Viewpoint

Beyond Model Choice: A Harness-Centered Framework for Designing Clinical Large Language Model Agents

Jialin Liu[1,2], MD; Siru Liu[3,4], PhD

[1]Department of Medical Informatics, West China Hospital of Sichuan University, Chengdu, Sichuan, China

[2]Department of Otolaryngology-Head and Neck Surgery, West China Hospital of Sichuan University, Chengdu, Sichuan, China

[3]Department of Biomedical Informatics, Vanderbilt University Medical Center, Nashville, TN, United States

[4]Department of Computer Science, Vanderbilt University, Nashville, TN, United States

Corresponding Author:
Siru Liu, PhD
Department of Biomedical Informatics
Vanderbilt University Medical Center
2525 West End Ave #1475
Nashville, TN, 37203
United States
Phone: 1 6158755216
Email: siru.liu@vumc.org

Abstract

Clinical large language model agents depend on both model capability and the surrounding system that selects information, constrains actions, and verifies execution. In this Viewpoint, we argue that this harness should be an explicit unit of clinical agent design, evaluation, and reporting. Our aim is to help clinical informatics and development teams specify this layer and to help evaluators and institutional governance teams assess its contributions and risks. Selected benchmarks show substantial performance differences across harness configurations, but architecture comparisons, bundled improvements, and within-model component comparisons support different levels of causal attribution. We propose a revisable 3-tier framework: tier 1 specifies general orchestration functions, including prompts, tools, output schemas, control flow, memory, and generic verification; tier 2 specifies their clinical adaptations for knowledge integration, differential reasoning, missingness and temporality, safety verification, and workflow alignment; and tier 3 specifies a failure-driven evaluation cycle applied to both. Our contribution is a framework that makes these connections explicit. We provide classification rules, an illustrative clinical example, a proposed evaluation protocol, and a minimum reporting checklist. Observable execution traces are distinguished from private model reasoning, with provenance and privacy safeguards supporting retrospective reconstruction. Evaluation should separate task success, clinical correctness, process safety, and operational reliability. Lightweight fine-tuning can complement harness design; their interaction requires controlled study. The framework is a proposal for empirical testing, not a validated taxonomy or evidence of deployment safety.

(J Med Internet Res 2026;28:e101269) doi: 10.2196/101269

KEYWORDS

large language model; agent; clinical decision support; harness; system design; failure analysis; benchmark

Introduction

Clinical large language model (LLM) research encompasses medical question answering, patient communication, and deployment governance [1-3]. Yet model identity alone does not specify how an agent retrieves information, uses tools, or verifies actions. Consider a hypothetical comparison of 2 health systems using the same LLM for patient-portal triage. A patient reports fatigue and leg heaviness, and the agent retrieves laboratory results and medications before drafting a response for clinician review. One configuration might unnecessarily recommend calling 911, overlook information warranting urgent review, or cite results from the wrong encounter. Another might verify encounter identity and data currency, flag missing information, and escalate unresolved uncertainty with a structured summary. This illustrative contrast highlights design choices around retrieval, prompting, tool availability, output constraints, and verification.

Benchmark findings provide different kinds of evidence. In Fast Healthcare Interoperability Resources (FHIR)-AgentBench, comprising 2931 questions, o4-mini answer correctness was 20% with a multiturn retriever and 50% with a multiturn

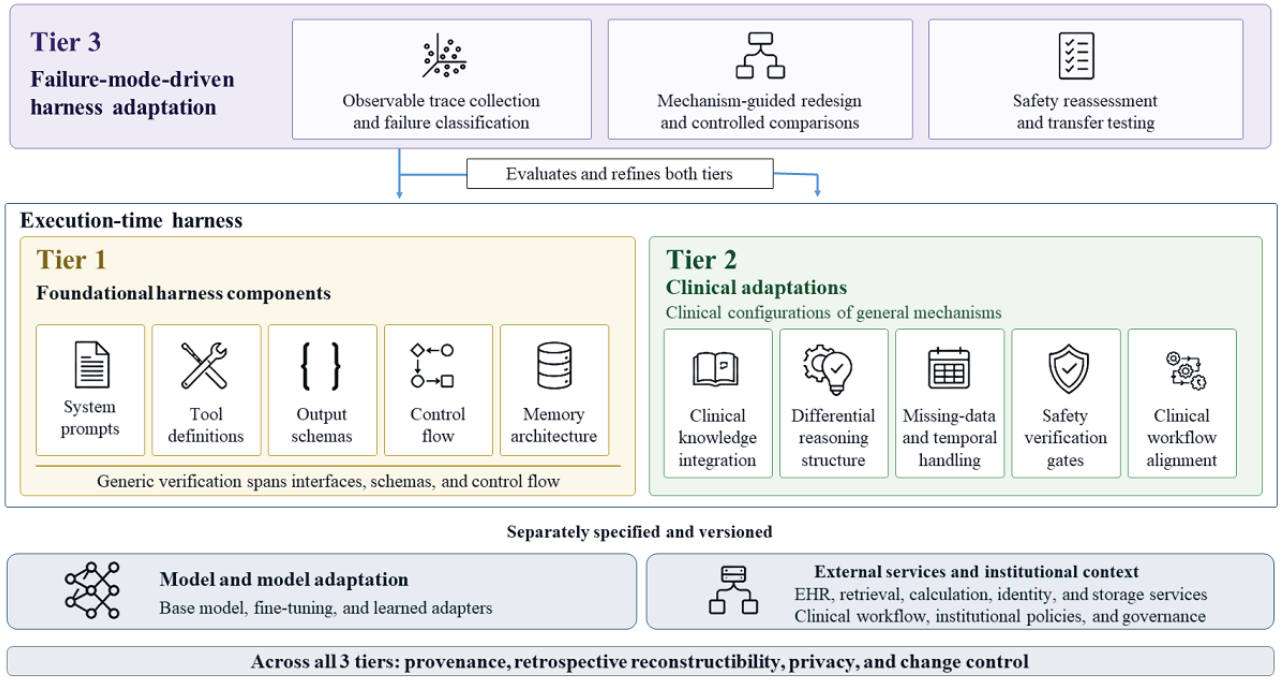
XSL•FO
RenderX

retriever plus code execution. This within-model architecture comparison supports the tested configuration change, with interpretation limited to the study's configurations and evaluation protocol. Under the same multiturn retriever plus code architecture, correctness across 4 models ranged from 44% to 50% [4]. MedAgentBench v2 reported 91% success with GPT-4.1, compared with the original benchmark's best result of 69.7% using Claude 3.5 Sonnet v2 [5,6]. This cross-model comparison changes the model, prompts, and tools together and cannot isolate a harness effect. In a more focused within-model component comparison, v2 reported 91% success without memory and 98% with feedback-derived memory under its reported training-test protocol [5]. AgentClinic reported up to 92% relative improvement in diagnostic accuracy for Llama-3 with a notebook persisting across cases, while tool benefits varied by model [7]. These selected results support controlled evaluation of harness contributions, not a general claim that harness changes outperform model upgrades.

We define the harness as the configurable execution time logic governing an LLM's interaction with its environment, including prompts, tool interfaces, output schemas, control flow, memory management, and verification [8-11]. Model training and learned parameters are distinguished from this layer, while recognizing that clinical systems may combine harness design with model adaptation. Existing work addresses medical agent architectures [12], biomedical agent engineering [13], clinical governance [14], and trace-based failure analysis [15]. A remaining challenge is to connect these perspectives through explicit component boundaries, reportable clinical configurations, and tests of specific failure mechanisms. Our contribution is a framework that makes these connections explicit, rather than a new agent architecture.

This Viewpoint aims to make clinical harness design reportable and empirically testable. We address clinical informatics and development teams, evaluators, and institutional governance teams. We advance four key messages: (1) the harness should be an explicit unit of design, evaluation, and reporting; (2) general orchestration functions should be distinguished from clinical adaptations; (3) failure-driven evaluation should assess task success, clinical correctness, process safety, and operational reliability separately; and (4) minimum reporting requirements should support reproducibility and comparison of harness designs. The following sections address these messages in turn. Figure 1 summarizes the proposed framework, which draws on selected literature and requires empirical validation.

Figure 1. Proposed 3-tier harness-centered framework for clinical large language model agents. Tier 1 identifies general harness functions, and tier 2 specifies their clinical configurations. Tier 3 provides evaluation and refinement practices applied to both. The tiers are analytic perspectives rather than successive maturity levels, and tier 3 does not represent greater clinical specificity. Model, adapter, and external service versions should be reported separately. Provenance, privacy, and change control are cross-cutting requirements. EHR: electronic health record.



Defining the Harness Boundary

For this Viewpoint, we adopt the following operational distinctions. The agent system comprises the model, harness, external services, and their integration into clinical work. Architecture describes the arrangement and interaction of these elements. Model parameters belong to the model; model adaptation refers to training-based modifications, including fine-tuning and learned adapters. The harness comprises prompts, tool exposure and invocation rules, schemas, controllers, memory read and write rules, and verification logic operating at execution time [8,10]. External infrastructure supplies data, computation, identity, and storage. Clinical workflow defines human tasks and responsibilities; organizational governance determines policies, accountability, and permitted uses [14].

XSL•FO
RenderX

Our decision rule is functional: include execution time orchestration and verification logic that determines what information or capabilities the agent receives, how execution proceeds, or whether a proposed action may proceed. Separately report the services and policies that this logic invokes or implements. Classification follows function rather than hosting location. Mixed subsystems should therefore be decomposed into their constituent functions, with interfaces and dependencies documented (Table 1). Rule engine logic that directly implements execution control or verification belongs to the harness; the clinical rules and institutional policies it implements should be documented and versioned separately.

Within the harness, general orchestration functions belong to tier 1, whereas their clinically specified adaptations belong to tier 2. For example, generic retrieval orchestration is tier 1, while clinical terminology mapping and temporal validity requirements are tier 2. A mixed implementation should report both functions. Tier 3 comprises evaluation and refinement practices applied to both tiers. A benchmark supplies an evaluation environment and scoring procedure; its controller belongs to the harness only when that controller forms part of the evaluated agent.

Table 1. Proposed classification rules for ambiguous system elements.

Element	Harness element	Separately reported element or dependency
Retrieval-augmented generation	Query construction, source selection, context assembly, and retrieval triggers	Corpus, index, embedding model, retrieval service, and their versions
Calculator	Tool schema, invocation conditions, input checks, and handling of results	Calculation implementation, supporting computation service, and their versions
Rule engine	Execution control and verification logic, including checks that gate actions	Clinical rule content, institutional policies, supporting rule services, and their versions
Authorization service	Controller logic that checks permissions and blocks or routes actions	Identity service, role assignments, and institutional authorization policy
Logging system	Instrumentation, event schema, and event-emission rules	Storage, access controls, retention service, and audit governance
Human approval	Routing, waiting, time-out, and execution blocking	Clinician review, staffing, responsibility, and approval policy
Fine-tuning or learned adapter	Selection or routing of a model or adapter at execution time	Training procedure, training data, and learned parameters

A 3-Tier Harness-Centered Framework

Overview

The tiers are analytic perspectives, not successive maturity levels (Figure 1 and Table 2). Tier 1 describes general orchestration and execution control functions. Tier 2 describes how those functions are configured through clinical knowledge, data semantics, risk requirements, and institutional clinical workflows. Tier 3 comprises evaluation and refinement practices applied to both. A component can have a tier 1 mechanism and a tier 2 configuration; for example, a generic validation hook can enforce a clinical allergy-check requirement. Provenance and reconstructibility apply across all 3 tiers.

XSL•FO
RenderX

Table 2. Proposed 3-tier harness-centered framework for clinical large language model (LLM) agents[a].

Tier	Function in the framework	Main components or practices	Medical agent failure modes addressed	Key design or evaluation question
Tier 1: foundational harness components	Establishes the general scaffolding required for any LLM agent to perceive inputs, use tools, produce structured outputs, and maintain state across turns	System prompt structure, tool definitions, output schemas, control flow, and memory architecture	Poorly defined agent role, ungrounded tool use or overly coarse-grained tool calls, unstructured outputs, chaotic multiturn behavior, and weak state persistence across turns	Has the agent been given the basic guides and sensors needed to operate reliably?
Tier 2: clinical adaptations	Adapts general agent scaffolding to clinical tasks, data structures, reasoning norms, safety constraints, and workflow conventions	Clinical knowledge integration, differential reasoning structure, missing data and temporal handling, safety verification gates, and clinical workflow alignment	Wrong terminology or code mapping; premature diagnostic closure; inference from absent, ambiguous, or stale data; unsafe or unverified clinical actions; and clinically unusable workflow outputs	Which clinical requirements should configure the general harness mechanisms?
Tier 3: failure mode–driven harness adaptation	Methodological discipline for evaluating and refining harness design, attributing effects, and assessing transferability	Trace collection and clustering, taxonomy-guided redesign, component ablation and interaction testing, safety reassessment, and cross-benchmark validation	Unattributed gains, bundled interventions without component attribution, benchmark-specific overfitting, and unsafe intermediate behavior missed by outcome grading	Does the modification reduce targeted failures with predicted trace changes, avoid unacceptable safety deterioration, and transfer beyond its development benchmark?

[a]Provenance and retrospective reconstructibility apply across all 3 tiers. Retain episode-linked evidence of model, harness, and tool versions; retrieved data or accessible versioned references; time stamps, clinical validity assessments, and missingness; memory updates; gate outcomes; authorization and action states; and relevant preaction and postaction evidence. Document capture omissions and reconstruction gaps. Reconstruction of the observed sequence does not imply exact replay.

Tier 1: Foundational Harness Components

Tier 1 comprises general mechanisms governing agent execution across domains [16,17]. We organize these into the following five components: (1) system prompts define roles, constraints, and reasoning norms; (2) tool definitions expose external capabilities through structured schemas, including FHIR queries and calculators [6] (benchmark comparisons suggest that abstracting low-level implementation details can improve performance [5]); (3) output schemas constrain responses and support downstream validation; (4) control flow sequences model interactions, tool use, and actions through single-turn execution, multistep loops, planning, or reflection [18,19]; and (5) memory architecture governs state retention and access within and across authorized tasks.

Memory requires explicit scope: within-episode memory retains current observations and pending actions; longitudinal patient memory links authorized encounters for one person; and cross-case memory spans patients or independent cases. Cross-case benchmark notebooks [7] do not establish that reusing patient-specific content across production cases is appropriate; such content requires a defined purpose and access boundary. Teams should report namespaces, read and write permissions, retention, reset conditions, and protections against retrieval from the wrong patient. Fixed-configuration evaluation should freeze the initial memory snapshot and prespecify permitted updates, including within-episode changes. Studies allowing cross-case learning during evaluation should separately report the adaptive protocol, case order, update sources, and exposure to test feedback.

Generic verification operates across tool interfaces, output schemas, and control flow through schema validation, permission checks, execution checks, and error handling. The guides-and-sensors distinction separates feedforward guidance before actions from feedback during or after execution, although components may serve both functions [11]. Tier 2 specifies clinical requirements, including patient and encounter matching, clinical prerequisites, and required clinician approval. Tier 3 evaluates the contributions, interactions, and failure modes of these mechanisms and their clinical configurations.

Tier 2: Clinical Adaptations

A harness design choice belongs to tier 2 when its specification or evaluation depends on clinical semantics, clinical consequences, or institutional clinical workflows. These functions need not be unique to medicine: membership depends on the clinical requirements governing a general mechanism, rather than clinical use alone. The 5 proposed categories are revisable, and their effects require empirical testing.

Clinical knowledge integration addresses mismatches between natural language clinical concepts, coded terminologies, and institutional data structures [5]. Candidate implementations include code-aware tool definitions, prompt-level references, and retrieval-augmented coding lookups [20-22]. Design choices should balance context length from inline references against latency from dynamic lookup. Implementations should return a selected code or an explicit unresolved status, with terminology version, source, and candidate alternatives. Evaluation should assess code and resource selection, relevant record recall, unsupported mappings, and ambiguity handling, distinguishing retrieval success from clinically valid interpretation.

XSL•FO
RenderX

Differential reasoning structure addresses premature closure in diagnostic and therapeutic reasoning. Analyses and simulation studies motivate testing structured alternatives, uncertainty labeling, and multiagent critique to counter anchoring [23-25]. Their independent contributions require controlled evaluation. Implementations can require a concise clinical output recording candidate diagnoses or interpretations, supporting or conflicting findings, uncertainty, and the next information need. This output does not require private chain-of-thought, and its fluency does not establish reasoning quality. Clinician-adjudicated evaluation should assess omitted alternatives, premature closure, unnecessary testing, and escalation appropriateness.

Missing data and temporal handling address whether information exists, when it was recorded, and whether it remains valid for the task [6]. Five states should be distinguished: not measured, supported by explicit evidence that measurement was not performed; not retrieved, when retrieval was omitted, failed, or incomplete; not documented, when no record is found within a verified documentation and search scope; not applicable, when the item is irrelevant under specified clinical or task conditions; and temporally invalid, when an available value fails the applicable temporal validity criteria. These states may overlap, and unresolved status should remain explicit. An empty query result establishes only that no matching record was returned within the queried scope. Wrappers should record query status, patient and encounter identifiers, units, measurement and retrieval timestamps, and applicable validity rules, explicitly marking unavailable fields. Prespecified rules should route unresolved missingness to clarification, further retrieval, or escalation. Evaluation should assess state classification and subsequent handling, separately reporting retrieval failures, temporally invalid data use, encounter mismatches, and unsupported inference from absent records [5,6].

Safety verification gates apply clinical criteria to actions with different consequences and constraints. Preaction checks verify patient and encounter identity, clinical prerequisites, relevant allergies or interactions, and required human approval. Intra-action checks enforce permissions and action constraints and recheck mutable state as needed. Postaction checks reconcile the intended change with execution status and authoritative electronic health record (EHR) state. A time-out does not establish failure: uncertain execution status should be recorded as unknown and authoritative state reconciled before retrying. These checks extend clinical decision support principles [26,27]. EHR-integrated evidence indicates sensitivity to prompt and input framing [28], while safety frameworks support evaluating safety separately from utility [29].

Clinical workflow alignment addresses institution-specific conventions while distinguishing configurable defaults from mandatory requirements. Morning laboratory timing may be configurable where policy permits; convenience defaults must not override required approvals or safety prohibitions. Workflow-specific wrappers should distinguish hold from discontinue and identify the intended active order to be changed [5]. Implementations may use policy-constrained defaults, disambiguating tool interfaces, and locally validated examples. Document each rule's configurable or mandatory status, its basis in local convention, clinical policy, or applicable legal or regulatory requirements, and its responsible owner, source, version, and effective date. Evaluate order selection, handoff completion, policy adherence, and clinician correction burden. Validate defaults locally rather than presenting them as universal clinical rules.

Tier 3: Failure Mode–Driven Harness Adaptation

An execution trace comprises time-ordered, observable artifacts from an agent episode: application visible model inputs and outputs, tool calls and observations, retrieval results, structured states, memory updates, action proposals, authorization and execution records, receipts, and state transitions. Private chain-of-thought is not required, and generated explanations do not establish faithful access to latent reasoning or a validated causal account. Report capture scope, omissions, and logging failures. The primary analysis unit is a task episode with a defined patient or case, starting state, goal, and stopping rule. Nested action events support localization; repeated episodes from the same patient or task template may be dependent.

Figure 2 presents the proposed workflow from baseline specification through failure localization, classification and mechanism formulation, controlled comparison, outcome and safety reassessment, and transfer testing [15]. Freeze the baseline model, harness, tools, data snapshot, and environment; separate development and test cases at the patient and task template levels where applicable. Prespecify important failures, severity categories, denominators, and acceptance criteria with clinical and operational experts. Analyze successful and failed episodes to identify the first relevant deviation and downstream consequences. Use an explicit codebook, double code a subset, report agreement, and adjudicate disagreements; automated clustering requires clinical review. Sequential redesign uses development cases. If held-out feedback informs modification, use a fresh holdout for confirmatory evaluation.

Compare baseline and modified configurations on the same held-out cases, holding other settings fixed. Repeat stochastic runs and report paired effects and uncertainty accounting for clustered episodes. Single-component ablations estimate conditional effects; suspected dependencies warrant factorial comparisons, such as timestamp availability crossed with validity gating. Report all configurations and interaction estimates with uncertainty on a prespecified outcome scale. Sensitivity analyses assess robustness to thresholds, time windows, and resource limits rather than isolate component contributions. When model adaptation also changes, cross base versus adapted models with baseline versus modified harnesses. If component comparisons are infeasible, limit attribution to the combined modification. Evidence supporting mechanism-specific improvement should include fewer targeted failures among eligible cases, the predicted trace change, and effects assessed against prespecified criteria. Report counts, denominators, unsuccessful interventions, and residual failures; aggregate success alone is insufficient.

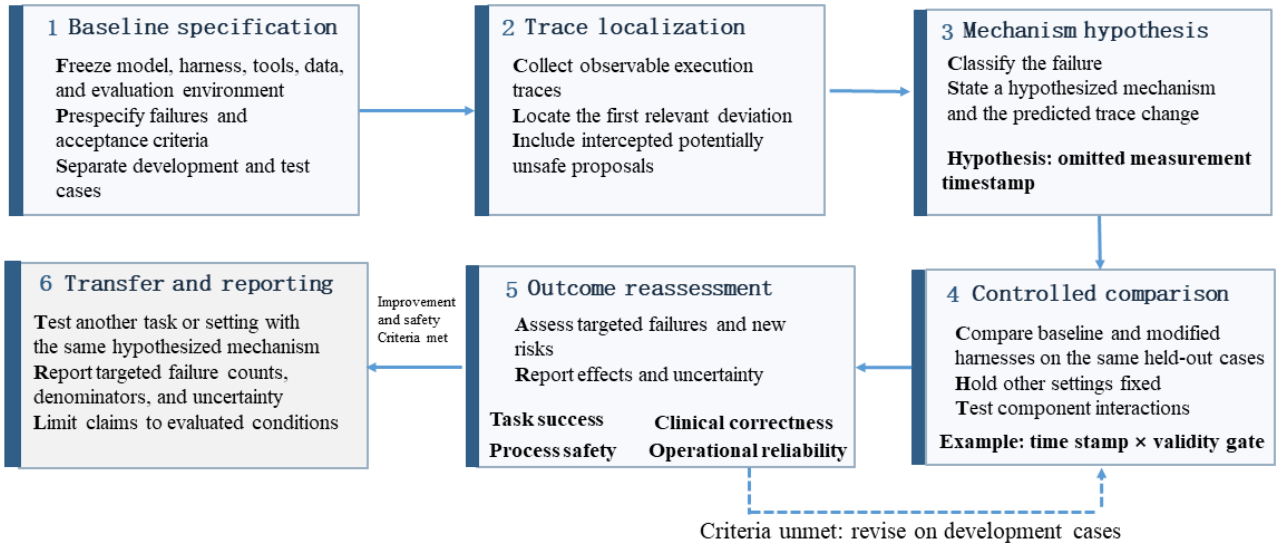
Evaluation should distinguish task success against the task specification, clinical correctness of interpretation and actions, process safety against identity and action constraints, and operational reliability through tool failures, time-outs, duplication, latency, and recovery. Report events in each

XSL•FO
RenderX

relevant domain. Correct final output does not erase wrong-patient retrieval, unauthorized actions, or other process safety failures. Assess unsafe proposals and executed actions, appropriate and inappropriate blocks, unauthorized attempts, confirmed state verification failures, and indeterminate outcomes, using counts, rates, exposure denominators, and uncertainty. Distinguish interception, execution, and observed harm, and classify near misses using prespecified criteria. Grade potential severity by plausible harm and reversibility separately from observed consequences. Report high-severity failures separately and examine relevant high-risk subgroups and edge cases. Prespecified serious events should trigger investigation and, where warranted, restriction or suspension pending review. Zero observed serious events do not establish safety. Transfer testing proceeds only when prespecified improvement and safety criteria are met.

Transfer tests should vary task structure, data environment, or institutional workflow while retaining the hypothesized failure mechanism. Candidate settings include read-only FHIR retrieval [4], conditional EHR ordering [5,6], and interactive diagnostic encounters [7]. A temporal validity intervention could transfer across laboratory-dependent medication tasks, whereas a local wrapper may require another site. Freeze each evaluated configuration, prespecify memory initialization and permitted updates, and prevent unintended cross-case contamination and test feedback leakage. Distinguish unchanged transfer from transfer requiring local reconfiguration. Compare against task-specific baselines, reporting targeted failure counts, eligible case denominators, paired effects, and uncertainty. Evidence supporting generalization should include sustained targeted failure reduction and the predicted trace change while meeting prespecified benefit and safety criteria. Without transfer testing, restrict claims to the evaluated setting.

Figure 2. Proposed failure-driven evaluation workflow for clinical large language model agents. The workflow comprises baseline specification, trace-level failure localization, failure classification and mechanism formulation, targeted modification and controlled comparison, reassessment of targeted failures and safety regressions, and transfer testing. Further redesign uses development cases, preserving held-out cases for evaluation.



Clinical Application and Reporting of the Framework

An Illustrative Clinical Task

Consider a hypothetical clinician request to review a low potassium alert and prepare any indicated replacement order under the hospital's protocol for approval. Tier 1 provides general retrieval and draft action interfaces, structured outputs, episode memory, and a controller that blocks execution pending approval. The EHR and authorization service remain external dependencies.

Tier 2 configures patient-scoped retrieval and draft orders, maps laboratory tests to local codes, and retrieves results, units, time stamps, renal function data, relevant medications, and existing replacement orders. Differential support considers current abnormality, historical results, and conflicting evidence. Missing data or uncertain timing triggers review. Safety gates enforce protocol constraints and block unsupported or duplicate orders. Before executing an approved action, checks reconfirm patient and encounter identity and current orders; postaction checks verify the intended EHR change. Local defaults cannot override mandatory requirements.

If a clinician intercepts a stale result proposal before execution, tier 3 records a temporal validity failure and an intercepted potentially unsafe proposal, even if the final recommendation is appropriate. An omitted measurement time stamp is a hypothesized mechanism. A proposed factorial comparison would cross time stamp availability with validity gating on held-out cases, holding the model and remaining configuration fixed. Evaluation would assess targeted failures, missed alerts, and unnecessary escalations; transfer testing would use another laboratory-dependent medication task. This is an illustrative

XSL•FO
RenderX

design and evaluation proposal, not treatment guidance or empirical validation.

Provenance and Privacy in Trace Collection

Provenance should support retrospective reconstruction of agent-mediated decisions [30]. For clinical agents, we propose the following recording requirements. Link each episode to the model provider, version, and decoding settings; harness commit and configuration; available and invoked tools, versions, and schemas; retrieved clinical content or accessible versioned references; measurement and retrieval times; missingness states; and memory snapshots or versioned updates. Record clinical validity criteria and whether retrieved data satisfied them at the decision time, together with safety gate identifiers, rule versions, and pass, fail, not evaluated, or unknown outcomes. Distinguish recommended, authorized, attempted, executed, failed, and execution unknown actions, retaining the authorizer, authorization time, action correlation identifiers, and relevant preaction and postaction state. Reconcile authoritative state after an uncertain write before retrying [31]. Hashes can help verify retained content's integrity but cannot recover missing content; references support reconstruction only while the specified versions remain accessible. Document reconstruction gaps. Reconstructing the observed sequence and available evidence does not guarantee exact replay, particularly with stochastic models or changing external services.

We propose layered retention: compact event records for consequential episodes, with richer, time-limited diagnostic capture for approved investigations or sampled evaluation. Store necessary patient data or versioned references within controlled institutional systems, planning for episode volume, payload size, retention duration, and source version availability. Apply least privilege access, encryption, access auditing, purpose restrictions, and retention and deletion schedules; assess vendor access and data residency before transmitting traces. The United States Health Insurance Portability and Accountability Act security rule addresses access, audit, integrity, and transmission safeguards [32], while the European Union's General Data Protection Regulation emphasizes purpose limitation, data minimization, and storage limitation [33]. Applicable duties and retention periods require local determination; this design is not a compliance guarantee.

Deidentification and pseudonymization may leave reidentification risks in clinical narratives. Prefer local analysis of protected traces and share reviewed taxonomies, aggregate counts, or synthetic examples when raw sharing is unjustified. Sampling and incomplete capture may miss rare unsafe trajectories; report capture coverage, dropped events, and logging failures alongside performance. Where required provenance is unavailable, document the gap and limit the affected action scope; absent logs do not establish safe execution.

Harness Engineering and Model Adaptation

Harness durability is a hypothesis about continuing needs for governed behavior, not a settled prediction about implementation. Medical training environments illustrate that model adaptation can improve task capability [34]. Parameter-efficient methods such as low-rank adaptation train additional parameters while keeping base-model weights fixed [35] and therefore belong to model adaptation under our boundary rule. Instruction tuning likewise constitutes model adaptation, whereas retrieval orchestration and execution time control belong to the harness; underlying services and institutional policies should be reported separately. Reduced training cost alone does not establish clinical safety or resolve data governance requirements.

Hybrid systems should separately version the base model, adapter, training data specification, retrieval configuration, and controller. For example, a local adapter may learn order terminology while the harness applies current policy and obtains approval. If training absorbs a prompt rule, document the function's relocation and reassess the system without double counting the change as an independent harness effect. A factorial comparison crossing base versus adapted models with baseline versus modified harnesses can assess complementarity and safety regressions. Externally hosted APIs are one deployment option, not a universal constraint on health systems. Future domain-specific models, tool learning, policy engines, and dynamically generated workflows may shift these boundaries. Regardless of how functions are distributed across learned models and execution time components, clinical systems require accountable, auditable governance.

Implementation and Reporting Considerations

Teams should specify tier 1 interfaces, identify tier 2 clinical requirements, and establish a tier 3 evaluation and incident response plan. Report harness components as explicitly as model identity, bound component attribution and generalization by the available comparisons, and provide versioned artifacts where disclosure is permitted. Table 3 proposes a minimum reporting set, not a validated reporting standard. These proposals complement clinical AI governance and life cycle guidance [14,36].

XSL•FO
RenderX

Table 3. Proposed minimum reporting set for clinical agent studies[a].

Item	Minimum information
Model and adaptation	Provider, model identifier and version, access date, and inference settings; adapter identifier and version, training data specification, and training procedure where applicable
Prompts and harness version	System prompt or reproducible specification, templates, configuration, harness commit or version, and dependencies
Tools and retrieval	Tool inventory, schemas, versions, permissions, retrieval sources and index versions, query construction, and context selection rules
Output and control flow	Output schema; branching, stopping rules, retries, time-outs, action reconciliation, and handoffs
Memory	Episode, patient, and cross-case scope, initialization, permitted updates, retention, isolation, reset conditions, and evaluation feedback exposure
Verification and approval	Pre-, intra-, and postaction checks; clinical thresholds, human approval points, and fail-safe behavior
Trace and reconstruction	Event schema, capture coverage, and omissions; episode-linked system and data versions, temporal validity assessments, missingness, memory updates, gate outcomes, authorization and action states, and preaction and postaction evidence; access to retained content or versioned references for authorized reconstruction; and retention, access and privacy controls, and reconstruction gaps
Environment and cases	Benchmark and task version, patient population, sites, data snapshot, initial state, development and test splits, case counts, and leakage controls
Evaluation and attribution	Task success, clinical correctness, process safety, and operational reliability; severity rubric, failure counts and denominators, adjudication, ablations and interaction analyses, sensitivity analyses, uncertainty, and residual failures and harms
Transfer and change control	External tasks and sites, local reconfiguration, and limits on generalization; monitoring owner, change-approval procedure, roll back, and revalidation triggers

[a]Report each item or indicate that it is not applicable, with justification. Where privacy, security, or commercial restrictions prevent disclosure, describe the omitted information and resulting limits on reproducibility.

Limitations and Future Research Directions

This is a proposed, revisable taxonomy and evaluation approach, not a validated classification system. Its clinical categories draw on selected benchmarks and may omit requirements of multimodal, pediatric, patient-facing, or resource-constrained settings. Synthetic and retrospective task performance does not establish prospective patient benefit. Functional boundary rules require mixed subsystems, such as retrieval-augmented generation pipelines, to be decomposed into general mechanisms, clinical configurations, and external dependencies, but cannot eliminate classification disagreements. Restricted trace capture may limit retrospective reconstruction, while constraints on cross-institutional data sharing may restrict external validation and comparison. Changing external services and model stochasticity may prevent exact replay. The worked example and experimental designs are illustrative; no original experiments were conducted.

Future research should address the following five directions: (1) assess the coverage and classification consistency of the tier 2 taxonomy and use controlled ablations and interaction-aware comparisons to estimate component contributions and interactions; (2) develop clinically grounded medical agent failure mode taxonomies that may differ from those in general domains; (3) test whether harness interventions generalize beyond their development tasks and settings through cross-benchmark and cross-task comparisons; (4) develop trace-level methods that complement outcome-level success rates by assessing process safety, operational reliability and efficiency, and the clinical appropriateness of observable outputs; and (5) assess the completeness and feasibility of the proposed minimum reporting set in Table 3 and whether its use improves reproducibility and cross-study comparison.

Conclusions

Clinical agent studies should treat the harness as an explicit unit of design, evaluation, and reporting. Selected benchmarks show material performance differences across harness configurations, but comparison with model substitution requires controlled, task-specific evaluation. Our proposed framework distinguishes general functions from clinical adaptations and links both to failure-specific evaluation, supporting controlled assessment of model and harness contributions and their interactions. Its practical value rests on testable mechanisms, separate outcome and safety assessments, versioned provenance, and clear limits on generalization. Harness design and model adaptation should be assessed together as evolving parts of a governed clinical system.

Funding

No external financial support or grants were received from any public, commercial, or not-for-profit entities for the research, authorship, or publication of this paper.

XSL•FO
RenderX

Data Availability

Data sharing is not applicable to this paper, as no new datasets were generated or analyzed during this study.

Generative AI Use

Following the initial drafting of the manuscript, the authors used ChatGPT (GPT-6.0; OpenAI) for AI-assisted language editing to enhance clarity and style. All suggested edits were subject to critical review, and the authors subsequently made additional substantive revisions. The authors assume complete responsibility for the integrity and accuracy of the final content.

Authors' Contributions

JL and SL conceptualized the manuscript and designed the framework. JL led the literature review and evidence synthesis, with literature analysis and conceptual refinement by SL. JL and SL contributed to drafting the manuscript. Both authors critically revised the manuscript, approved the final version, and agreed to be accountable for all aspects of the work.

Conflicts of Interest

None declared.

References

1. Singhal K, Tu T, Gottweis J, Sayres R, Wulczyn E, Amin M, et al. Toward expert-level medical question answering with large language models. Nat Med. Mar 2025;31(3):943-950. [doi: 10.1038/s41591-024-03423-7] [Medline: 39779926]
2. Marketing submission recommendations for a predetermined change control plan for artificial intelligence-enabled device software functions: guidance for industry and Food and Drug Administration staff. U.S. Food & Drug Administration. 2025. URL: https://www.fda.gov/regulatory-information/search-fda-guidance-documents/marketing-submission-recommendations-predetermined-change-control-plan-artificial-intelligence [accessed 2026-10-08]
3. Garcia P, Ma SP, Shah S, Smith M, Jeong Y, Devon-Sand A, et al. Artificial intelligence-generated draft replies to patient inbox messages. JAMA Netw Open. Mar 04, 2024;7(3):e243201. [FREE Full text] [doi: 10.1001/jamanetworkopen.2024.3201] [Medline: 38506805]
4. Lee G, Bach E, Yang E, Pollard T, Johnson A, Choi E, et al. FHIR-AgentBench: benchmarking LLM agents for realistic interoperable EHR question answering. Proceedings of Machine Learning Research. 2026. URL: https://openreview.net/pdf?id=Le1hGVQNb8 [accessed 2026-11-04]
5. Chen E, Postelnik S, Black K, Jiang Y, Chen JH. MedAgentBench v2: improving medical LLM agent design. Pac Symp Biocomput. 2026;31:354-371. [FREE Full text] [doi: 10.1142/9789819824755_0025] [Medline: 41758153]
6. Jiang Y, Black KC, Geng G, Park D, Zou J, Ng AY, et al. MedAgentBench: a virtual EHR environment to benchmark medical LLM agents. NEJM AI. Sep 2025;2(9):AIdbp2500144. [doi: 10.1056/AIdbp2500144] [Medline: 42932685]
7. Schmidgall S, Ziaei R, Harris C, Kim JW, Reis EP, Jopling J, et al. AgentClinic: a multimodal benchmark for tool-using clinical AI agents. NPJ Digit Med. Apr 27, 2026;9(1):499. [FREE Full text] [doi: 10.1038/s41746-026-02674-7] [Medline: 42045532]
8. Meng Q, Wang Y, Chen L, Li Y, Wu W, Jiang W, et al. Agent harness for large language model agents: a survey. Preprints.org. Preprint posted online on April 28, 2026. [FREE Full text] [doi: 10.20944/PREPRINTS202604.0428.V3]
9. My AI adoption journey. Mitchell Hashimoto. Feb 05, 2026. URL: https://mitchellh.com/writing/my-ai-adoption-journey [accessed 2026-04-20]
10. Effective harnesses for long-running agents. Anthropic. Nov 26, 2025. URL: https://www.anthropic.com/engineering/effective-harnesses-for-long-running-agents [accessed 2026-04-20]
11. Böckeler B. Harness engineering for coding agent users. Martin Fowler. Apr 02, 2026. URL: https://martinfowler.com/articles/harness-engineering.html [accessed 2026-04-21]
12. Wang W, Ma Z, Wang Z, Wu C, Ji J, Chen W, et al. A survey of LLM-based agents in medicine: how far are we from Baymax? In: Findings of the Association for Computational Linguistics: ACL 2025. Stroudsburg, PA. Association for Computational Linguistics; 2025.
13. Moore JH, Tatonetti NP. From prompt engineering to agent engineering: expanding the AI toolbox with autonomous agentic AI collaborators for biomedical discovery. BioData Min. Nov 13, 2025;18(1):78. [FREE Full text] [doi: 10.1186/s13040-025-00502-4] [Medline: 41233899]
14. Cao W, Zhang Q, Liu J, Liu S. From agents to governance: essential AI skills for clinicians in the large language model era. J Med Internet Res. Jan 14, 2026;28:e86550. [FREE Full text] [doi: 10.2196/86550] [Medline: 41534067]
15. Song K, Jayarajan A, Ding Y, Su Q, Zhu Z, Liu S, et al. Aegis: taxonomy and optimizations for overcoming agent-environment failures in LLM agents. arXiv. Preprint posted online on August 27, 2025. [doi: 10.48550/arXiv.2508.19504]

XSL•FO
RenderX

16. Yang J, Jimenez C, Wettig A, et al. SWE-agent: agent-computer interfaces enable automated software engineering. In: Advances in Neural Information Processing Systems 37. San Diego, CA. Neural Information Processing Systems Foundation; 2024:50528-50652.
17. Yao S, Shinn N, Razavi P, Narasimhan K. τ-bench: a benchmark for tool-agent-user interaction in real-world domains. arXiv. Preprint posted online on June 17, 2024. [doi: 10.48550/arXiv.2406.12045]
18. Yao S, Zhao J, Yu D, Du N, Shafran I, Narasimhan K, et al. ReAct: synergizing reasoning and acting in language models. In: 11th International Conference on Learning Representations, ICLR 2023. Amherst, MA. OpenReview.net; 2023.
19. Shinn N, Cassano F, Gopinath A, Narasimhan K, Yao S. Reflexion: language agents with verbal reinforcement learning. In: Oh A, Naumann T, Globerson A, Saenko K, Hardt M, Levine S, editors. NIPS '23: Proceedings of the 37th International Conference on Neural Information Processing Systems. New York, NY. Curran Associates Inc; 2023:8634-8652.
20. Liu S, McCoy AB, Wright A. Improving large language model applications in biomedicine with retrieval-augmented generation: a systematic review, meta-analysis, and clinical development guidelines. J Am Med Inform Assoc. Apr 01, 2025;32(4):605-615. [FREE Full text] [doi: 10.1093/jamia/ocaf008] [Medline: 39812777]
21. Xiong G, Jin Q, Wang X, Zhang M, Lu Z, Zhang A. Improving retrieval-augmented generation in medicine with iterative follow-up questions. Pac Symp Biocomput. 2025;30:199-214. [FREE Full text] [doi: 10.1142/9789819807024_0015] [Medline: 39670371]
22. Wada A, Tanaka Y, Nishizawa M, Yamamoto A, Akashi T, Hagiwara A, et al. Retrieval-augmented generation elevates local LLM quality in radiology contrast media consultation. NPJ Digit Med. Jul 02, 2025;8(1):395. [FREE Full text] [doi: 10.1038/s41746-025-01802-z] [Medline: 40604147]
23. Moëll B, Sand Aronsson F, Akbar S. Medical reasoning in LLMs: an in-depth analysis of DeepSeek R1. Front Artif Intell. Jun 18, 2025;8:1616145. [FREE Full text] [doi: 10.3389/frai.2025.1616145] [Medline: 40607450]
24. Ke Y, Yang R, Lie SA, Lim TX, Ning Y, Li I, et al. Mitigating cognitive biases in clinical decision-making through multi-agent conversations using large language models: simulation study. J Med Internet Res. Nov 19, 2024;26:e59439. [FREE Full text] [doi: 10.2196/59439] [Medline: 39561363]
25. Kim Y, Park C, Jeong H, Chan YS, Xu X, McDuff D, et al. MDAgents: an adaptive collaboration of LLMs for medical decision-making. NIPS '24: Proceedings of the 38th International Conference on Neural Information Processing Systems. 2024. URL: https://dl.acm.org/doi/10.5555/3737916.3740438 [accessed 2026-11-04]
26. Liu S, McCoy AB, Sittig DF, Peterson JF, Lasko TA, Wright A. A standard-based taxonomy of features that affect user response to clinical decision support alerts. BMC Med Inform Decis Mak. Oct 21, 2025;25(1):389. [FREE Full text] [doi: 10.1186/s12911-025-03231-0] [Medline: 41121206]
27. Wright A, Phansalkar S, Bloomrosen M, Jenders RA, Bobb AM, Halamka JD, et al. Best practices in clinical decision support: the case of preventive care reminders. Appl Clin Inform. 2010;1(3):331-345. [FREE Full text] [doi: 10.4338/ACI-2010-05-RA-0031] [Medline: 21991299]
28. Rodriguez-Nava G, Keyes T, Ambers N, Miranti E, Viana-Cardenas EP, Tariq W, et al. Using secure artificial intelligence agents integrated within the electronic medical record for the evaluation of blood culture appropriateness-Northern California, 2025. Infect Control Hosp Epidemiol. Feb 2026;47(2):206-209. [doi: 10.1017/ice.2025.10349] [Medline: 41216923]
29. Kanithi P, Christophe C, Pimentel MA, Raha T, Munjal P, Saadi N, et al. MEDIC: comprehensive evaluation of leading indicators for LLM safety and utility in clinical applications. Trans Mach Learn Res. 2026:1-47. [FREE Full text]
30. Souza R, Gueroudji A, DeWitt S, et al. PROV-AGENT: unified provenance for tracking AI agent interactions in agentic workflows. In: 2025 IEEE International Conference on eScience (eScience). New York, NY. IEEE; 2025.
31. Oswald J. Reliable execution of consequential AI agent actions under retries and ambiguous outcomes. Zenodo. Sep 26, 2026. URL: https://zenodo.org/records/22969881 [accessed 2026-11-02]
32. Summary of the HIPAA Security Rule. U.S. Department of Health and Human Services. URL: https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html [accessed 2026-10-08]
33. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). European Union. 2016. URL: https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng [accessed 2026-10-08]
34. Xu R, Zhuang Y, Zhong Y, Yu Y, Wang Z, Tang X, et al. MedAgentGym: a scalable agentic training environment for code-centric reasoning in biomedical data science. arXiv. Preprint posted online on June 4, 2025. [doi: 10.48550/arXiv.2506.04405]
35. Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, et al. LoRA: low-rank adaptation of large language models [Poster]. In: Proceedings of the Tenth International Conference on Learning Representations. 2022. Presented at: ICLR 2022; April 25-29, 2022; Virtual Event. URL: https://openreview.net/forum?id=nZeVKeeFYf9
36. Lekadir K, Frangi AF, Porras AR, Glocker B, Cintas C, Langlotz CP, et al. FUTURE-AI: international consensus guideline for trustworthy and deployable artificial intelligence in healthcare. BMJ. Feb 05, 2025;388:e081554. [FREE Full text] [doi: 10.1136/bmj-2024-081554] [Medline: 39909534]

XSL•FO
RenderX

Abbreviations

EHR: electronic health record
FHIR: Fast Healthcare Interoperability Resources
LLM: large language model

Edited by I Steenstra; submitted 13.May.2026; peer-reviewed by H Su, K Luz, K Ray Amat; comments to author 07.Oct.2026; revised version received 27.Oct.2026; accepted 28.Oct.2026; published 09.Nov.2026

Please cite as:
Liu J, Liu S
Beyond Model Choice: A Harness-Centered Framework for Designing Clinical Large Language Model Agents
J Med Internet Res 2026;28:e101269
URL: https://www.jmir.org/2026/1/e101269
doi: 10.2196/101269
PMID:

©Jialin Liu, Siru Liu. Originally published in the Journal of Medical Internet Research (https://www.jmir.org), 09.Nov.2026. This is an open-access article distributed under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work, first published in the Journal of Medical Internet Research (ISSN 1438-8871), is properly cited. The complete bibliographic information, a link to the original publication on https://www.jmir.org/, as well as this copyright and license information must be included.

XSL•FO
RenderX