MEDitor: A Plug-and-Play Medical Multi-Agent Routing Protocol

Yunwei Wang[1,2], Xianrui Zhang[1], Qiankun Ma[1], Rongjie Wang[1], Zhen Song[1], Haofei Wang[1,2,*]

[1]Pengcheng Laboratory, Shenzhen, China
[2]Southern University of Science and Technology, Shenzhen, China
[*]Corresponding author
[1]{wangjw04, zhangxr01, maqk, wangrj, songzh01, wanghf}@pcl.ac.cn

Abstract

Reliable medical reasoning with large language models (LLMs) is hindered by a practical trade-off between accuracy and deployment cost: larger models often perform strongly but are expensive to host, while compact open-source medical LLMs can fail on difficult cases. We propose MEDitor, a training-free, plug-and-play multi-agent routing protocol that improves accuracy and traceability by selectively escalating disagreement cases. MEDitor first queries two lightweight medical experts and terminates early upon agreement. For disagreements, it escalates to an evidence-grounded coordinator that constructs option-aligned evidence, followed by a candidate-constrained judge that selects between the competing candidates or abstains. A deterministic fallback guarantees reproducibility. Across five English-language medical multiple-choice QA (MCQA) benchmarks, including the expert-level MedXpertQA-Text benchmark, MEDitor achieves 65.94% average accuracy, competitive with representative ∼70B open-source models on the same evaluation suite. These results show that lightweight expert orchestration can provide a parameter-efficient and auditable route to competitive medical MCQA accuracy by reserving evidence-grounded arbitration for contested cases.

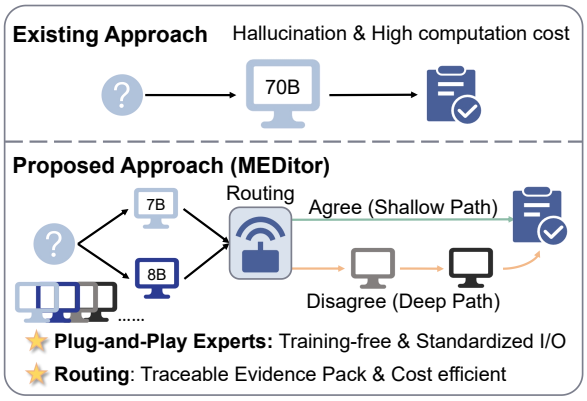


Figure 1: Schematic overview of MEDitor. MEDitor follows a triage-style routing strategy: it exits early when lightweight experts agree (Shallow Path) and escalates only disagreement cases to an evidence-grounded arbitration stack (Deep Path), reducing unnecessary Deep-Path calls while preserving traceable decision artifacts.

1 Introduction

Large Language Models (LLMs) are increasingly explored for clinical decision support, yet real-world deployment is constrained by privacy policies and strict inference budgets. Under these constraints, proprietary frontier-model APIs (e.g., GPT-4 (OpenAI, 2023)) are frequently infeasible, and deploying large open-source models (e.g., Gemma (Gemma Team, 2024)) on-premises can be cost-prohibitive, motivating the use of smaller, privacy-compliant experts. However, deploying a single lightweight model yields a pronounced reliability bottleneck under the long-tailed difficulty distribution of medical queries. Although lightweight models achieve favorable cost and latency on easy instances, their error rates can increase on difficult queries requiring compositional or multi-step reasoning, where errors are consequential. In addition, the rapidly evolving open-source model stack exhibits non-monotonic behavior across benchmarks and failure modes, making upgrades difficult to validate and increasing operational risk. This motivates our central objective: to recover near-large-model reliability using only lightweight, interchangeable, privacy-compliant components, while routing computation in proportion to query difficulty and maintaining verifiable decision traces.

To address this, we propose MEDitor, a training-free, open-source, plug-and-play multi-agent routing protocol for medical QA, as shown in Figure 1. MEDitor operationalizes a conditional-

The code is available at: https://github.com/100120023003/MEDitor-code

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 42848–42862
October 24-29, 2026 ©2026 Association for Computational Linguistics

computation strategy: straightforward cases are resolved through a low-cost Shallow Path, whereas contested cases are escalated to a more expensive Deep Path for evidence-grounded arbitration. Concretely, MEDitor runs two heterogeneous lightweight experts and uses disagreement as a routing signal. MEDitor targets a resource-aware operating point by concentrating additional computation on cases where expert disagreement indicates elevated error risk. Agreement cases terminate through the Shallow Path, while contested cases receive evidence-grounded arbitration through the Deep Path.

A key requirement in medical settings is traceability of the final decision. Retrieval alone does not guarantee faithful reasoning: generated claims may still be weakly supported or misaligned with the retrieved evidence. To mitigate this, MEDitor uses candidate-constrained, evidence-grounded arbitration in the Deep Path. An Evidence-Grounded Coordinator (EGC) constructs a candidate-aligned evidence pack, and a Constrained Judge resolves disagreement between expert proposals under a deterministic fallback. This design separates candidate proposal from candidate verification, improving auditability while keeping the protocol modular and lightweight. MEDitor's main technical contribution is the integration of disagreement-conditioned routing, standardized expert artifacts, candidate-aligned evidence construction, fixed-candidate rationale refinement, constrained judging, and deterministic fallback into a single training-free protocol.

We summarize main contributions as follows:

- **A training-free, plug-and-play routing protocol for medical MCQA.** We present MEDitor, a modular multi-agent routing framework that treats lightweight medical LLMs as interchangeable experts through a standardized artifact interface, facilitating on-premises deployment without additional training.
- **Evidence-grounded arbitration with traceable decision artifacts.** We design a Deep-Path resolution stack that separates candidate proposal from candidate verification. An Evidence-Grounded Coordinator assembles option-aligned evidence packs for competing candidates, and a Constrained Judge resolves disagreement under a deterministic fallback, improving reliability while preserving transparent decision traces.
- **A systematic empirical study with reproducible routing and arbitration analyses.** We conduct experiments on five medical multiple-choice QA benchmarks, including MedXpertQA-Text for expert-level specialist reasoning, and report results stratified by agreement/disagreement with targeted ablations of grounding, refinement, and constrained judging. The accompanying routing diagnostics and deployment measurements characterize when deeper deliberation is invoked, what accuracy gains it provides, and what latency/memory overhead it incurs.

2 Related Work

2.1 Benchmarks and Medical LLMs

We evaluate protocol-level reliability on five medical QA benchmarks spanning exam-style clinical reasoning, literature-grounded biomedical QA, and expert-level specialist reasoning, including MedQA (Jin et al., 2021), MedMCQA (Pal et al., 2022), PubMedQA (Jin et al., 2019), the medical subset of MMLU (Hendrycks et al., 2021), and MedXpertQA-Text (Zuo et al., 2025). We use only the text-only portion of MedXpertQA and do not extend the evaluation to multimodal VQA. We also note explanation-annotated benchmarks such as MedExQA (Kim et al., 2024c). In parallel, open-source biomedical LLMs increasingly narrow the gap to frontier systems via continued pretraining and instruction tuning, including generalist pipelines (e.g., UltraMedical (Zhang et al., 2024), ChiMed-GPT (Tian et al., 2024)) and specialist models (e.g., BioMistral (Labrak et al., 2024), HuatuoGPT-o1 (Chen et al., 2024), OpenBioLLM (Pal and Sankarasubbu, 2024), Med42 (Christophe et al., 2024)). Our work complements model-centric advances by targeting protocol-level reliability under a modular, swappable expert pool.

2.2 Hallucination and Evidence Grounding

RAG is widely used to mitigate medical hallucinations, and large-scale medical RAG evaluations highlight both gains and sensitivity to corpora, retrievers, and long-context failures (Xiong et al., 2024). Medical hallucination has also been explicitly benchmarked (Pal et al., 2023; Pandit et al., 2025), and abstention/selective prediction provides a practical handle for controlling risk under uncertainty (Wen et al., 2025; Machcha et al., 2025;

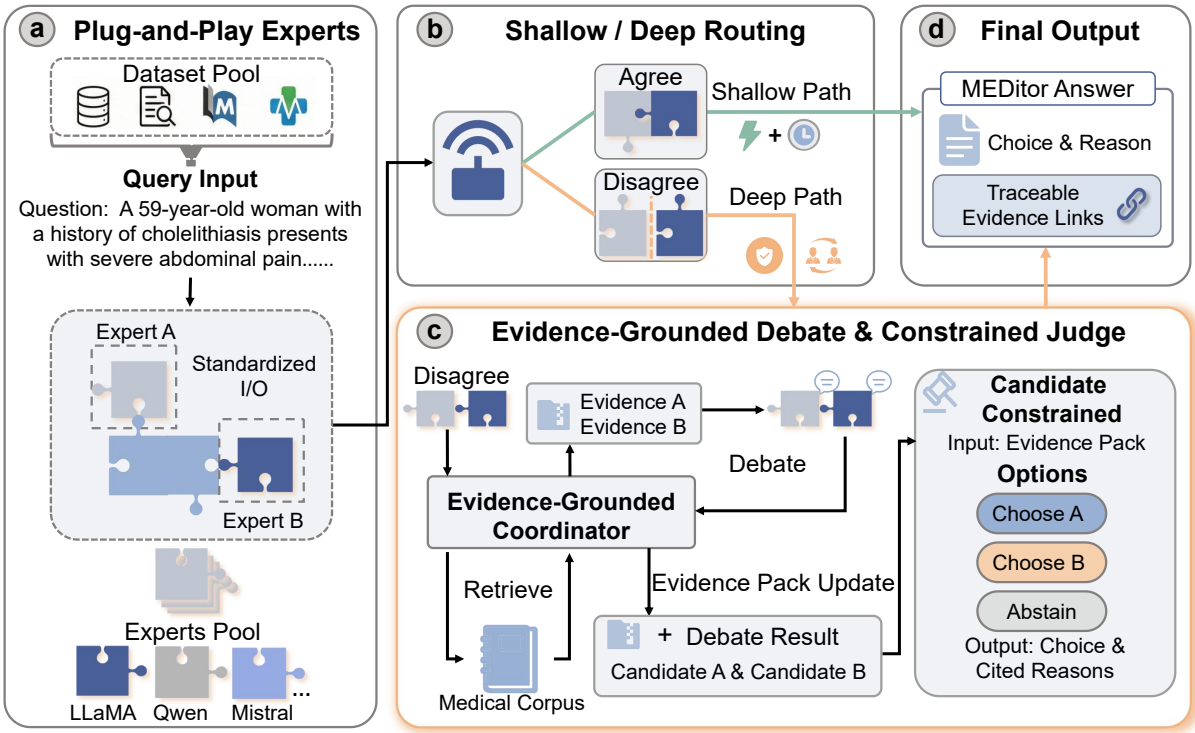


Figure 2: System architecture of MEDitor. a) **Plug-and-play experts**: heterogeneous lightweight LLMs are treated as interchangeable experts via a standardized artifact interface. b) **Disagreement-triggered routing**: if experts agree, MEDitor exits early (Shallow Path); otherwise it escalates to a Deep Path. c) **Evidence-Grounded Coordinator (EGC)**: top-K passages are retrieved from a medical corpus and organized into option-aligned evidence packs for the two fixed candidates. d) **Candidate-constrained arbitration and traceable output**: the debate module refines rationales without changing the candidate set, and the judge selects only between the two expert-induced candidates or abstains; the final answer is accompanied by evidence references and a decision trace.

Mao et al., 2025). MEDitor builds on these insights with option-aligned evidence packs and candidate-constrained comparison.

2.3 Multi-agent Collaboration

Agentic collaboration has been explored for medical reasoning and decision-making (Tang et al., 2024; Kim et al., 2024b) and supported by orchestration frameworks such as HuggingGPT (Shen et al., 2023) and AutoGen (Wu et al., 2024); interactive medical simulators further benchmark multi-agent settings (Li et al., 2024; Fan et al., 2024). A recurring limitation is that many systems apply uniform deliberation, incurring high cost and latency, while LLM-based arbitration can exhibit systematic evaluation biases (Zheng et al., 2023; Liu et al., 2023; Gu et al., 2024). MEDitor integrates triage-style routing with candidate-constrained arbitration (Prometheus-7B (Kim et al., 2024a)) to support bounded arbitration and artifact-level auditability.

3 Method

MEDitor is a training-free routing protocol for medical multiple-choice QA (Figure 2). Each instance is $x = (q, O)$, where q is the question and $O = \{o_1, \ldots, o_m\}$ is the set of answer options. MEDitor queries two heterogeneous experts, denoted by $e_{\mathbf{A}}$ and $e_{\mathbf{B}}$, through a standardized artifact interface. Each expert returns a candidate option together with a lightweight artifact that summarizes its rationale and decisiveness.

MEDitor uses agreement as a routing signal. If the two experts agree, the system returns the shared answer through a Shallow Path. Otherwise, it escalates to a Deep Path that performs candidate-constrained, evidence-grounded arbitration: an Evidence-Grounded Coordinator (EGC) constructs a candidate-aligned evidence pack, an optional debate module refines expert rationales, and a Constrained Judge resolves the disagreement under a deterministic fallback. For escalated instances, MEDitor records evidence references and decision metadata as a traceable decision pack.

3.1 Setup and Routing

Given an instance x, MEDitor interacts with heterogeneous experts through a standardized minimal contract, which enables modular replacement of expert and retrieval components. We consider two experts $e_{\mathbf{A}}$ and $e_{\mathbf{B}}$ and optionally apply self-consistency (Wang et al., 2023) by drawing n samples per expert. Each expert produces (i) a candidate option index $y_e \in \{1, \ldots, m\}$, (ii) a representative rationale r_e, and (iii) a decisiveness proxy $\Delta_e \in [0, 1]$ that captures within-expert stability (Appendix B). We encapsulate these outputs as

$$H_e = (y_e, r_e, \Delta_e). \quad (1)$$

We denote the expert invocation as $P(e, x, n)$ and define $C(H_e) = y_e$ as the candidate option.

Routing is determined by agreement between the two expert candidates. When $y_{\mathbf{A}} = y_{\mathbf{B}}$, MEDitor terminates early and returns the shared answer. When $y_{\mathbf{A}} \neq y_{\mathbf{B}}$, it invokes a structured Deep Path consisting of evidence construction, optional rationale refinement, and constrained arbitration. This conditional design concentrates expensive computation on instances where base-expert errors are more likely, reducing Deep-Path calls relative to a no-routing policy.

3.2 Evidence Pack via EGC

For Deep-Path cases, an Evidence-Grounded Coordinator (EGC) retrieves and organizes candidate-aligned evidence into an evidence pack

$$E = \mathcal{G}(x, H_{\mathbf{A}}, H_{\mathbf{B}}) = (V_{\mathbf{A}}, V_{\mathbf{B}}, U, I_{\mathbf{A}}, I_{\mathbf{B}}), \quad (2)$$

where $V_{\mathbf{A}}$ and $V_{\mathbf{B}}$ denote evidence aligned to the two competing candidates, U stores shared or uncertain snippets, and $I_{\mathbf{A}}, I_{\mathbf{B}}$ are traceable reference sets with normalized support weights. We summarize candidate support by

$$S_e = \sum_{(p_{\mathrm{id}}, w) \in I_e} w, \qquad S_{\mathbf{A}} + S_{\mathbf{B}} = 1. \quad (3)$$

EGC acts only as a coordinator: it organizes evidence for the fixed candidate set and does not output a final answer. Full reference-set definitions, passage identifiers, and normalization details are deferred to Appendix C.

3.3 Evidence-Grounded Debate

For disagreement cases, MEDitor optionally performs three rounds of evidence-grounded rationale refinement between the two experts. At each round, each expert receives the EGC evidence and the other expert's preceding rationale, and updates its own rationale while keeping y_e and Δ_e fixed: $H'_e = (y_e, r'_e, \Delta_e)$. When debate is disabled, we set $H'_e = H_e$.

Algorithm 1 MEDitor: Routing with EGC and Constrained Judge

Require: Instance x; experts $e_{\mathbf{A}}, e_{\mathbf{B}}$; samples n
Ensure: Final label y

$H_{\mathbf{A}} \leftarrow P(e_{\mathbf{A}}, x, n)$
$H_{\mathbf{B}} \leftarrow P(e_{\mathbf{B}}, x, n)$
$y_{\mathbf{A}} \leftarrow C(H_{\mathbf{A}})$; $y_{\mathbf{B}} \leftarrow C(H_{\mathbf{B}})$
if $y_{\mathbf{A}} = y_{\mathbf{B}}$ **then**
  return $y_{\mathbf{A}}$
end if
$E \leftarrow \mathrm{EGC}(x, H_{\mathbf{A}}, H_{\mathbf{B}})$
$H'_{\mathbf{A}} \leftarrow H_{\mathbf{A}}$; $H'_{\mathbf{B}} \leftarrow H_{\mathbf{B}}$
if DEBATE **then**
  $(H'_{\mathbf{A}}, H'_{\mathbf{B}}) \leftarrow D(H_{\mathbf{A}}, H_{\mathbf{B}}, E)$
end if
$d \leftarrow \mathrm{JUDGE}(H'_{\mathbf{A}}, H'_{\mathbf{B}}, E)$
if $d = y_{\mathbf{A}}$ **then**
  return $y_{\mathbf{A}}$
end if
if $d = y_{\mathbf{B}}$ **then**
  return $y_{\mathbf{B}}$
end if
return $\mathrm{FALLBACK}(H'_{\mathbf{A}}, H'_{\mathbf{B}}, E)$

3.4 Constrained Judge and Final Decision

A Constrained Judge arbitrates between the two fixed candidates using the evidence pack. In our implementation, the judge outputs exactly one control token from {`CHOOSE_A`, `CHOOSE_B`, `ABSTAIN`}, which is parsed into $d \in \{y_{\mathbf{A}}, y_{\mathbf{B}}, N\}$, where N denotes abstention. The judge compares only the two expert-induced candidates and does not answer from scratch.

When the judge abstains, MEDitor resolves the case deterministically using expert decisiveness and evidence support:

$$\mathrm{score}(e) = \alpha \Delta_e + (1 - \alpha) S_e, \quad \alpha \in [0, 1]. \quad (4)$$

The system then returns the candidate with the larger score. Exact tie-breaking details are given in Appendix A. This design keeps arbitration reproducible while preserving a bounded, traceable decision space. Algorithm 1 summarizes the full decision procedure.

4 Experiments

We evaluate MEDitor for five medical multiple-choice QA benchmarks. Our goal is to improve accuracy and traceability for open-source experts (typically ∼7B) through a structured consultation

stack comprising EGC, Evidence-Grounded Debate, and a Constrained Judge, while characterizing the latency and memory overhead introduced by routed arbitration. Unless stated otherwise, all reported numbers are exact-match accuracy (%) on the official evaluation splits.

4.1 Benchmarks

We report results on five medical QA benchmarks spanning exam-style clinical reasoning, literature-grounded biomedical QA, and expert-level specialist reasoning: MedQA (Jin et al., 2021), PubMedQA (Jin et al., 2019), MedMCQA (Pal et al., 2022), the medical subset of MMLU (Hendrycks et al., 2021), and MedXpertQA-Text (Zuo et al., 2025). MedXpertQA-Text is the text-only subset of MedXpertQA and evaluates harder specialist-level medical reasoning without introducing multimodal inputs.

For PubMedQA, which uses three options, we map {Yes, No, Maybe} to {A, B, C} for uniform prompting/parsing and map back for evaluation. The remaining datasets are evaluated under their original multiple-choice formats. We follow each dataset's official evaluation protocol and compute exact-match accuracy as the primary metric. Dataset statistics and formatting details are deferred to Table A1.

4.2 Experimental setup

We instantiate MEDitor with two heterogeneous open-source medical experts, denoted Expert A and Expert B, together with a Coordinator and a Constrained Judge. For a given run, the checkpoint assigned to each role is fixed and reported explicitly in the experiment configuration. Rather than hard-coding one fixed backbone combination in the main script, our implementation specifies the role-specific models at runtime via separate arguments for Expert A, Expert B, the Coordinator, and the Judge. In the default MEDitor instantiation, Expert A is UltraMedical-8B, Expert B is Huatuo-8B, the Coordinator is Llama-3.1-8B-Instruct, and the Constrained Judge is Prometheus-7B.

The Debate stage does not introduce a separate model. On disagreement cases, the same Expert A and Expert B checkpoints conduct three rounds of rationale refinement. In each round, both experts receive the EGC evidence and the other expert's preceding rationale and may revise only their rationale. The candidate label y_e and decisiveness score Δ_e remain fixed throughout the debate. Final arbitration is performed by Prometheus-7B, which outputs only CHOOSE_A, CHOOSE_B, or ABSTAIN. Expert A and Expert B use the same prompt template and decoding configuration, so their complementarity comes from checkpoint differences rather than prompt asymmetry.

For MEDitor variants and reimplemented multi-agent baselines, we keep benchmark formatting, answer parsing, retrieval backend, and role assignments fixed whenever applicable, and vary only the orchestration protocol. Single-model baselines are evaluated with the same benchmark formatting and answer parser but do not use retrieval or arbitration modules. Thus, the comparisons isolate the effect of routing and arbitration under a fixed role configuration, rather than conflating protocol changes with formatting, retrieval, or backbone differences.

Expert prompting and decoding. Expert A and Expert B receive the same question, option set, prompt template, and decoding configuration. For both experts, we use self-consistency with $n = 10$ samples and temperature 0.7; the final expert candidate is obtained by majority voting over sampled option labels. Thus, the two experts differ only in their underlying checkpoints rather than in prompt design or decoding hyperparameters. This setting isolates model-level complementarity from prompt-induced heterogeneity and makes the expert interface easier to reproduce. The Coordinator and Constrained Judge use fixed role-specific prompts for evidence organization and candidate-constrained arbitration, respectively.

Expert artifacts and cached base predictions. For each expert $e \in \{A, B\}$ and instance x, we construct an artifact $H_e = (y_e, r_e, \Delta_e)$, where y_e is the predicted option, r_e is a representative rationale, and $\Delta_e \in [0, 1]$ is a decisiveness proxy derived from the expert outputs. For both experts, we adopt self-consistency-style sampling and majority voting (Wang et al., 2023), using $n = 10$ samples with temperature 0.7, and Δ_e is computed from the vote margin between the top two option labels. We precompute the base expert artifacts for all instances and reuse them across ablations so that all downstream comparisons share the same underlying expert behavior.

Routing and evaluated subsets. We define the disagreement subset as the fixed set of instances for which the precomputed candidates differ, i.e., $y_{\mathbf{A}} \neq y_{\mathbf{B}}$, and report Acc@Dis (the accuracy of the

Model	MedQA	PQA	MMCQA	MMLUM	MXQAT	Avg_5
∼7B single model						
UltraMedical-8B (Zhang et al., 2024)	74.07	75.20	**63.16**	70.98	9.91	58.66
BioMistral-7B (Labrak et al., 2024)	42.65	73.18	42.40	53.62	8.79	44.13
Huatuo-8B (Chen et al., 2024)	**78.95**	77.80	60.38	77.81	**16.48**	**62.28**
Llama-3.1-8B-Instruct (Llama Team, 2024)	58.65	**78.40**	57.99	73.92	13.83	56.56
Qwen3-8B (Yang et al., 2025)	66.50	72.20	60.81	**80.81**	13.79	58.83
OpenBioLLM-8B (Pal and Sankarasubbu, 2024)	51.76	71.10	54.91	62.90	8.63	49.86
Meerkat-8B (Kim et al., 2025)	74.20	75.80	62.70	74.30	16.28	60.66
∼70B single model						
OpenBioLLM-70B (Pal and Sankarasubbu, 2024)	78.16	**79.70**	**74.01**	74.86	17.88	64.92
Med42-70B (Christophe et al., 2024)	66.60	67.20	60.58	73.71	16.65	56.95
Llama-3.1-70B-Instruct (Llama Team, 2024)	**79.42**	77.80	71.83	**79.89**	**18.91**	**65.57**
∼7B multi-agent collaboration						
MedAgents (Tang et al., 2024)†	54.30	72.48	54.90	75.90	15.90	54.70
MDAgents (Kim et al., 2024b)†	67.80	73.40	53.60	75.80	15.20	57.16
MEDitor (Shallow-only)	78.01	76.50	61.77	74.39	16.32	61.40
MEDitor (Debate only)	79.41	78.80	64.18	78.77	16.95	63.62
MEDitor (Judge only)	78.94	77.80	64.92	79.34	17.04	63.61
MEDitor (Proposed)	**82.46**	**82.20**	**66.12**	**81.34**	**17.57**	**65.94**

Table 1: Overall comparison across five medical QA benchmarks. Avg_5 is computed over MedQA, PubMedQA, MedMCQA, MMLU-med, and MedXpertQA-Text. † indicates reimplementation using the same expert backbones as MEDitor, with the orchestration protocol changed to follow the corresponding method.

disagreement subset) on this subset. For disagreement cases, MEDitor applies the routed Deep Path consisting of: (i) Evidence-Grounded Coordinator retrieval and structuring with top-$K = 5$ passages, (ii) an optional three-round rationale-refinement debate in which the two experts exchange evidence-conditioned rationales while keeping their candidate labels fixed, (iii) a Constrained Judge that outputs either $y_{\mathbf{A}}$, $y_{\mathbf{B}}$, or abstain. Abstentions are resolved by a deterministic fallback. We measure efficiency using both the disagreement rate and the Deep-Path module calls per instance.

4.3 Performance Comparison

Table 1 reports accuracy across five medical QA benchmarks. We organize the compared systems into three tiers: (i) individual ∼7B medical experts, (ii) representative ∼70B models, and (iii) ∼7B multi-agent systems built from lightweight experts. To save space in compact tables, we abbreviate PubMedQA as PQA, MedMCQA as MMCQA, MMLU-med as MMLUM, and MedXpertQA-Text as MXQAT.

Across the five benchmarks, MEDitor consistently outperforms both single-expert baselines and shallow multi-agent variants. Notably, MEDitor reaches 65.94% Avg_5, slightly exceeding Llama-3.1-70B-Instruct and OpenBioLLM-70B under the same five-benchmark average. This result supports the central parameter-efficiency claim: protocol-level orchestration of lightweight experts can reach 70B-competitive accuracy without relying on a single monolithic 70B medical model. Given the small Avg_5 margin against the strongest ∼70B baseline, we interpret this result as 70B-competitive performance rather than statistically decisive superiority.

Using paired bootstrap resampling over evaluation instances, MEDitor achieves an Avg5 of 65.94% (95% CI: [64.96, 66.91]), compared with 65.57% (95% CI: [64.58, 66.55]) for Llama-3.1-70B-Instruct. The paired difference is +0.37 percentage points (95% CI: [-0.44, 1.18], $p = 0.37$), indicating no statistically significant difference. We therefore characterize MEDitor as 70B-competitive rather than statistically superior. In contrast, MEDitor improves over the matched Shallow-only system by 4.54 points (95% CI: [3.75, 5.34], $p < 0.001$), confirming that the gain from the full routed protocol over shallow agreement routing is statistically reliable.

MedXpertQA-Text remains substantially harder than conventional medical QA benchmarks, as reflected by lower absolute scores across both ∼7B and ∼70B open-source models. Because MedXpertQA-Text is substantially harder than the other four benchmarks, Avg_5 is lower in absolute

value than the previously reported four-benchmark average; we therefore emphasize both per-dataset performance and the five-benchmark average. This comparison should be interpreted as evidence of 70B-competitive accuracy under a lightweight-expert orchestration protocol, not as a claim of lower single-query latency than all baselines.

Finally, MEDitor's modular design standardizes the expert interface, allowing base models to be replaced without changing the core pipeline. For external multi-agent baselines marked with †, we use the same expert backbones as MEDitor and change only the orchestration protocol, so the comparison controls for base expert identity rather than conflating framework differences with backbone differences. We validate MEDitor's robustness to expert replacement in Section 4.4.

The Shallow-only, Debate only, and Judge only rows in Table 1 are not external baselines; they are reduced MEDitor variants that isolate different protocol components under the default expert pair. Shallow-only tests consensus-based early stopping, Debate only tests disagreement-triggered rationale refinement, and Judge only tests candidate-constrained arbitration. All three variants outperform the matched-backbone external multi-agent baselines, indicating that the proposed routing and arbitration primitives are individually competitive. We do not report an EGC-only row because the Evidence-Grounded Coordinator is not a standalone solver: it organizes candidate-aligned evidence for downstream arbitration but does not itself output the final answer. Its effect is therefore evaluated in Section 4.6 through partial stacks that include EGC with Debate and/or Judge.

4.4 Robustness to expert replacement

To test protocol robustness under expert replacement, we replace the default expert pair with Qwen3-8B and Meerkat-8B while keeping the input format, retrieval setup, Coordinator, and Judge unchanged. Table 2 reports results across the same five-benchmark evaluation suite. MEDitor improves over both constituent experts and the Shallow-only variant, indicating that the gains are not due to simple ensembling alone but to disagreement-triggered evidence-grounded arbitration. Under the updated expert pair, MEDitor achieves 64.04 Avg_5, outperforming Qwen3-8B, Meerkat-8B, and Shallow-only by 5.21, 3.38, and 4.29 points, respectively. These results support that MEDitor's gains arise from protocol-level orchestration rather than dependence on a specific expert pair.

System	MedQA	PQA	MMCQA	MMLUM	MXQAT	Avg_5
Qwen3-8B	66.50	72.20	60.81	80.81	13.79	58.83
Meerkat-8B	74.20	75.80	62.70	74.30	16.28	60.66
Shallow-only	70.69	73.40	61.55	78.14	14.97	59.75
MEDitor	**76.89**	**77.20**	**64.78**	**83.17**	**18.14**	**64.04**

Table 2: Expert-replacement robustness across five benchmarks. Qwen3-8B and Meerkat-8B are the constituent single-expert baselines; Shallow-only and MEDitor use them as the replaced expert pair.

4.5 Routing Analysis

We conduct routing diagnostics on the four standard benchmarks, where fully cached expert artifacts are available for agreement/disagreement stratification. MedXpertQA-Text is included in the main accuracy comparison but excluded from this routing-specific analysis. On these four benchmarks, disagreement provides a useful triage signal, concentrating expensive Deep Path resources on the subset with elevated error risk. The experts reach a consensus on 44.9%–76.2% of queries, which are resolved via the low-cost Shallow Path and bypass EGC, debate, and judging calls. Consequently, the computationally intensive Deep Path is invoked for only 23.8%–55.1% of instances.

Beyond aggregate accuracy, we examine what the routing signal captures. Using the precomputed expert artifacts, we partition each benchmark into an agreement subset routed to the Shallow Path and a disagreement subset routed to the Deep Path. Figure 3a) shows a pronounced difficulty gap: agreement cases are handled reliably, whereas disagreement cases are substantially harder. Figure 3b) reports Acc@Dis on this fixed disagreement subset and compares MEDitor against the better of the two experts, defined per dataset as $\max(\text{Acc@Dis}_{\mathbf{A}}, \text{Acc@Dis}_{\mathbf{B}})$. Across the four standard benchmarks, MEDitor consistently improves on contested instances, suggesting that evidence-grounded coordination and candidate-constrained judging help select the better-supported hypothesis rather than re-solving the question. Figure 3c) compares module-call cost under two policies: *No Routing* sends every instance through EGC/debate/judge, whereas *Routing* invokes these modules only for disagreement cases.

Table 4 separates two structural sources of unrecoverable error. Shared expert errors terminate in the Shallow Path, while disagreement cases

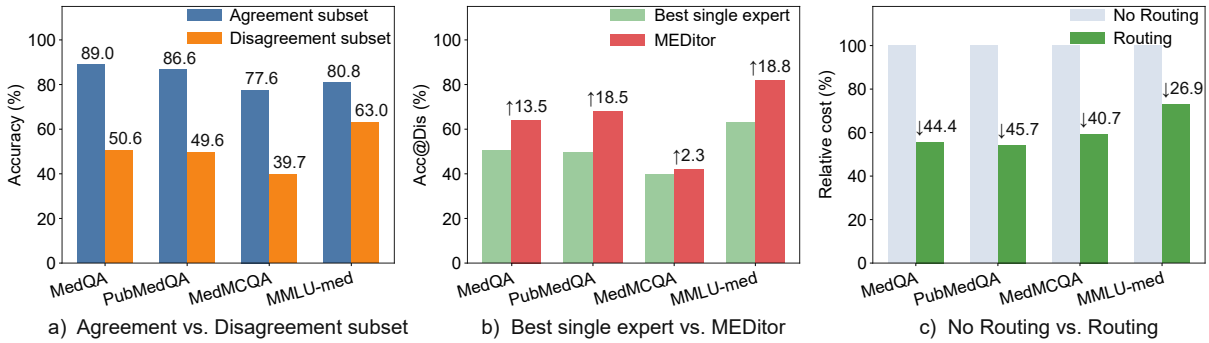


Figure 3: Routing necessity and cost. a) Accuracy gap between the agreement subset (Shallow Path; experts predict the same option, $y_{\mathbf{A}} = y_{\mathbf{B}}$) and the disagreement subset (Deep Path; $y_{\mathbf{A}} \neq y_{\mathbf{B}}$), showing that most difficulty concentrates on disputed instances. b) Accuracy on the fixed disagreement subset (Acc@Dis), defined by precomputed expert predictions $y_{\mathbf{A}} \neq y_{\mathbf{B}}$. "Best single expert" is computed per dataset as $\max(\text{Acc@Dis}_{\mathbf{A}}, \text{Acc@Dis}_{\mathbf{B}})$ on the same subset. MEDitor consistently improves Acc@Dis via Evidence-Grounded, candidate-constrained arbitration. c) Expected relative compute of two policies under the same module-call accounting: No Routing runs EGC/debate/judge on every instance and is normalized to 100%, whereas Routing invokes the Deep Path only on disagreements, with savings determined by the benchmark-specific disagreement rate.

Setting	MedQA	PQA	MMCQA	MMLUM	MXQAT	Avg_5
Shallow-only	78.01	76.50	61.77	74.39	16.32	61.40
EGC+Debate	81.15	79.80	65.34	78.82	16.89	64.40
EGC+Judge	79.89	79.30	65.11	80.37	17.23	64.38
Debate+Judge	81.34	79.40	65.23	79.91	17.17	64.61
Full MEDitor	**82.46**	**82.20**	**66.12**	**81.34**	**17.57**	**65.94**

Table 3: End-to-end ablation across five benchmarks. All variants use the default expert pair and the same disagreement-triggered routing rule.

Dataset	Agree	Acc@Agr.	AW/All	Cov@Dis	Oracle
MedQA	73.92	88.95	8.17	82.53	87.27
PubMedQA	76.20	86.61	10.20	88.24	87.00
MedMCQA	67.80	77.57	15.20	72.53	75.95
MMLU-med	44.90	80.80	8.62	88.50	85.04

Table 4: Routing failure decomposition (%). AW/All denotes agree-but-wrong cases among all instances, Cov@Dis is the fraction of disagreement cases for which at least one expert proposes the gold answer, and Oracle is the accuracy ceiling of the current disagreement-only, candidate-constrained protocol.

whose gold answer is absent from $\{y_A, y_B\}$ fall outside the candidate-constrained decision space. Together, these two terms yield oracle ceilings of 75.95–87.27% across the four routing-analysis benchmarks.

4.6 Ablation studies

We ablate Deep-Path components using the same cached expert artifacts and the same disagreement-triggered routing rule. Agreement cases are handled identically by the Shallow Path across all variants; therefore, end-to-end differences mainly reflect how each variant resolves the fixed disagreement subset defined by $y_{\mathbf{A}} \neq y_{\mathbf{B}}$.

Table 3 compares Shallow-only, partial Deep-Path variants, and the full MEDitor stack. EGC provides candidate-aligned evidence, Debate refines rationales without changing the candidate set, and the Constrained Judge selects between the two expert-induced candidates or abstains under deterministic fallback. Full MEDitor achieves the best Avg_5, suggesting that evidence grounding, rationale refinement, and bounded arbitration provide complementary gains.

Since EGC does not directly output final answers, we evaluate it through partial stacks rather than as an EGC-only solver. The non-monotonic behavior of partial variants across datasets also suggests that isolated modules are not always sufficient: arbitration without adequate grounding may be brittle, while debate without final constrained judging may not fully resolve candidate conflict.

As a complementary error analysis, we examine how the Deep Path changes predictions on the 2,398 disagreement cases, using Expert A as the fixed reference. MEDitor converts 672 initially incorrect predictions to correct answers and changes 293 initially correct predictions to incorrect answers; another 679 correct and 754 incorrect predictions remain unchanged. The resulting net correction is 379 cases. These transitions show that the Deep Path contributes primarily by correcting contested expert decisions, while a substantial residual error set remains unresolved. Fallback performance is stable across $\alpha \in \{0, 0.25, 0.5, 0.7, 0.75, 1.0\}$,

α	0	.25	.50	.70	.75	1.0
Fallback Acc.	36.2	36.7	37.0	37.1	37.1	36.8
Overall Acc.	65.83	65.90	65.93	65.94	65.94	65.92

Table 5: Sensitivity to the fallback coefficient α. Overall accuracy varies by at most 0.11 points over the tested range.

with end-to-end accuracy varying by 0.11 points (Table 5). We use $\alpha = 0.7$ in the reported experiments.

4.7 Deployment Cost Characterization

We characterize MEDitor as a routed-arbitration system and report its end-to-end latency and aggregate peak GPU memory under warmed serving. On the five-benchmark evaluation suite, only 35.96% of instances enter the Deep Path, whereas a no-routing policy would apply EGC, Debate, and Judge to every instance. MedXpertQA-Text is excluded from the routing figure but included in the aggregate deployment-cost estimate, where cached expert artifacts are available for computing the overall Deep-Path rate. Table A2 reports the corresponding latency, memory, and no-routing cost accounting.

4.8 Evaluation on Real-World Clinical Cases

To further assess whether MEDitor can operate beyond benchmark-style QA, we evaluate it on 312 de-identified hospital-derived instances from an arrhythmia department, focusing on atrial-fibrillation-related clinical tasks such as disease diagnosis, evidence grounding and field extraction. The hospital-domain base model is Huatuo-8B. The study passed the hospital's internal ethics review process, and all cases were retrospectively de-identified before model inference. The evaluation is offline and non-interventional: model outputs were not shown to clinicians, were not used for diagnosis or treatment, and did not affect patient care.

MEDitor achieves 78.0% accuracy on this real-world hospital set, compared with 57.0% for the strongest Huatuo-8B baseline, using retrospective reference labels derived from finalized clinical diagnoses or documented management decisions. These results suggest that MEDitor can improve offline performance on noisy hospital-style inputs in an atrial-fibrillation scenario. This study is intended as an offline feasibility demonstration rather than evidence of clinical deployment readiness; additional details are provided in Appendix D.

5 Conclusion

We introduced MEDitor, a training-free, plug-and-play routing protocol for medical MCQA. MEDitor uses expert disagreement to trigger an evidence-grounded Deep Path that combines candidate-aligned retrieval, rationale refinement, and constrained arbitration. Across five benchmarks, the protocol reaches 65.94% Avg_5 with lightweight open-source experts and remains competitive with the strongest $\sim$70B baseline in our evaluation. Routing confines the full Deep Path to 35.96% of instances, while the agreement/disagreement analysis shows where additional computation is most useful. The modular expert interface and traceable decision artifacts provide a practical basis for further work on auditable medical MCQA.

Limitations

MEDitor routes only disagreement cases to the Deep Path. This reduces computation but can miss agreement-but-wrong cases when both experts share the same bias. Candidate-constrained arbitration improves auditability by restricting the final decision to $\{y_{\mathbf{A}}, y_{\mathbf{B}}\}$ or abstention, but it limits recovery when both expert-induced candidates are wrong.

The Deep Path also depends on retrieval quality and judge reliability. Missing, weak, or poorly aligned evidence can lead to inconclusive arbitration, and a compact Constrained Judge may miss fine-grained biomedical distinctions under ambiguous evidence. Our deployment cost characterization reports both routing-based proxies and latency/memory measurements, but these results are tied to a fixed serving stack and may vary with hardware, batching, and implementation.

Finally, although we include a retrospective feasibility test on de-identified hospital cases from an arrhythmia department, this experiment is offline, non-interventional, and aggregate-level. The study passed the hospital's internal ethics review process, and model outputs were not used for patient care. Therefore, the hospital-data experiment should not be interpreted as clinical validation or evidence of safety or effectiveness in deployment. Clinical use would require prospective validation, clinician-in-the-loop evaluation, safety monitoring, and institution-specific governance.

Ethical Considerations

This work uses a small retrospective set of de-identified real-world hospital cases only for offline technical evaluation. The hospital case study was conducted on arrhythmia-department records with atrial-fibrillation-related clinical tasks, including disease diagnosis, evidence grounding and field extraction. The study passed the hospital's internal ethics review process before analysis.

Direct identifiers were removed before model inference, and the examples reported in the paper are further redacted or paraphrased to avoid exposing patient-specific details, institution names, exact dates, or other quasi-identifiers. The case-level records, identifiable metadata, and patient-specific model outputs are not released.

The study was retrospective, offline, and non-interventional. Model inference was isolated from the clinical workflow and had no role in diagnosis, treatment, or other patient-care decisions. Any future clinical deployment would require prospective validation, clinician-in-the-loop evaluation, safety monitoring, and institution-specific governance.

Acknowledgments

This work was supported by the Major Key Project of Peng Cheng Laboratory under Grant PCL2025A12 and in part by National Natural Science Foundation of China under Grant 62202248.

References

Junying Chen, Zhenyang Cai, Ke Ji, Xidong Wang, Wanlong Liu, Rongsheng Wang, Jianye Hou, and Benyou Wang. 2024. Huatuogpt-o1, towards medical complex reasoning with LLMs. *Preprint*, arXiv:2412.18925.

Clément Christophe, Praveen K Kanithi, Tathagata Raha, Shadab Khan, and Marco A F Pimentel. 2024. Med42-v2: A suite of clinical LLMs. arXiv preprint arXiv:2408.06142.

Zhihao Fan, Jialong Tang, Wei Chen, Siyuan Wang, Zhongyu Wei, Jun Xi, Fei Huang, and Jingren Zhou. 2024. AI hospital: Benchmarking large language models in a multi-agent medical interaction simulator. arXiv preprint arXiv:2402.09742.

Gemma Team. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2024. A survey on LLM-as-a-judge. *Preprint*, arXiv:2411.15594.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*.

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

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. 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)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Hyunjae Kim, Hyeon Hwang, Jiwoo Lee, Sihyeon Park, Dain Kim, Taewhoo Lee, Chanwoong Yoon, Jiwoong Sohn, Jungwoo Park, Olga Reykhart, Thomas Fetherston, Donghee Choi, Soo Heon Kwak, Qingyu Chen, and Jaewoo Kang. 2025. Small language models learn enhanced reasoning skills from medical textbooks. *npj Digital Medicine*, 8(1):240.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024a. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4334–4353, Miami, Florida, USA. Association for Computational Linguistics.

Yubin Kim, Chanwoo Park, Hyewon Jeong, Yik Siu Chan, Xuhai Xu, Daniel McDuff, Hyeonhoon Lee, Marzyeh Ghassemi, Cynthia Breazeal, and Hae Won Park. 2024b. Mdagents: An adaptive collaboration of LLMs for medical decision-making. In *Advances in Neural Information Processing Systems*, pages 79410–79452, Vancouver, Canada. NeurIPS 2024 (NeurIPS 37).

Yunsoo Kim, Jinge Wu, Yusuf Abdulle, and Honghan Wu. 2024c. MedExQA: Medical question answering benchmark with multiple explanations. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pages 167–181, Bangkok, Thailand. Association for Computational Linguistics.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. BioMistral: A collection of open-source pretrained large language models for medical domains. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5848–5864,

Bangkok, Thailand. Association for Computational Linguistics.

Junkai Li, Yunghwei Lai, Weitao Li, Jingyi Ren, Meng Zhang, Xinhui Kang, Siyu Wang, Peng Li, Ya-Qin Zhang, Weizhi Ma, and Yang Liu. 2024. Agent hospital: A simulacrum of hospital with evolvable medical agents. *Preprint*, arXiv:2405.02957.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Llama Team. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Sravanthi Machcha, Sushrita Yerra, Sharmin Sultana, Hong Yu, and Zonghai Yao. 2025. Do large language models know when not to answer in medical QA? In *Proceedings of the 2nd Workshop on Uncertainty-Aware NLP (UncertaiNLP 2025)*, pages 27–35, Suzhou, China. Association for Computational Linguistics.

Yuzhen Mao, Thibaut Durand, Nazanin Mehrasa, Jiawei He, and Martin Ester. 2025. Calibrating LLMs for selective prediction: Balancing coverage and risk. In *NeurIPS 2025 Workshop on Socially Responsible and Trustworthy Foundation Models*.

OpenAI. 2023. GPT-4 technical report. *Preprint*, arXiv:2303.08774.

Ankit Pal and Malaikannan Sankarasubbu. 2024. OpenBioLLMs: Advancing open-source large language models for healthcare and life sciences. Hugging Face model repository.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2023. Med-HALT: Medical domain hallucination test for large language models. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pages 314–334, Singapore. Association for Computational Linguistics.

Shrey Pandit, Jiawei Xu, Junyuan Hong, Zhangyang Wang, Tianlong Chen, Kaidi Xu, and Ying Ding. 2025. MedHallu: A comprehensive benchmark for detecting medical hallucinations in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2858–2873, Suzhou, China. Association for Computational Linguistics.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. HuggingGPT: Solving AI tasks with ChatGPT and its friends in hugging face. In *Advances in Neural Information Processing Systems*, volume 36, pages 38154–38180. Curran Associates, Inc.

Xiangru Tang, Anni Zou, Zhuosheng Zhang, Ziming Li, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. 2024. MedAgents: Large language models as collaborators for zero-shot medical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 599–621, Bangkok, Thailand. Association for Computational Linguistics.

Yuanhe Tian, Ruyi Gan, Yan Song, Jiaxing Zhang, and Yongdong Zhang. 2024. ChiMed-GPT: A Chinese medical large language model with full training regime and better alignment to human preferences. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7156–7173, Bangkok, Thailand. Association for Computational Linguistics.

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

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. 2025. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. 2024. Autogen: Enabling next-gen LLM applications via multi-agent conversations. In *First Conference on Language Modeling*.

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. 2024. Benchmarking retrieval-augmented generation for medicine. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6233–6251, Bangkok, Thailand. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Kaiyan Zhang, Sihang Zeng, Ermo Hua, Ning Ding, Zhang-Ren Chen, Zhiyuan Ma, Haoxin Li, Ganqu Cui, Biqing Qi, Xuekai Zhu, Xingtai Lv, Jin-Fang Hu, Zhiyuan Liu, and Bowen Zhou. 2024. UltraMedical:

Building specialized generalists in biomedicine. In *Advances in Neural Information Processing Systems*, volume 37, pages 26045–26081.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc.

Yuxin Zuo, Shang Qu, Yifei Li, Zhang-Ren Chen, Xuekai Zhu, Ermo Hua, Kaiyan Zhang, Ning Ding, and Bowen Zhou. 2025. MedXpertQA: Benchmarking expert-level medical reasoning and understanding. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 80961–80990. PMLR.

A Implementation Details

Serving and hardware. We serve MEDitor components and ∼7B baselines through vLLM on 4×RTX 4090 GPUs (24GB each). The ∼70B baseline is evaluated on a separate multi-GPU server with sufficient aggregate GPU memory. All experiments use the same OpenAI-compatible API interface, and all latency measurements exclude model initialization and use warmed serving processes.

Prompt contracts. Expert A and Expert B use the same expert prompt template. The prompt asks the model to answer a multiple-choice medical question and return exactly one option label from the provided option set. This shared prompt contract ensures that expert disagreement reflects checkpoint-level differences rather than prompt asymmetry. The Coordinator and Judge use separate fixed prompts: the Coordinator constructs candidate-aligned evidence without choosing a final answer, and the Judge performs candidate-constrained arbitration.

Decoding budgets. Unless otherwise stated, Expert A and Expert B use self-consistency with $n = 10$ samples, temperature 0.7, and a maximum generation budget of 256 tokens per sample. The Coordinator uses a maximum budget of 128 tokens, and the Constrained Judge uses 16 tokens. We use the same answer parser for all expert outputs.

Self-consistency and artifact construction. Self-consistency is applied symmetrically to both Expert A and Expert B. For each expert, we sample $n = 10$ label-only outputs with temperature 0.7, parse each output into an option label, and use majority voting to determine the expert candidate y_e. The decisiveness proxy Δ_e is computed from the vote margin between the top two option labels. After y_e is fixed, we invoke the same expert checkpoint once more with a candidate-conditioned rationale prompt to obtain r_e; this rationale-generation call cannot change y_e or Δ_e. The resulting expert artifacts are cached and reused across protocol ablations.

Retrieval backend and evidence pack. For disagreement cases, we use MedRAG with the MedCPT retriever and top-$K = 5$ passages. The Evidence-Grounded Coordinator organizes retrieved snippets into a candidate-aligned evidence pack with traceable passage identifiers. The Coordinator only organizes retrieved evidence for the two expert-induced candidates and does not introduce a new candidate or final answer.

Judge and fallback. The Constrained Judge outputs one control token from {`CHOOSE_A`, `CHOOSE_B`, `ABSTAIN`}. We parse `CHOOSE_A` as selecting Expert A's candidate, `CHOOSE_B` as selecting Expert B's candidate, and `ABSTAIN` as abstention. Abstentions are resolved by a deterministic fallback that combines expert decisiveness and evidence support with a fixed coefficient $\alpha = 0.7$.

Routed-arbitration cost accounting. The relevant routing comparison holds the MEDitor pipeline fixed and contrasts disagreement-triggered execution with a policy that applies EGC, Debate, and Judge to every instance. Let ρ be the disagreement rate, T_{shallow} the Shallow-only latency, and T_{extra} the additional Deep-Path latency on routed cases:

$$\begin{aligned} \mathbb{E}[T_{\text{route}}(\rho)] &= T_{\text{shallow}} + \rho T_{\text{extra}}, \\ T_{\text{noroute}} &= T_{\text{shallow}} + T_{\text{extra}}. \end{aligned} \tag{5}$$

With $\rho = 35.96\%$, $T_{\text{shallow}} = 4.03$s, and $\mathbb{E}[T_{\text{route}}] = 22.65$s, we estimate $T_{\text{extra}} = 51.78$s and $T_{\text{noroute}} = 55.81$s. Thus, routing reduces Deep-Path calls by 64.04% and expected latency by 59.4% relative to always applying the full Deep Path. These savings are relative to a no-routing MEDitor-style policy, not to single-model inference.

Dataset	Format	Choice	Test Size	Domain
MedQA	Question + Answer	A/B/C/D	1273	US Medical Licensing Examination
MedMCQA	Question + Answer	A/B/C/D and Explanations	4183	AIIMS+NEET PG entrance exams
PubMedQA	Question + Context + Answer	Yes/No/Maybe	500	PubMed paper abstracts
MMLU-med	Question + Answer	A/B/C/D	1089	Medical subjects from MMLU (multiple exam-style sources)
MedXpertQA-Text	Question + Answer	Multiple-choice A-J	2450	Expert-level medical reasoning across specialties

Table A1: Benchmark summary.

Operating point	Acc. (Avg_5)	Deep-path rate	Mean lat. (s)	P95 lat. (s)	Agg. peak GPU (GB)	Reduction vs. no-routing Deep-Path calls / lat.
Reference single-model operating points						
Best single ∼7B expert	62.28	–	2.41	3.08	20.59	–
Best single ∼70B expert	65.57	–	4.87	6.99	150.71	–
MEDitor operating points						
Shallow-only	61.40	0.00%	4.03	4.35	41.18	100.0% / n/a
Full MEDitor (routed)	**65.94**	35.96%	22.65	23.16	75.89	**64.04% / 59.4%**
No-routing Deep Path†	–	100.00%	55.81	–	–	0.0% / 0.0%

Table A2: Routed-arbitration cost accounting on the five-benchmark evaluation suite under warmed serving processes. Single-model rows are reference operating points and do not provide MEDitor-style evidence-grounded arbitration. The relevant routing comparison is between Full MEDitor and a no-routing Deep-Path policy that applies EGC, Debate, and Judge to every instance. †No-routing latency is estimated from the observed Shallow-only latency, Full MEDitor latency, and the average disagreement rate $\rho = 35.96\%$ using Eq. 5. GPU memory denotes aggregate peak allocated memory across all devices used by the corresponding serving process.

B Self-Consistency Aggregation and Decisiveness Proxy

This appendix details how we instantiate the expert artifact interface $H_e = (y_e, r_e, \Delta_e)$ under optional self-consistency. These choices are *implementation details* and do not affect MEDitor's protocol-level I/O contract.

Self-consistency sampling. For each expert $e \in \{A, B\}$ and instance $x = (q, O)$, we draw n samples (e.g., by varying decoding randomness or seeds), producing sampled option labels $\{y_e^{(j)}\}_{j=1}^{n}$ where $y_e^{(j)} \in \{1, \ldots, m\}$.

Majority-vote candidate. We count votes for each option id l:

$$\begin{aligned} v_e(l) &= \sum_{j=1}^{n} \mathbb{I}[y_e^{(j)} = l], \\ y_e &= \arg\max_{l \in \{1,\ldots,m\}} v_e(l). \end{aligned} \tag{6}$$

Ties in $\arg\max$ are broken deterministically (e.g., by a fixed ordering over option ids).

Decisiveness proxy. Let l_1 and l_2 be the top-1 and top-2 option ids under $v_e(\cdot)$ (after deterministic tie-breaking). We define a normalized margin as a simple within-expert stability proxy:

$$\Delta_e = \frac{v_e(l_1) - v_e(l_2)}{n} \in [0, 1]. \tag{7}$$

Intuitively, larger margins indicate that the expert's prediction is less sample-sensitive. When $n = 1$, we set $\Delta_e = 1$ by convention.

Candidate-conditioned rationale generation. The n self-consistency samples are used only to determine the aggregated candidate y_e and decisiveness proxy Δ_e. After majority voting fixes y_e, we invoke the same expert checkpoint once more with a separate rationale-generation prompt that includes the question, answer options, and the fixed candidate y_e. The resulting explanation is stored as r_e. This call is candidate-conditioned: it may elaborate the reasoning for y_e but cannot change the candidate label or Δ_e. Thus, rationale generation does not participate in the majority vote.

Artifact interface. Finally, the expert artifact passed to downstream MEDitor modules is

$$H_e = (y_e, r_e, \Delta_e), \tag{8}$$

which contains (i) the candidate option id, (ii) an associated rationale, and (iii) a stability proxy.

C EGC Evidence References: Passage IDs, Retriever Scores, and Weight Normalization

This appendix specifies how EGC constructs traceable, weighted evidence references $I_{\mathbf{A}}, I_{\mathbf{B}}$. The goal is to (i) preserve auditability via stable passage identifiers and (ii) provide a comparable evidence-support signal via normalized weights.

Passage identifiers. Each retrieved passage is assigned a unique passage identifier $p_{\mathrm{id}} = (\mathrm{doc}_{\mathrm{id}},\ \mathrm{span}_{\mathrm{id}})$, where $\mathrm{doc}_{\mathrm{id}}$ is the source document identifier and $\mathrm{span}_{\mathrm{id}}$ indexes a concrete text span (e.g., a chunk index under a fixed chunking policy). We store p_{id} rather than raw text to keep evidence traceable and reproducible under a fixed corpus snapshot.

Retriever scores. Given an instance x and a candidate y_e, the retriever returns a ranked list of passages $\{(p_{\mathrm{id},k}, s_k)\}_{k=1}^{K}$, where $s_k \in \mathbb{R}$ is the retriever relevance score (e.g., similarity score for dense retrieval or BM25 score for sparse retrieval). Scores are not assumed to be calibrated across queries.

Score-to-weight normalization (within-candidate). For each candidate $e \in \{\mathbf{A}, \mathbf{B}\}$, we convert retriever scores into normalized within-candidate weights:

$$\tilde{w}_k = \frac{\exp(s_k/\tau)}{\sum_{j=1}^{K} \exp(s_j/\tau)},$$

where τ is a temperature hyperparameter. These weights are used for ranking/truncation within the same candidate and are *not* assumed to be calibrated across different candidates.

Deduplication and truncation (retained mass). When multiple retrieved entries map to the same p_{id} (e.g., repeated retrieval across sub-queries), we merge them by summing weights:

$$\tilde{w}(p_{\mathrm{id}}) = \sum_{k:p_{\mathrm{id},k}=p_{\mathrm{id}}} \tilde{w}_k.$$

We then keep the top-K_e passages by $\tilde{w}(p_{\mathrm{id}})$ to form a compact candidate-specific set

$$\tilde{I}_e \ = \ \{(p_{\mathrm{id}}, \tilde{w}(p_{\mathrm{id}}))\}_{p_{\mathrm{id}} \in \mathrm{TopK}_e}.$$

We define the *retained relevance mass*

$$R_e \ = \sum_{(p_{\mathrm{id}},\tilde{w}) \in \tilde{I}_e} \tilde{w} \in (0, 1],$$

which measures how much of the within-candidate relevance distribution is preserved after truncation.

Union renormalization and evidence support (cross-candidate comparable). To compare support across the two candidates, we renormalize weights over the union $\tilde{I}_{\mathbf{A}} \cup \tilde{I}_{\mathbf{B}}$:

$$w(p_{\mathrm{id}}) \ = \ \frac{\tilde{w}(p_{\mathrm{id}})}{\sum_{(p'_{\mathrm{id}},\tilde{w}') \in \tilde{I}_{\mathbf{A}} \cup \tilde{I}_{\mathbf{B}}} \tilde{w}'},$$

$$\sum_{(p_{\mathrm{id}},w) \in I_{\mathbf{A}} \cup I_{\mathbf{B}}} w = 1,$$

and define the final reference sets

$$I_e \ = \ \{(p_{\mathrm{id}}, w(p_{\mathrm{id}}))\}_{(p_{\mathrm{id}},\tilde{w}) \in \tilde{I}_e}.$$

We then compute evidence support

$$S_e = \sum_{(p_{\mathrm{id}},w) \in I_e} w,$$

hence $S_{\mathbf{A}} + S_{\mathbf{B}} = 1$. We use S_e as the deterministic relative evidence-support signal in the abstention fallback (Section 3.4).

D Offline Clinical-Case Set

We additionally evaluate MEDitor on 312 de-identified real-world clinical cases from an arrhythmia department. The cases focus on atrial-fibrillation-related clinical tasks and are converted into text-only QA instances covering disease diagnosis, evidence grounding and field extraction. Each instance is constructed from anonymized clinical summaries, examination findings, electrocardiographic descriptions, treatment records, and candidate diagnostic or management options. Reference labels are retrospectively derived from finalized clinical diagnoses or documented management decisions.

The hospital-domain study uses Huatuo-8B as the base medical model for the clinical-task baseline, and MEDitor applies the same routed arbitration protocol to improve robustness over this base setting. Direct identifiers were removed before model inference. The study was reviewed and approved through the hospital's internal ethics review process. It is retrospective, offline, and non-interventional: model outputs were not shown to

clinicians during patient care, were not used for diagnosis or treatment, and did not affect any clinical decision.

MEDitor achieves approximately 78.0% accuracy on this real-world hospital set, compared with approximately 57.0% for the Huatuo-8B baseline. This result suggests that the routed arbitration protocol can operate on noisier hospital-style atrial-fibrillation QA inputs, but it should not be interpreted as evidence of clinical safety, clinical effectiveness, or deployment readiness.

De-identified real-world atrial-fibrillation case study

Task. Extract the documented disease diagnosis from a de-identified arrhythmia-department record. The answer should be grounded in the discharge diagnosis or other explicit diagnostic evidence in the case record.

Anonymized case context. The patient was admitted for atrial-fibrillation-related evaluation after postoperative palpitations and a persistently rapid ventricular rate. The record reports that an electrocardiogram showed atrial fibrillation with rapid ventricular response. During hospitalization, rate-control and anticoagulation-related management were documented. The final discharge diagnosis explicitly lists atrial fibrillation.

Retrieved evidence used by MEDitor.

- **Chief complaint:** atrial fibrillation was discovered for more than 20 days.
- **History of present illness:** postoperative palpitations were followed by electrocardiographic evidence of atrial fibrillation with rapid ventricular rate.
- **Treatment signal:** amiodarone was administered for ventricular-rate control, with anticoagulation-related management documented in the admission course.
- **Discharge diagnosis:** the record explicitly documents *atrial fibrillation* as the discharge diagnosis.

Multi-agent decision trace.

- **Base medical model:** Huatuo-8B is used as the hospital-domain clinical baseline.
- **Expert proposal:** one expert extracts `atrial fibrillation` directly from the discharge-diagnosis field.
- **Evidence grounding:** the retrieved snippets support the diagnosis through both electrocardiographic evidence and the final discharge diagnosis.
- **Arbitration:** MEDitor preserves the diagnosis supported by the explicit discharge-diagnosis evidence rather than expanding the answer to unrelated comorbidities or treatments.

Final answer. `atrial fibrillation`. This example illustrates that MEDitor can apply the same evidence-grounded routing and arbitration protocol to a real-world arrhythmia-department scenario, where the model must connect noisy longitudinal record fields with the final documented diagnosis while preserving a concise and auditable answer.

E System Prompts

This section summarizes the fixed prompt contracts used by MEDitor. The prompts define a reusable interface for expert proposal, evidence coordination, rationale refinement, and candidate-constrained arbitration.

Prompt contracts for MEDitor

Expert A/B. `You are a senior physician. Answer the multiple-choice medical question. Return exactly one option label from the given option set.`

EGC. `You are a retrieval coordinator for medical multiple-choice questions. Summarize retrieved evidence and map it to the two expert claims. Do not answer the question. Do not choose between Expert A and Expert B. Ground evidence using snippet identifiers such as [A1] and [B2].`

EGC output. `[Key Evidence]; [Evidence favoring Candidate A]; [Evidence favoring Candidate B]; [Conflicts / Uncertainties]; [Retrieval Note].`

Debate. `Refine rationales conditioned on EGC evidence. Keep the candidate labels fixed to the two expert-induced options. Do not introduce a new answer option.`

Judge. `Compare Candidate A and Candidate B using the evidence pack. Do not re-solve the question. Output exactly one token from {CHOOSE_A, CHOOSE_B, ABSTAIN}. Abstention is resolved by deterministic fallback.`

F Qualitative Case Studies

This section presents two qualitative examples: one agreement case resolved by the Shallow Path and one disagreement case escalated to the Deep Path.

Case study 1: Shallow Path / Agreement — id=586

Task: MedQA. **Question**: A 34-year-old man has headaches, blurry vision, erectile dysfunction, bitemporal visual field deficit, and an intrasellar mass. Which pituitary cell type is the mass derived from? **Options**: (A) Thyrotrophs (B) Lactotrophs (C) Somatotrophs (D) Gonadotrophs.
Experts: $y_A = \mathrm{B}$, $y_B = \mathrm{B} \Rightarrow$ **early stop**. **Final**: Pred = B, Gold = B.
Takeaway: Headache and bitemporal hemianopsia suggest a pituitary macroadenoma compressing the optic chiasm, and sexual dysfunction is consistent with hyperprolactinemia. MEDitor correctly returns the consensus answer without invoking EGC, Debate, or Judge.

Case study 2: Deep Path / Disagreement — id=166

Task: MedQA. **Question**: A 60-year-old man with progressive exertional dyspnea worked for 20 years in glass manufacturing. Chest radiograph shows interstitial fibrosis, reticulonodular infiltrates, and eggshell calcification of multiple lymph nodes. What is the most likely diagnosis? **Options**: (A) Berylliosis (B) Silicosis (C) Asbestosis (D) Talcosis.
Experts: $y_A = \mathrm{B}$ vs. $y_B = \mathrm{D} \Rightarrow$ **routed to Deep Path**. **EGC evidence**: glass manufacturing is consistent with silica exposure, and eggshell calcification of hilar or mediastinal lymph nodes is a classic clue for silicosis. **Judge**: selects Candidate A. **Final**: Pred = B, Gold = B.
Takeaway: EGC does not propose a new option. It grounds the two fixed candidates, after which the Constrained Judge selects the better-supported candidate.