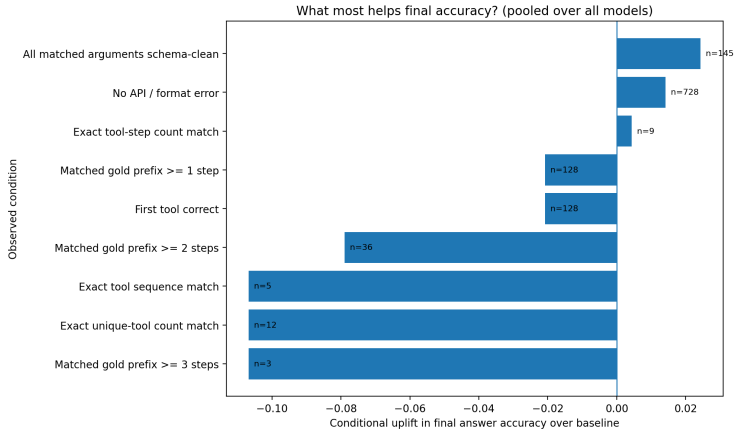


Figure 13: **Pooled conditional advantage analysis.** The strongest positive descriptive signals come from schema-clean matched arguments and the absence of API/format errors. These are descriptive associations, not causal interventions.

H.4 Trajectory length, prefix survival, and horizon effects

Table 13 shows that gold-prefix survival collapses rapidly after the first action. Even the strongest autonomous controllers only survive the first step on roughly one-third of tasks, drop to single digits or low teens by step 2, and are essentially at zero by step 3. This explains why strict trajectory success is effectively absent: the rollout rarely stays aligned long enough for full end-to-end faithfulness.

Figure 14 gives the complementary picture. The middle panel shows that answer accuracy degrades with gold trajectory length, while the right panel shows the same collapse in prefix survival. The left panel should be interpreted descriptively: tasks with longer matched prefixes are also the hardest long-horizon cases, so prefix-conditioned accuracy is not a causal oracle result.

Table 13: Gold-prefix survival under autonomous rollout (%).

Model	Step 1	Step 2	Step 3
Qwen3.5-9B	**33.64**	9.35	0.97
GPT_4o	32.71	**17.76**	**1.94**
Qwen3-8B	24.30	2.80	0.00
gpt-5.4-mini	10.28	1.87	0.00
gpt-5.4-nano	7.48	0.93	0.00
gemini-3.1-flash-lite-preview	2.80	0.00	0.00
claude-haiku-4-5	0.93	0.00	0.00
claude-opus-4-6	0.00	0.00	0.00

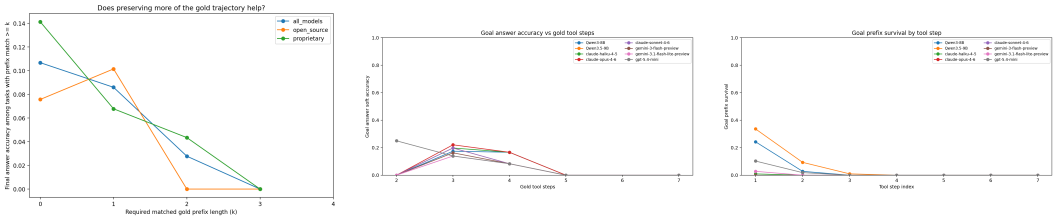


Figure 14: **Trajectory-length diagnostics. Left:** prefix-conditioned answer accuracy. **Middle:** answer accuracy as a function of gold tool-step count. **Right:** prefix survival by step index. Together these plots show that performance drops with horizon and collapses rapidly after the first action.

H.5 Tool-level bottlenecks and localized grounding

Figure 15 shows that autonomous tool matching is concentrated on relatively easy global-perception tools, especially `ImageDescription`. The weakest alignment is on `RegionAttributeDescription`, which requires localized visual grounding and is rarely reached correctly.

Table 14 complements this by aggregating structural argument pathology by tool, conditioned on *correct tool calls*. Once the correct tool is reached, schema problems are very rare for `ImageDescription` and `OCR`, but remain much more frequent for `RegionAttributeDescription`. This supports the main-text conclusion that current limitations are not merely generic formatting failures: after controller repair, the residual difficulty is accurate long-horizon local evidence use.

H.6 Cross-model qualitative failure cases

Figure 16 complements the aggregate metrics by showing how MedCTA failures unfold at trajectory level. The first case is a *shared* failure: both GPT-5.4 and Qwen3.5-9B miss the correct portal/superior-mesenteric venous thrombosis, but one fails through an immediate argument-format break while the other degrades through repeated API-error loops. The second case shows *semantic drift*: both models ignore the evidence chain needed to map chart labels to assay types and instead answer from broad medical priors. The third case highlights *long-horizon evidence failure*: GPT-5.4 produces a fluent but sign-reversed developmental comparison without using tools, whereas Qwen3.5-9B never recovers after an incorrect initial visual grounding. These examples align closely with the quantitative findings that MedCTA is primarily limited by controller instability, wrong evidence recruitment, and poor obedience to localized or multi-step evidence.

Table 14: Structural argument pathology aggregated by gold tool, conditioned on *correct-tool calls only*. Schema errors are rare for `ImageDescription` and `OCR` and concentrate in `RegionAttributeDescription`.

Tool	n	Schema OK	Any issue
GoogleSearch	1	100.0	0.0
ImageDescription	118	100.0	0.8
OCR	32	100.0	3.1
RegionAttributeDescription	10	90.0	40.0

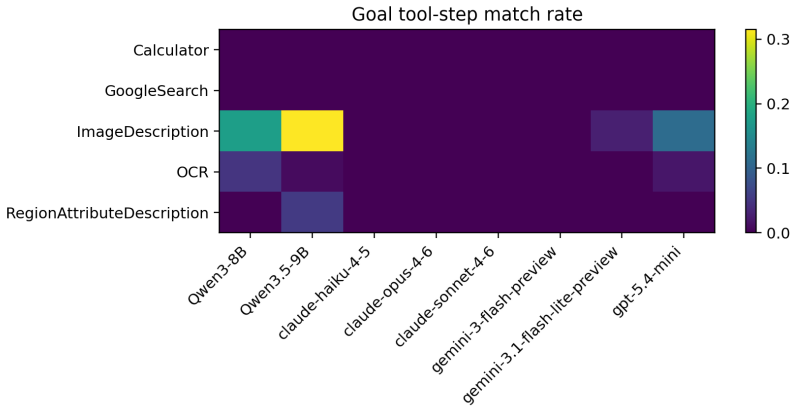


Figure 15: **Autonomous tool-step match rate by gold tool.** Successful matching is concentrated on `ImageDescription`, with much weaker alignment on `OCR` and near-zero matching on more demanding tools such as `RegionAttributeDescription`.

Cross-model qualitative examples on MedCTA

Case 1: Abdominal CT venous thrombosis — shared failure across both models

Query: *Based on the CT image, what type of venous thrombosis is present?*

Gold trajectory

- `ImageDescription`: identifies an upper-abdominal CT at the porta hepatis.
- `Region..Description`: localizes low-attenuation material in the portal venous system extending into the superior mesenteric vein.
- **Gold answer:** portal vein thrombosis with superior mesenteric vein thrombosis.

GPT-5.4

- First tool call fails with `ARGS_ERROR`: the model appends a free-text interpretation into the JSON arguments.
- That inserted text already reframes the image as a *chest* CT with pulmonary arterial filling defects.
- **Predicted answer:** pulmonary embolism.
- **Failure type:** protocol break followed by a wrong anatomic prior.

Qwen3.5-9B

- Starts with a valid `ImageDescription`, but only obtains a generic liver/vessel summary.
- Then enters repeated `NoAction`/API-error loops before issuing a coarse full-image region query.
- **Predicted answer:** deep vein thrombosis (DVT).
- **Failure type:** unstable controller followed by a generic thrombosis shortcut.

Takeaway. Both models fail on the same case, but in different ways: GPT-5.4 breaks the tool protocol immediately, whereas Qwen3.5-9B stays in the loop longer but never stabilizes enough to gather the decisive local evidence.

Case 2: Sentinel-node metastasis detection methods — semantic drift from figure evidence

Query: *Using the provided figure, identify the methods used to detect metastases in sentinel nodes.*

Gold trajectory

- `ImageDescription`: recognizes a bar chart comparing H&E, CK-18, CEA, hTRT, and MUC-1.
- `OCR`: extracts the assay labels.
- `GoogleSearch` + `Region..Description`: maps the labels to histologic vs. molecular detection.
- **Gold answer:** conventional histology, H&E staining, and RT-PCR/Southern blot assay.

GPT-5.4

- Skips the evidence chain entirely and answers directly without calling any tool.
- Compresses the figure into a generic "IHC + RT-PCR" story.
- **Predicted answer:** immunohistochemistry and RT-PCR.
- **Failure type:** premature answer with partial but incomplete method mapping.

Qwen3.5-9B

- Begins with a vague chart description and then accumulates repeated API errors.
- Later OCR recovers the labels (`H&E`, `CK-18`, `CEA`, `hTRT`, `MUC-1`), but the final answer ignores them.
- **Predicted answer:** blue dye injection, radioisotope lymphoscintigraphy, combined method, surgical excision.
- **Failure type:** semantic drift from chart evidence to unrelated clinical prior knowledge.

Takeaway. This case shows that failure is not only about syntax. Even when relevant labels are available, the models may abandon the figure-specific evidence path and answer from broad medical priors instead.

Case 3: Gpc3 embryonic kidney comparison — evidence-free inversion vs. rollout collapse

Query: *Compare Gpc3+ and Gpc3− embryonic kidneys across E12.0, E13.5, and E16.5, focusing on ureteric bud branching, kidney size, and cortical–medullary architecture.*

Gold trajectory

- `GoogleSearch`: retrieves known Gpc3 developmental phenotypes.
- `ImageDescription + OCR`: identifies stage labels and genotype layout.
- `Region..Description`: confirms enhanced branching at E12.0, larger mutant kidney at E13.5, and abnormal cortical organization/cysts by E16.5.
- **Gold answer:** mutant kidneys show enhanced branching, enlargement, and later structural disorganization.

GPT-5.4

- Produces a fluent direct answer without calling any tool.
- Reverses the main biological trend: it says the mutant kidneys are *smaller*, *less branched*, and *hypoplastic*.
- **Predicted answer:** reduced branching, smaller kidneys, delayed maturation.
- **Failure type:** evidence-free but linguistically confident inversion of the correct comparison.

Qwen3.5-9B

- First visual grounding step already fails: the figure is misdescribed as a purple tissue sample or brain/tumor image.
- OCR later extracts noisy fragments of the stage labels, but the rollout never recovers and ends in repeated API-error loops.
- **Predicted answer:** no final answer produced.
- **Failure type:** incorrect global grounding followed by controller collapse.

Takeaway. The two models expose different failure modes on the same long-horizon comparison task: GPT-5.4 remains fluent but ungrounded, while Qwen3.5-9B loses the task much earlier at the perception/controller interface.

Figure 16: **Cross-model qualitative failure cases on MedCTA.** The examples are chosen to span three recurrent benchmark failure modes: **(i)** early controller/protocol failure with shortcut answering, **(ii)** semantic drift from figure evidence to broad prior knowledge, and **(iii)** long-horizon comparison failure caused by either evidence-free reasoning or rollout collapse. Together they make the aggregate diagnostic story concrete: MedCTA errors are not only wrong final answers, but failures of protocol stability, evidence acquisition, and evidence obedience.

I MEDCTA EXAMPLES

Example 1



10444014_F1.jpg

Query:
Identify the organ shown and state whether this anatomical variant is common or rare.

Involved Tools: `ImageDescription, GoogleSearch`

Steps:

- Inspect the gross specimen to identify the organ and any abnormal accessory structures.
- Use external medical knowledge to determine whether the observed variant is common or rare.

Reasoning Trace:

- **ImageDescription:** The specimen is consistent with a gallbladder and includes an additional smaller sac-like structure, suggesting an accessory or duplicated biliary structure.
- **GoogleSearch:** Accessory or duplicated gallbladder is a rare congenital biliary anomaly, typically identified incidentally during imaging or surgery.

Answer:

Gallbladder, Rare

Justification: The gross specimen shows a gallbladder with an additional accessory sac-like structure, consistent with an accessory or duplicated gallbladder. This is a rare congenital anatomical variant rather than a common finding.

Figure 17: **Agentic reasoning example for anatomical variant identification**

Example 2



10444016_F2.jpg

Query:
Determine the imaging plane or view of the image.

Involved Tools:
`OCR, ImageDescription, GoogleSearch, RegionAttributeDescription`

Steps:

- Use OCR to extract orientation markers and scan annotations.
- Use global image description to infer the anatomical layout.
- Use external radiology knowledge to distinguish standard MRI planes.
- Verify local attributes such as airway appearance and left–right symmetry.

Reasoning Trace:

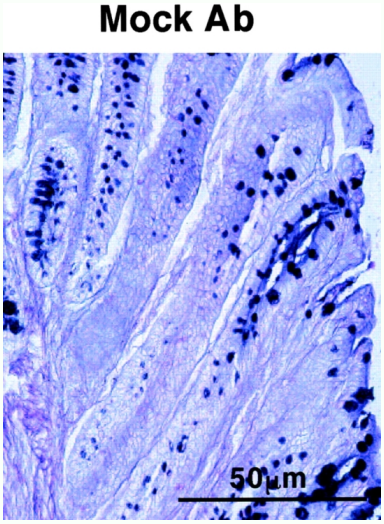
- **OCR:** The image contains markers such as `R` and `L`, along with MRI metadata.
- **ImageDescription:** The slice shows symmetric left and right soft tissues with a central airway-like structure.
- **GoogleSearch:** Standard radiology references note that bilateral symmetry and cross-sectional anatomy are consistent with common MRI plane cues.
- **RegionAttributeDescription:** The image contains a central round airway-like lumen and symmetric lateral neck soft tissues.

Answer:
Coronal MRI view

Justification: The agent combines OCR-based orientation cues, global anatomical layout, and regional attributes. The observed bilateral symmetry of the cervical soft tissues, together with the central airway-like structure and supporting radiology knowledge, leads to the prediction that the image is a **coronal MRI view**.

Figure 18: **Agentic reasoning example for imaging plane identification**

Example 3



`10477546_F8.jpg`

Query:
Specify the epithelial cell type of interest, and staining technique used.

Involved Tools: `OCR, ImageDescription, GoogleSearch`

Steps:

- Extract any textual cues (e.g., labels, scale bars) using OCR.
- Analyze tissue morphology and staining pattern from the image.
- Use domain knowledge to identify cell type and staining technique.

Reasoning Trace:

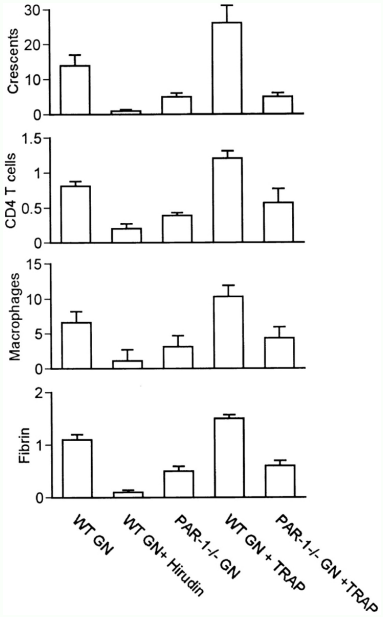
- **OCR:** Extracted scale bar (`50 µm`), confirming histological context.
- **ImageDescription:** The image shows intestinal crypt–villus architecture with columnar epithelium and numerous mucin-filled cells.
- **GoogleSearch:** Goblet cells are mucin-secreting epithelial cells; PAS–Alcian blue staining highlights mucin in blue.

Answer:
Goblet cells, PAS–Alcian blue stain

Justification: The intestinal epithelium shows characteristic mucin-filled goblet cells. The strong blue staining of mucin, along with dark nuclear counterstaining and absence of eosinophilic cytoplasm, is consistent with PAS–Alcian blue staining commonly used to highlight mucin-producing cells.

Figure 19: **Agentic reasoning example for epithelial cell and stain identification**

Example 4



10662791_F2.jpg

Query:
Extract the numerical values of the bars in the top pane and compute the difference between the highest and lowest values.

Involved Tools: `OCR, ImageDescription, Calculator`

Steps:

- Identify the structure of the figure and locate the top panel.
- Extract axis values and scale using OCR.
- Estimate bar heights relative to the axis.
- Compute the difference between maximum and minimum values.

Reasoning Trace:

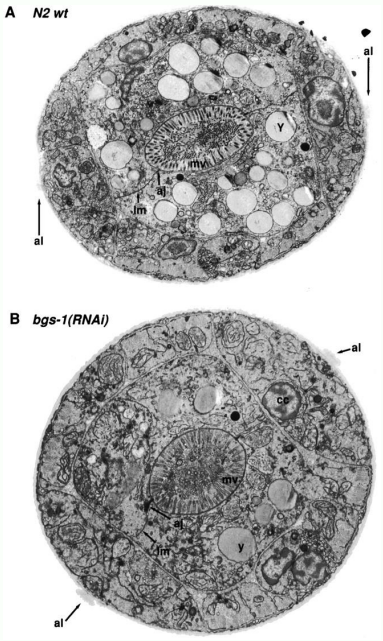
- **ImageDescription:** The figure contains multiple panels with bar charts; the top panel ranges approximately from 0 to 30.
- **OCR:** Extracted axis ticks `0, 10, 20, 30`, enabling estimation of bar values.
- **Estimation:** The highest bar is approximately 26, and the lowest is approximately 1.
- **Calculator:** Computes the difference as $26 - 1 = 25$.

Answer:
25

Justification: Using the extracted axis scale and visual estimation of bar heights, the highest value is approximately 26 and the lowest is approximately 1. Their difference is therefore 25.

Figure 20: **Agentic reasoning example for numerical extraction and computation**

Example 5



10811831_F8.jpg

Query:
What is the cell type in the image?

Involved Tools: `ImageDescription, OCR, RegionAttributeDescription`

Steps:

- Analyze the global ultrastructure using image description.
- Extract and verify structural labels using OCR.
- Examine regional attributes to identify specific cell types.

Reasoning Trace:

- **ImageDescription:** Electron micrograph shows intestinal lumen lined by epithelial cells, with labeled features including microvilli (mv), adherens junctions (aj), and seam cells.
- **OCR:** Extracted labels `mv, aj, lm, y, cc`, confirming annotated ultrastructural features.
- **RegionAttributeDescription:** Peripheral cells correspond to seam cells, distinct from intestinal epithelial cells and coelomocytes.

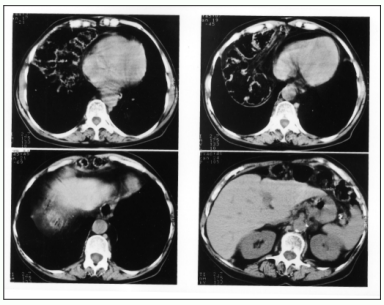
Answer:

Seam cells

Justification: The peripheral cells identified in the electron micrograph correspond to seam cells, distinct from the central intestinal epithelial cells lining the lumen and the coelomocyte in the pseudocoelomic space.

Figure 21: **Agentic reasoning example for ultrastructural cell type identification**

Example 6



`10917128_F1.jpg`

Query:
You are given axial CT slices of the lower thorax/upper abdomen. What does the CT scan show?

Involved Tools: `OCR, ImageDescription, RegionAttributeDescription`

Steps:

- Extract any textual or positional cues from the image.
- Identify anatomical structures and abnormalities across slices.
- Verify regional findings to confirm suspected pathology.

Reasoning Trace:

- **ImageDescription:** Axial CT slices show thoracic and upper abdominal anatomy with abnormal gas-filled structures in the right anterior mediastinum.
- **OCR:** No clinically informative labels detected.
- **RegionAttributeDescription:** Presence of gas-containing bowel loops within the mediastinum confirms abnormal herniation.

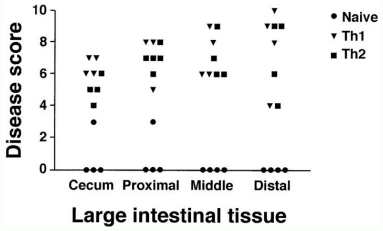
Answer:

Bowel herniation

Justification: The CT images show gas-filled bowel loops abnormally located within the right anterior mediastinum, consistent with bowel herniation into the thoracic cavity.

Figure 22: **Agentic reasoning example for thoracic CT abnormality detection**

Example 7



`11781367_fig5.jpg`

Query:
How many cells are discussed in the figure?

Involved Tools: `ImageDescription, OCR, RegionAttributeDescription`

Steps:

- Identify the overall figure type and legend entries.
- Extract exact labels and group names using OCR.
- Verify which cell groups are being compared in the plot.

Reasoning Trace:

- **ImageDescription:** The figure is a strip/scatter plot with legend entries `Naive`, `Th1`, and `Th2`.
- **OCR:** Confirms the large intestinal regions and the three groups shown in the legend.
- **RegionAttributeDescription:** The biologically relevant compared cell groups are **Th1** and **Th2**, while `Naive` is the control condition.

Answer:

Two cells: Th1 and Th2

Justification: The figure discusses disease patterns associated with two cell types, **Th1** and **Th2**. The `Naive` group serves as a control rather than a cell type of interest.

Figure 23: **Agentic reasoning example for identifying cell groups in a disease plot**

J LLM-AS-JUDGE PROMPTS

LLM-as-Judge Prompts

(a) Clinical Faithfulness (F_{acc})

```
You are a medical trajectory evaluator.

Evaluate only Clinical Faithfulness (F_acc).

Definition:
Clinical Faithfulness measures whether the predicted reasoning
process follows a clinically valid step-by-step workflow.

Focus ONLY on:
- sequence of reasoning
- whether later conclusions are supported by earlier steps
- whether the workflow is clinically sensible
- whether there are logical jumps or contradictions in the
reasoning path

Ignore:
- minor factual wording issues
- missing details unless they break the reasoning chain
- whether all findings are fully covered
- the final answer itself

Important:
- This metric is about LOGICAL WORKFLOW, not completeness.
- A trajectory can be factually incomplete but still faithful.
- A trajectory can have a good-looking final answer but low
faithfulness
if the reasoning path is poor.
- Evaluate only from the trajectory content provided.

Scoring guide:
- 1.0 = clinically coherent stepwise reasoning, no major
logic flaws
- 0.7 = mostly reasonable workflow with some weak jumps
- 0.4 = partially logical but important workflow issues
- 0.1 = mostly illogical or unsupported reasoning
- 0.0 = contradictory or clinically nonsensical reasoning

Return JSON only:
{
  "score": number
}
```

(b) Context Integration Score (C_s)

```
You are a medical trajectory evaluator.

Evaluate only Context Integration Score (C_s).

Definition:
Context Integration Score measures how well the predicted
trajectory
uses the available multimodal evidence.

Focus ONLY on:
- whether tool outputs are actually used
- whether image findings, OCR outputs, region descriptions, and
evidence are integrated into the reasoning
- whether the trajectory is grounded in available context
rather than generic guessing

Ignore:
- whether the reasoning order is ideal
- whether every medical fact is correct
- whether the final answer is complete
- the final answer itself

Important:
- This metric is about EVIDENCE USAGE, not logic or
correctness alone.
- A trajectory can be logically organized but still have low
context
integration if it ignores tool evidence.
- A trajectory can partially use evidence and should get
partial credit.
- Evaluate only from the trajectory content provided.

Scoring guide:
- 1.0 = directly and effectively integrates relevant context
- 0.7 = uses some important context but not all
- 0.4 = weak or superficial use of context
- 0.1 = almost no meaningful evidence use
- 0.0 = ignores available context entirely or is unrelated

Return JSON only:
{
  "score": number
}
```

(c) Semantic Completeness (S_{comp})

```
You are a medical trajectory evaluator.

Evaluate only Semantic Completeness (S_comp).

Definition:
Semantic Completeness measures whether the trajectory covers all
clinically necessary findings required by the task.

Focus ONLY on:
- whether required reasoning components and findings are
present in the trajectory
- whether important findings are omitted
- whether the trajectory fully covers the task requirements

Ignore:
- reasoning order
- elegance of evidence usage
- small factual imprecision unless it removes a required component
- the final answer itself

Important:
- This metric is about COVERAGE and MISSING INFORMATION.
- A concise trajectory can score 1.0 if it includes all
required findings.
- A logically good trajectory can still score low if key
clinical findings
are missing.
- Evaluate only from the trajectory content provided.

Scoring guide:
- 1.0 = all required clinical content is covered
- 0.7 = most important content covered, some omissions
- 0.4 = only partial coverage
- 0.1 = very incomplete
- 0.0 = missing essentially all required content or unrelated

Return JSON only:
{
  "score": number
}
```