PubMed Reasoner: Dynamic Reasoning-based Retrieval for Evidence-Grounded Biomedical Question Answering

Yiqing Zhang[1,2] **and Xiaozhong Liu**[2] **and Fabricio Murai**[2]

[1]PayPal, [2]Worcester Polytechnic Institute

yiqizhang@paypal.com, {xliu14,fmurai}@wpi.edu

Abstract

Trustworthy biomedical question answering (QA) systems must not only provide accurate answers but also justify them with current, verifiable evidence. Retrieval-augmented approaches partially address this gap but lack mechanisms to iteratively refine poor queries, whereas self-reflection methods kick in only after full retrieval is completed. In this context, we introduce PubMed Reasoner, a biomedical QA agent composed of three stages: **self-critic query refinement** evaluates MeSH terms for coverage, alignment, and redundancy to enhance PubMed queries based on partial (metadata) retrieval; **reflective retrieval** processes articles in batches until sufficient evidence is gathered; and **evidence-grounded response generation** produces answers with explicit citations. PubMed Reasoner with a GPT-4o backbone achieves **78.32%** accuracy on PubMedQA, slightly surpassing human experts, and showing consistent gains on MMLU Clinical Knowledge. Moreover, LLM-as-judge evaluations prefer our responses across: reasoning soundness, evidence grounding, clinical relevance, and trustworthiness. By orchestrating retrieval-first reasoning over authoritative sources, our approach provides practical assistance to clinicians and biomedical researchers while controlling compute and token costs. [1] [2]

1 Introduction

Trustworthiness is essential in biomedical domains. Biomedical question answering (QA) systems must be factually grounded, current, and interpretable. Yet, large language models (LLMs) that rely primarily on parametric memory can hallucinate (Kalai et al., 2025), drift out of date, or omit key evidence (Guan et al., 2023; Xu et al., 2024). While prior works have explored retrieval-augmented methods, to our knowledge, this is the first work to explicitly support citation-backed responses on biomedical QA datasets, ensuring that each response is transparently grounded in published evidence.

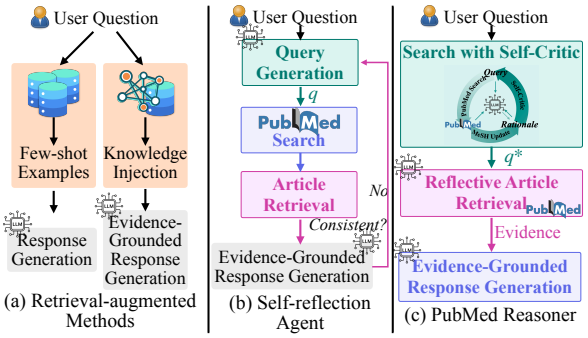

Figure 1: RAG, Self-reflection vs. PubMed Reasoner. **(a) RAG baselines:** uses few-shot exemplars or custom databases but lack retrieval feedback. **(b) Self-reflection agents:** inspired our proposal; generates responses first and only reflects after completion. **(c) PubMed Reasoner:** a search-first approach that performs self-critic query refinement, reflective article retrieval in batches w/ early stopping once evidence is sufficient, and evidence-grounded response generation with explicit citations.

Existing retrieval-augmented generation (RAG) approaches remain limited in biomedical settings (see Fig. 1a): **(i) Few-shot prompting** retrieves a few in-distribution exemplars (few-shot samples) and prompt the LLM to imitate them (Nori et al., 2023b,a). While this can improve accuracy on similar cases, it does not yield structured explanations (Singhal et al., 2025). The link between examples is built via clustering and similarity search, which requires sizable, well-labeled exemplar data. As a result, exemplar RAG is brittle and biased: it optimizes for local pattern matching rather than concept coverage or causal explanations. **(ii) Private knowledge databases** consist of bespoke stores such as entity graphs (Abu-Rasheed et al., 2024) or schema-aligned tables (Arslan et al., 2024) which

[1]The full implementation is publicly available at https://github.com/swiftzhang125/ACL_2026_PubMed_Reasoner, including code, prompt templates, LLM-as-judge evaluation, and baseline systems.

[2]A longer version with detailed algorithm, configuration and prompts is available on arXiv https://arxiv.org/pdf/2603.27335.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 16753–16763

can support step-wise explanations, but require strong priors and heavy maintenance, with limited reusability across domains. In biomedicine, where PubMed and MeSH (Medical Subject Headings) terms continue to grow, keeping such stores complete and current is prohibitively expensive.

A parallel line of work equips LLMs with web search capabilities ("deep research"), producing responses with citations. Yet, these systems typically lack the ability to constrain retrieval to authoritative biomedical sources such as PubMed, often yielding citations from less reliable or incomplete domains. Recent "self-reflection" agents (Figure 1b) provide another direction, using consistency checks (Wang et al., 2022) or reward-based signals (Leike et al., 2018; Shinn et al., 2023) to refine final answers. However, reflection occurs only after retrieval and response generation are complete, making the process computationally costly and unable to correct poor upstream retrieval or query formulation.

Our approach. We present PubMed Reasoner, a multi-stage agent that mirrors the workflow of a biomedical researcher. Unlike prior methods that reflect only after producing responses, PubMed Reasoner introduces feedback much earlier. During query planning, a self-critic evaluates candidate MeSH terms and their Boolean composition directly against live PubMed metadata, eliminating the need for static private databases. This structured feedback prevents low-quality queries from propagating downstream and balances recall with precision. The refined query is then issued to PubMed, where a reflective retriever processes articles in small batches and halts once sufficient evidence is gathered, controlling token usage. Finally, PubMed Reasoner synthesizes an evidence-grounded response with explicit inline citations, ensuring transparency and interpretability.

Instead of treating the LLM as a one-shot generator, PubMed Reasoner introduces a **dynamic reasoning-based paradigm**, orchestrating three interconnected stages over external evidence. This work makes the following contributions:

- We shift biomedical QA beyond one-shot generation by introducing an iterative workflow that plans, retrieves, and reasons over external evidence.
- We integrate self-critic query refinement with batch-wise reflective retrieval over the PubMed database, enabling robust evidence grounding without maintaining private knowledge stores.
- We demonstrate consistent improvements in accuracy, explanation quality, and computational efficiency on PubMedQA and MMLU Clinical Knowledge, outperforming strong LLM, RAG and self-reflection baselines.

2 Preliminaries

PubMed and MeSH Terms. PubMed is the primary biomedical literature database, indexing over 35 million references. Each article is annotated with Medical Subject Headings (MeSH), a controlled vocabulary that organizes biomedical concepts into a hierarchical taxonomy. Queries often combine MeSH terms with Boolean operators (AND, OR, NOT), enabling structured retrieval.

Biomedical Question Answering. The input is a natural language question Q (e.g., "*Do leukotrienes play a key role in asthma?*"), and the output is a response R that is both factually accurate and explicitly supported by authoritative literature.

Problem Setup. Given a user question Q, an optional task specification T (e.g., "*answer yes/no with a justification*"), and optional context C, the objective is to retrieve a set of relevant biomedical articles A and synthesize a final response R that is evidence-grounded and interpretable.

Search Result Assessment Metrics. To enable iterative query refinement, we adapt evaluation dimensions from search benchmarks (Gao et al., 2013; Jiang et al., 2024) to the biomedical QA setting, defining three structured feedback signals:

- **Coverage:** Does the MeSH term retrieve articles that represent the core biomedical concept?
- **Alignment:** Are the retrieved articles relevant to the original question?
- **Redundancy:** Does the MeSH term overlap with others, reducing retrieval efficiency?

These signals guide improvements to the evolving query, ensuring that downstream reasoning is grounded in a high-quality evidence pool.

3 Proposed Method

We introduce PubMed Reasoner, a multi-stage agent framework inspired by the workflow of a biomedical researcher. Unlike direct LLM-based answering or one-shot retrieval-augmented generation, PubMed Reasoner explicitly separates reasoning into three phases: **(1) Search with Self-Critic Query Refinement**, **(2) Reflective Article Retrieval with Early Stopping**, and **(3) Evidence-Grounded Response Generation**. This design ensures that the system does not rely solely on

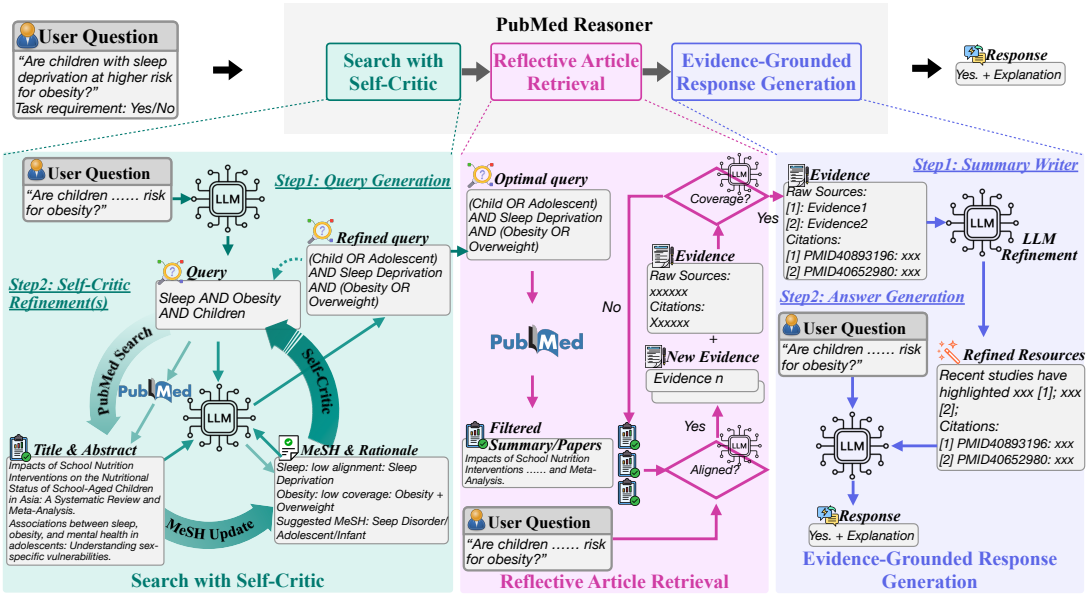

Figure 2: PubMed Reasoner stages. **(1) Search with Self-Critic Query Refinement.** From a user question, MeSH terms and a structured query are proposed. Self-critic evaluates each term for *coverage*, *alignment*, and *redundancy*, iteratively refining the query. **(2) Reflective Article Retrieval with Early Stopping.** PubMed Reasoner queries PubMed, filters results by title/abstract, and extracts supporting evidence in batches, checking whether accumulated evidence sufficiently covers the question; if so, retrieval is terminated early to save tokens and avoid unnecessary processing. **(3) Evidence-Grounded Response Generation.** Retained evidence is synthesized into final answer with explicit inline citations, ensuring factual grounding and traceability.

parametric memory, but grounds its explanation in authoritative biomedical literature.

3.1 Search with Self-Critic Query Refinement

Query Generation. The first stage constructs a structured query that serves as the initial search plan for PubMed retrieval. Given a user question Q grounded in biomedical knowledge, together with optional contextual information C, a large language model (LLM) is employed to generate a broad set of candidate Medical Subject Headings (MeSH) terms. Each candidate term is accompanied by a brief rationale explaining its relevance to the query. Formally, this process is defined as:

$$\mathcal{M}_0 \leftarrow \mathrm{LLM}_{\mathrm{mesh\_gen}}(Q, C), \tag{1}$$

where $\mathcal{M}_0 = \{(\mathrm{MeSH}_i, r_i)\}_{i=1}^{N}$, MeSH_i denotes a candidate MeSH term, and r_i provides the corresponding semantic justification for its inclusion.

From the initial candidate pool $\mathcal{M}_0$, the LLM selects a subset of MeSH terms based on multiple criteria, including the model's confidence for each term, the plausibility of its accompanying rationale, and the degree of semantic alignment between the term and the input question:

$$\mathcal{M}_0' = \mathrm{LLM}_{\mathrm{mesh\_select}}(\mathcal{M}_0, Q, C), \tag{2}$$

where $\mathcal{M}_0' \subseteq \mathcal{M}_0$ denotes the set of candidate terms with their rationales. The selected subset $\mathcal{M}_0'$ is then combined using Boolean operators to form the initial structured query:

$$q_0 \leftarrow \mathrm{LLM}_{\mathrm{query\_gen}}(\mathcal{M}_0', Q, C). \tag{3}$$

Core concepts are typically linked with AND to enforce precision, while broader synonyms or alternatives are connected with OR to maximize recall. Temporal filters (e.g., publication date restrictions) are then applied to constrain retrieval during evaluation. This formulation mirrors the workflow of human researchers, who begin with broad but structured queries and iteratively refine them based on preliminary search results.

Iterative Self-Critic Query Refinement. Once the initial query is generated, the self-critic mechanism guides iterative refinement. At each iteration t, PubMed Reasoner submits the current query q_{t-1} to the PubMed search engine, which returns a ranked list of retrieved records (e.g., title, abstract, and PubMed ID):

$$\mathcal{A}_t \leftarrow \mathrm{PubMedSearch}(q_{t-1}). \tag{4}$$

From the ranked results $\mathcal{A}_t$, we extract metadata fields that serve as inputs to the self-critic. We

denote these self-critic signals as:

$$\mathcal{S}_t = \big\{(\text{title}_i,\ \text{abstract}_i)\big\}_{i=1}^{|\mathcal{A}_t|}. \tag{5}$$

Rather than analyzing full texts which would incur substantial computational and token costs, the self-critic operates solely on $\mathcal{S}_t$. This design enables efficient evaluation of candidate MeSH terms while preserving sufficient semantic information for relevance assessment. Given the current candidate MeSH pool $\mathcal{M}'_{t-1}$, each term is then evaluated along three structured dimensions (Sec. 2), which collectively provide actionable feedback for iterative query refinement: Coverage measures whether the concept represented by a candidate MeSH_i term appears in the self-critic signals $\mathcal{S}_t$; Alignment evaluates whether the articles associated with each candidate MeSH_i term are pertinent to the user question; Redundancy identifies whether a candidate term overlaps with, or is superfluous given other terms in the current set, as well as the logical composition (AND/OR) implied by the query intent.

The evaluation of $\mathcal{M}'_{t-1}$ along these dimensions is done through the following operators:

$$\text{Cvg}_t \leftarrow \text{LLM}_{\text{coverage}}\big(\mathcal{M}'_{t-1}, \mathcal{S}_t\big), \tag{6}$$

$$\text{Align}_t \leftarrow \text{LLM}_{\text{alignment}}\big(\mathcal{M}'_{t-1}, \mathcal{S}_t, Q\big), \tag{7}$$

$$\text{Redun}_t \leftarrow \text{LLM}_{\text{redundancy}}\big(\mathcal{M}'_{t-1}, \mathcal{S}_t, q_{t-1}\big). \tag{8}$$

Each of Cvg_t, Align_t, and Redun_t is a list of pairs $\{(y_{t,i},\ r_{t,i})\}_{i=1}^{|\mathcal{M}'_{t-1}|}$, where $y_{t,i} \in \{\text{Yes}, \text{No}\}$ is a binary outcome indicating the result of the corresponding operator for candidate term MeSH_i, and $r_{t,i}$ provides a textual rationale that informs subsequent refinement. In particular, for Redun_t, $r_{t,i}$ both explains the redundancy verdict and includes the recommended boolean linkage under the previous search query q_{t-1}.

Given the previous MeSH terms, current self-critic signals $\mathcal{S}_t$, and per–term feedback, we refine the candidate MeSH set:

$$\begin{aligned}\mathcal{M}_t \ \leftarrow\ \text{LLM}_{\text{update}}\Big(\ &\mathcal{M}'_{t-1},\ Q,\ C,\\ &\mathcal{S}_t, \text{Cvg}_t,\ \text{Align}_t,\ \text{Redun}_t\Big),\end{aligned} \tag{9}$$

where $\mathcal{M}_t$ is the revised set (with rationales). The update favors terms that increase coverage without harming alignment, and prunes (or merges) terms flagged as redundant.

From the refined MeSH terms $\mathcal{M}_t$, PubMed Reasoner drafts a new query, then enforces syntactic validity with a rule-based normalizer:

$$\tilde{q}_t \ \leftarrow\ \text{LLM}_{\text{refine}}\big(\mathcal{M}_t,\ \mathcal{S}_t,\ \mathcal{H},\ Q,\ C\big), \tag{10}$$

$$q_t \ \leftarrow\ \text{QueryNormalize}\big(\tilde{q}_t\big), \tag{11}$$

where $\mathcal{H}$ stores the query history $\{q_0, ..., q_{t-1}\}$. The refinement aims to balance recall and precision using the proxies above: encourage additions that improve coverage (recall) if alignment remains high, and remove or demote terms that are misaligned or redundant (precision). The self-critic loop then repeats with q_t until a stopping rule is met (e.g., no gain in coverage/alignment or a budget limit). The final optimized query q^* is returned for downstream retrieval.

3.2 Reflective Article Retrieval with Early Stopping

PubMed Reasoner operates on the final retrieved records $\mathcal{A}^\star = \{a_1, \ldots, a_M\}$ returned by the search step (Eq. 4) together with their corresponding metadata $\mathcal{S}^*$(Eq. 5). Since PubMed search prioritizes records by relevance, highly pertinent evidence is expected to appear early in the ranked list. Accordingly, PubMed Reasoner enforces an early-stopping rule: once sufficient supporting evidence has been accumulated, retrieval halts to avoid further token consumption.

Coarse Filtering. Each retrieved record is first screened using metadata $\mathcal{S}^*$ to assess coarse relevance to the query. Only plausibly relevant records are retained for subsequent processing:

$$\big\{\big(v_i, r_i^{\text{filter}}\big)\big\}_{i=1}^{|\mathcal{A}^\star|} \leftarrow \text{LLM}_{\text{filter}}\big(\mathcal{S}^\star,\ Q\big), \tag{12}$$

where $v_i \in \{\text{Yes}, \text{No}\}$ indicates keep/drop and r_i^{filter} provides a short rationale. Since $\mathcal{A}^\star$ is ranked by PubMed, the retained set $\mathcal{A}^+ = \{\, a_i \in \mathcal{A}^\star :\ v_i = \text{Yes} \,\}$ preserves the original order and enables early prioritization of high-quality evidence. To control retrieval cost, coarse filtering is applied only to the top $M_{\max}$ ranked articles, the maximum budget of articles allowed for downstream processing.

Reflective Evidence Extraction. For each $a_i \in \mathcal{A}^+$, we extract candidate evidence and evaluate **alignment** (i.e., does the candidate evidence directly address Q?):

$$\big\{(E_i,\ \text{align}_i,\ r_i^{\text{align}})\big\}_{i=1}^{|\mathcal{A}^+|} = \text{LLM}_{\text{extract}}\big(\mathcal{A}^+, Q\big), \tag{13}$$

where E_i is the extracted passage, $\text{align}_i \in \{\text{Yes}, \text{No}\}$ indicates whether the passage directly

addresses Q, and r_i^{align} provides a rationale. The evolving evidence pool is then

$$\mathcal{E} \;=\; \{\, E_i : \text{align}_i = \text{Yes} \,\}. \tag{14}$$

Batching and Early Stopping. To reduce token costs, we process the ranked results in batches. We partition $\mathcal{A}^+$ into batches $\{\mathcal{B}_1, \ldots, \mathcal{B}_K\}$ of size m (e.g., m=5). After processing batch b, we update the evidence pool: $\mathcal{E}^{(b)} = \mathcal{E}^{(b-1)} \cup \{E_i : a_i \in \mathcal{B}_b,\ \text{align}_i = \text{Yes}\}, \quad \mathcal{E}^{(0)} = \varnothing$. A reflective module then judges whether the current evidence $\mathcal{E}^{(b)}$ provides sufficient **coverage** of the query (i.e., whether the major aspects of Q are addressed).

$$\texttt{is\_sufficient} \leftarrow \text{LLM}_{\text{CoverageCheck}}(\mathcal{E}^{(b)}, Q) \tag{15}$$

Retrieval stops early when $\texttt{is\_sufficient} = \text{Yes}$, the marginal utility of additional articles becomes negligible, or the token budget is reached.

3.3 Evidence-Grounded Response Generation

Given the final curated evidence pool $\mathcal{E}$, PubMed Reasoner composes a final response by integrating the most relevant findings into a coherent, citation-supported explanation.

Summary–of–Evidence (SoE). To convert the vetted evidence $\mathcal{E}$ into a compact and citable representation, PubMed Reasoner groups passages by article and distills key observations that directly address question Q. This produces a structured summary $\text{SoE} \leftarrow \text{LLM}_{\text{summary}}(\mathcal{E}, Q)$, ensuring that every factual claim remains linked to its supporting source. Each retained citation is preserved explicitly, promoting transparency and reproducibility.

Response Generation. Finally, the system generates the user-facing answer by conditioning on the question Q, task requirements T, optional context C, and the SoE. Formally, the final response is produced as $R \leftarrow \text{LLM}_{\text{response}}(Q, T, C, \text{SoE})$, ensuring that the explanation is grounded, verifiable, and aligned with the task specification.

This staged design turns the LLM into a reasoning orchestrator: every statement in the final answer is linked to specific citations, improving interpretability and clinical trustworthiness.

4 Experiments

We evaluate PubMed Reasoner on two biomedical QA datasets: PubMedQA (Jin et al., 2019) and MMLU Clinical Knowledge (MMLU-CK) (Wang et al., 2024). For each dataset, we report both prediction accuracy and explanation quality, and conduct ablation studies to quantify the contribution of each component of PubMed Reasoner to overall performance. PubMedQA additionally provides supporting context for each question; in our setting, we restrict retrieval to articles published within the official dataset's specified year range to ensure consistency.

4.1 Evaluation Metrics

We assess model performance along several dimensions: prediction accuracy, explanation quality, cost analysis and evidence sufficiency depth.

- **Accuracy:** Proportion of questions for which the model produces the correct answer label.
- **Explanation Quality:** We evaluate explanation quality using a shared rubric with both LLM-based and human assessment. We perform pairwise comparisons and assign five-point Likert scores on four axes:
  - **Reasoning Soundness** evaluates logical coherence. The response should be consistent and follow a clear reasoning chain.
  - **Evidence Grounding** assesses whether claims are explicitly supported by biomedical literature, reducing hallucinations.
  - **Clinical Relevance** measures how directly the response addresses the biomedical question, penalizing generic content.
  - **Trustworthiness** evaluates alignment with established biomedical knowledge and avoidance of misleading statements.
- **Cost Analysis:** To quantify the computational efficiency of each method, we measure four complementary cost indicators: (1) input token usage, (2) output token usage, (3) number of LLM API calls, and (4) number of PubMed search calls issued during retrieval.
- **Query Term Quality:** To evaluate the effectiveness of search query formulation, we measure precision and recall of the MeSH terms generated after self-critic refinement, both crucial for downstream retrieval and reasoning performance.

4.2 Baselines

We compare PubMed Reasoner with the following representative baselines:

- **LLM Baseline:** Strong LLMs (Gemini-2.5 Pro and GPT-4o) that answer questions without explicit retrieval or search planning.
- **Human Performance:** Reported human accuracy on PubMedQA, which serves as an approximate upper bound for model performance.

Table 1: Accuracy on PubMedQA and MMLU-CK test sets for PubMed Reasoner variants, strong LLMs, RAG method, self-reflection agents, and human performance. Best result per column is **bold-faced**; second best is underlined.

	Method	PubMedQA	MMLU-CK
Without retrieval	Gemini-2.5 Pro	75.64%	60.52%
	GPT4o (leaderboard)	75.20%	60.64%
	Human performance	78.00%	–
RAG method	with Gemini	72.30%	58.94%
	with GPT	73.28%	58.26%
Self-reflection	with Gemini	77.08%	60.67%
	with GPT	77.12%	60.79%
PubMed Reasoner	with Gemini	77.26%	61.36%
	with GPT	**78.32%**	**63.21%**

- **RAG Method:** An LLM-based retrieval-augmented generation baseline that generates a single search query, retrieves relevant articles directly from the PubMed search engine, and incorporates them into response generation.
- **Self-reflection Agent:** A reasoning-based baseline that combines PubMed retrieval with self-reflection to iteratively refine the final answer. It generates candidate responses and improves them through an explicit reflection step.

For fair comparison, both the RAG and self-reflection baselines share the same query generation, article retrieval, summarization, and question-answering prompts as PubMed Reasoner, with the self-reflection agent using one additional prompt for the reflection stage.

4.3 PubMedQA

Accuracy. Table 1 shows that PubMed Reasoner achieves near-human or superior accuracy on PubMedQA. In particular, PubMed Reasoner (GPT) attains 78.32% accuracy, slightly exceeding the reported human expert performance. Compared to other baselines, PubMed Reasoner consistently outperforms both direct LLM inference and RAG-based methods, improving upon the GPT baseline by 3.12% and yielding clear gains over RAG. Notably, standard RAG underperforms direct LLM inference, likely due to noisy retrieval caused by unrefined, one-shot query generation that introduces weakly aligned or irrelevant evidence into the model context. Relative to the self-reflection agent, PubMed Reasoner achieves modest but consistent accuracy improvements. Importantly, these gains are obtained with substantially lower computational cost (shown in Table 3). Together, these results demonstrate that PubMed Reasoner offers a more favorable accuracy–efficiency trade-off than existing baselines, delivering practical accuracy improvements while significantly reducing computational overhead.

Explanation Quality. On PubMedQA, PubMed Reasoner with the Gemini backbone consistently outperforms Gemini-2.5 Pro across all four LLM-judge dimensions, as reported in Table 2. The pairwise win rate rises by 14.9% in Reasoning Soundness, by 14.6% in Evidence Grounding, by 14.2% in Clinical Relevance, and by 17.2% in Trustworthiness, while tie rates remain low. The average Likert score also increases respectively by 0.168, 0.180, 0.146, and 0.163, yielding explanations that are more coherent, better grounded in evidence, more clinically focused, and more trustworthy.

Cost Analysis. As shown in Table 3, PubMed Reasoner achieves substantial reductions in computational overhead compared to the self-reflection baseline. Across key cost metrics, PubMed Reasoner reduces input token usage by 55.34%, lowers the number of LLM API calls by 41.82%, and decreases PubMed search calls by 41.36%. These efficiency gains stem from structured query planning and early stopping, which avoid unnecessary retrieval and redundant reasoning steps. Although the overall cost of PubMed Reasoner remains higher than that of the one-shot LLM baseline, it produces citation-grounded and verifiable responses, offering substantially stronger reliability than direct generation. Compared to the RAG baseline, which often yields shallow or weakly aligned search results, PubMed Reasoner performs more targeted retrieval and accumulates higher-quality evidence.

Query Term Quality. As shown in Table 4, PubMed Reasoner maintains high precision while substantially improving recall due to self-critic refinement during query planning. With a GPT backbone, recall improves by 20.99% over the GPT baseline and by 7.6% over the self-reflection agent. With a Gemini backbone, recall improves by 17.18% over the baseline and also surpasses the self-reflection agent. These results demonstrate that self-critic refinement broadens conceptual coverage without sacrificing alignment, producing higher-quality search queries.

4.4 MMLU Clinical Knowledge

Accuracy. On MMLU-CK dataset (Table 1), self-reflection offers only a marginal lift over the raw LLM baseline. In contrast, PubMed Reasoner delivers consistent gains over the respective back-

Table 2: Explanation quality evaluation on PubMedQA (left) and MMLU-CK (right) test sets: Gemini w/o retrieval vs. PubMed Reasoner. Win/tie/loss rates from pairwise comparisons judged by GPT-4; average Likert scores (1–5).

Metric	PubMedQA					MMLU-CK				
	Loss / Tie / Win (%)			Avg. Likert (1–5)		Loss / Tie / Win (%)			Avg. Likert (1–5)	
	Gemini	Tie	Ours	Gemini	Ours	Gemini	Tie	Ours	Gemini	Ours
Reasoning Soundness	39.7	5.7	**54.6**	3.416	**3.584**	44.0	10.8	**45.2**	3.307	**3.699**
Evidence Grounding	40.8	3.8	**55.4**	3.421	**3.601**	25.2	11.7	**64.1**	3.209	**3.595**
Clinical Relevance	39.2	7.4	**53.4**	3.438	**3.584**	34.4	20.0	**45.6**	3.525	**3.732**
Trustworthiness	38.7	5.4	**55.9**	3.424	**3.587**	35.3	15.9	**48.8**	3.386	**3.712**

Table 3: Cost comparison on PubMedQA and MMLU-CK using a shared GPT-4 backbone for PubMed Reasoner and baselines. For PubMedQA, input and output token counts are reported as ratios relative to the direct LLM baseline. For each metric, the lower value between the self-reflection agent and PubMed Reasoner is **bolded**.

	PubMedQA				MMLU-CK			
Method	**Input Tokens**	**Output Tokens**	**LLM Calls**	**Search Calls**	**Input Tokens**	**Output Tokens**	**LLM Calls**	**Search Calls**
GPT-4o	542.70	144.90	1	0	90.24	98.64	1	0
RAG method	×1.64	×2.40	2	1	×4.00	×3.06	2	1
Self-reflection	×225.27	×13.60	13.08	3.52	×1914.53	×**24.35**	14.61	4.77
PubMed Reasoner	×**97.25**	×**12.67**	**7.61**	**2.49**	×**1098.26**	×24.86	**10.72**	**3.59**

Table 4: Search quality on PubMedQA. Precision/recall computed by comparing set of predicted MeSH terms from the final query against gold MeSH annotations.

Method	Precision	Recall
Gemini-2.5 Pro	0.9422	0.6848
+Self-reflection	0.9745	0.7900
+PubMed Reasoner	**0.9826**	**0.8025**
GPT-4o	**0.9874**	0.7052
+Self-reflection	0.9562	0.7928
+PubMed Reasoner	0.9868	**0.8532**

bones: 1.11% over Gemini-2.5 Pro and 2.69% over GPT-4o. This underscores that our self-critic improves accuracy beyond what self-reflection can achieve. Moreover, as shown in Table 3, these accuracy gains are achieved with substantially lower computational overhead.

Explanation Quality. On MMLU-CK, PubMed Reasoner consistently surpasses Gemini-2.5 Pro across all LLM-judge dimensions (Table 2). Pairwise win rates rise by 1.2% in reasoning soundness, 38.9% in evidence grounding, 11.2% in clinical relevance, and 13.5% in trustworthiness, with low tie rates. Average Likert scores also increase by 6–12% across dimensions, indicating clearer logic, stronger evidence support, sharper clinical focus, and greater trustworthiness. In sum, PubMed Reasoner produces better clinical explanations.

Cost Analysis. As shown in Table 3, the MMLU-CK results exhibit a pattern similar to PubMedQA: PubMed Reasoner consistently incurs substantially lower computational cost than the self-reflection baseline. In particular, PubMed Reasoner requires fewer input tokens, fewer LLM API calls, and fewer PubMed search calls, reflecting a more efficient retrieval and reasoning workflow on this broader clinical knowledge benchmark as well.

Query Term Quality. MMLU does not provide MeSH term annotations, hence cannot be assessed.

4.5 Ablation Study

Ablation Setup. We ablate four components:**(1) LLM-human preference agreement**, **(2) self-critic refinement**, **(3) reflective evidence extraction** and **(4) batching and early stopping** using the PubMed Reasoner (Gemini) variant

LLM-human preference agreement. To assess consistency between LLM- judge and human evaluation, we additionally report agreement between LLM judgments and human judgments. We evaluate explanation quality using LLM-based rubric scoring together with human preference validation. We conduct a human preference study on a random sample of 39 instances. For each sampled question, the compared system outputs are anonymized and presented in randomized order to three human experts, who indicate which response they prefer. We then report majority human preference and compare it against the LLM judge's pairwise preference to assess whether the automatic evaluation is directionally consistent with expert judgment. After binarizing pairwise preferences into *Win* versus

Table 5: Ablation on PubMedQA with PubMed Reasoner (Gemini): impact of self-critic module. *Evidence-Grounded Response Rate (EGR)*: fraction of questions whose final response cites ≥ 1 extracted findings that pass coverage and alignment checks in reflective stage.

Method / Config.	Accuracy	EGR
PubMed Reasoner	77.26%	82.64%
w/o self-critic	75.60%	64.46%

Non-win (i.e., merging loss and tie), the agreement between the LLM judge and the human evaluators yields a Cohen's κ of 0.448, indicating moderate alignment.

Self-Critic Refinement. Table 5 reports accuracy and the Evidence-Grounded Response Rate (EGR)—the fraction of questions whose final response cites at least one extracted finding that passes coverage and alignment checks. Removing the self-critic markedly lowers EGR (drop of 28.20%) and also reduces accuracy 2.2%. This confirms that the self-critic is not only improving final correctness, but also producing robust, well-formed queries that retrieve on-point evidence.

Reflective Evidence Extraction. We disable the alignment filter and instead summarize all retrieved articles before producing a final answer. As shown in Table 7, accuracy remains unchanged, but the quality of explanations shifts: reflective integration improves reasoning soundness and clinical relevance, while introducing more diverse evidence slightly lowers grounding precision.

Batching and Early Stopping. We further study the effect of the early-stopping mechanism on the PubMedQA Gemini variant. In the full setting, the model reads up to 20 retrieved papers in batches of five and stops once the accumulated evidence is sufficient to answer the question. To evaluate whether this mechanism is necessary, we replace it with two reduced-context alternatives, motivated by context-length constraints that prevent supplying all 20 full papers at once: **(1) Abstract Only**, which uses only the abstract/metadata of each paper, and **(2) Paper Summary**, which concatenates per-paper summaries before answer generation. As shown in Table 6, both variants perform substantially worse than PubMed Reasoner, indicating that high-level condensations fail to preserve essential reasoning cues. These results highlight the importance of presenting evidence at an appropriate granularity, which our selective early-stopping framework better maintains while remaining context-efficient.

Table 7: Ablation on PubMedQA with PubMed Reasoner (Gemini): impact of reflective retrieval (RR).

Metric	*w/o* RR	Ours
Accuracy (%)	77.24	**77.26**
Reasoning Soundness	3.422	**3.554**
Evidence Grounding	**3.554**	3.427
Clinical Relevance	3.443	**3.573**
Trustworthiness	**3.432**	**3.432**

Method	**Accuracy (%)**
Abstract Only	67.20
Paper Summary	64.12
PubMed Reasoner	**77.26**

Table 6: Ablation study comparing reduced-context variants to our selective early-stopping framework on PubMedQA.

5 Related Work

LLMs & Retrieval-Augmented Methods. Despite the remarkable progress achieved by LLMs in natural language understanding, reasoning, and generation, when applied to high-stakes biomedical tasks, they often hallucinate facts or rely on outdated parametric memory, raising concerns about reliability and safety (Guan et al., 2023; Xu et al., 2024). To address these issues, RAG has emerged as a promising paradigm. Existing RAG approaches either incorporate structured resources such as knowledge graphs (Abu-Rasheed et al., 2024) or augment prompts with retrieved content from domain-specific corpora (Arslan et al., 2024). While these strategies improve factual grounding, they face persistent challenges: retrieval systems often struggle with the *coverage–relevance trade-off*, returning either too little evidence or overwhelming the model with irrelevant content (Liu et al., 2024). Moreover, once an initial query is issued, most RAG pipelines lack mechanisms for iterative refinement, making them brittle in complex biomedical scenarios (Dai et al., 2024).

Search Query Optimization. A complementary direction focuses on enhancing the query quality. Early work in information retrieval explored query expansion and relevance feedback (Sparck Jones, 1974; Crane and Bernier, 1951), but such methods often relied on manual heuristics (Arasu et al., 2001) and lacked semantic explanation capabilities (Boytsov, 2011). More recently, query optimization has also been framed as a reinforcement learning problem, where the model learns to improve retrieval performance through policy gradients or

preference-based objectives such as PPO (Schulman et al., 2017), DPO (Rafailov et al., 2023), or GRPO (Shao et al., 2024). While effective, these methods typically require a separate reward model or explicit training signals, making them computationally expensive and less flexible in specialized domains like biomedicine. In contrast, our work introduces a training-free self-critic mechanism that performs query refinement without external supervision or gradient updates. Unlike prior RL-based or heuristic based method, the self-critic provides fine-grained feedback on MeSH query terms, iteratively improves retrieval quality, mitigates error propagation and enhances factual grounding.

Self-Reflection and Reasoning Agents. Another line of research seeks to improve LLM reliability through self-reflection and agent-based reasoning. Self-reflection methods allow models to reexamine their own outputs and refine answers, while reward modeling (Leike et al., 2018; Choudhury, 2025) and verbal reinforcement learning (Shinn et al., 2023) aim to align reasoning with human-like preferences. Self-consistency sampling further increases robustness by aggregating multiple reasoning trajectories (Wang et al., 2022). However, these methods generally intervene at the *answer-generation stage*, which makes them computationally expensive and unable to prevent low-quality retrieval from propagating downstream. As a result, they provide limited control over the evidence collection process itself.

6 Conclusion

Altogether, these results demonstrate that combining structured self-critique with evidence-based integration moves biomedical QA closer to expert-level explanation, while remaining efficient and reproducible. More broadly, our findings suggest design principles for multi-stage LLM agents in high-stakes domains: shifting reflection earlier in the pipeline can prevent compounding errors; explicit grounding in external evidence improves transparency and reliability; and adaptive mechanisms such as early stopping enable practical deployment without sacrificing rigor. Notably, self-critic is especially effective in multi-step reasoning settings, where revising only problematic steps rather than regenerating the entire chain ensures both efficiency and logical consistency. These mechanisms can be generalized beyond biomedicine, providing a blueprint for building trustworthy, domain-specialized LLM systems in areas such as law, finance, and scientific discovery.

7 Ethical Considerations

Our work studies biomedical question answering in a high-stakes domain, where incorrect or unsupported outputs may lead to misleading conclusions if used without expert oversight. Although our framework improves evidence-grounded reasoning, it should be viewed as a research tool rather than a substitute for clinical expertise. In addition, LLM-as-a-Judge evaluations may not fully align with human judgments, and retrieval may miss relevant evidence, especially for specialized or long-tail topics. To better understand these risks, we incorporate human annotation to examine discrepancies between automated and human evaluation.

8 Impact Statement

Our work seeks to improve biomedical question answering through retrieval-enhanced and evidence-grounded reasoning, with potential benefits for scalable evaluation and knowledge-intensive research support. By encouraging models to rely on external evidence rather than only internal parametric knowledge, the framework may improve transparency and answer reliability.

However, the approach also carries limitations and risks. The self-critic mechanism is heuristic and may inherit biases from the underlying LLM, especially in underrepresented biomedical subdomains. The retrieval process may also miss lower-ranked but relevant articles, potentially limiting evidence coverage for long-tail questions. Moreover, because the pipeline is largely linear, errors made in earlier stages may propagate downstream without effective correction.

For these reasons, the method should be used with caution and should not replace expert human judgment in high-stakes biomedical applications. We view it as a supportive research tool rather than a fully autonomous decision-making system.

9 Acknowledgments

We sincerely thank Yiting Lin and Zhuhao Zhang from Zhejiang University School of Medicine, and Christine Yao from University of Massachusetts T.H. Chan School of Medicine for their valuable assistance with the human annotation task. Their efforts were important for comparing and verifying the alignment between LLM-as-a-Judge evaluations and human judgments.

References

Hasan Abu-Rasheed, Christian Weber, and Madjid Fathi. 2024. Knowledge graphs as context sources for llm-based explanations of learning recommendations. In *2024 IEEE Global Engineering Education Conference (EDUCON)*, pages 1–5. IEEE.

Arvind Arasu, Junghoo Cho, Hector Garcia-Molina, Andreas Paepcke, and Sriram Raghavan. 2001. Searching the web. *ACM Transactions on Internet Technology (TOIT)*, 1(1):2–43.

Muhammad Arslan, Hussam Ghanem, Saba Munawar, and Christophe Cruz. 2024. A survey on rag with llms. *Procedia computer science*, 246:3781–3790.

Leonid Boytsov. 2011. Indexing methods for approximate dictionary searching: Comparative analysis. *Journal of Experimental Algorithmics (JEA)*, 16:1–1.

Sanjiban Choudhury. 2025. Process reward models for llm agents: Practical framework and directions. *arXiv preprint arXiv:2502.10325*.

EJ Crane and Charles L Bernier. 1951. Indexing and index-searching. *Punched Cards: Their Applications to Science and Industry*, page 331.

Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. 2024. Bias and unfairness in information retrieval systems: New challenges in the llm era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6437–6447.

Wanling Gao, Yuqing Zhu, Zhen Jia, Chunjie Luo, Lei Wang, Zhiguo Li, Jianfeng Zhan, Yong Qi, Yongqiang He, Shiming Gong, and 1 others. 2013. Bigdatabench: a big data benchmark suite from web search engines. *arXiv preprint arXiv:1307.0320*.

Jian Guan, Jesse Dodge, David Wadden, Minlie Huang, and Hao Peng. 2023. Language models hallucinate, but may excel at fact verification. *arXiv preprint arXiv:2310.14564*.

Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanmin Wu, Jiayi Lei, Pengshuo Qiu, Pan Lu, Zehui Chen, Chaoyou Fu, Guanglu Song, and 1 others. 2024. Mmsearch: Benchmarking the potential of large models as multi-modal search engines. *arXiv preprint arXiv:2409.12959*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W Cohen, and Xinghua Lu. 2019. Pubmedqa: A dataset for biomedical research question answering. *arXiv preprint arXiv:1909.06146*.

Adam Tauman Kalai, Ofir Nachum, Santosh S Vempala, and Edwin Zhang. 2025. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*.

Jan Leike, David Krueger, Tom Everitt, Miljan Martic, Vishal Maini, and Shane Legg. 2018. Scalable agent alignment via reward modeling: a research direction. *arXiv preprint arXiv:1811.07871*.

Zheng Liu, Yujia Zhou, Yutao Zhu, Jianxun Lian, Chaozhuo Li, Zhicheng Dou, Defu Lian, and Jian-Yun Nie. 2024. Information retrieval meets large language models. In *Companion Proceedings of the ACM Web Conference 2024*, pages 1586–1589.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023a. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*.

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, and 1 others. 2023b. Can generalist foundation models outcompete special-purpose tuning? case study in medicine. *arXiv preprint arXiv:2311.16452*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Mohamed Amin, Le Hou, Kevin Clark, Stephen R Pfohl, Heather Cole-Lewis, and 1 others. 2025. Toward expert-level medical question answering with large language models. *Nature Medicine*, 31(3):943–950.

Karen Sparck Jones. 1974. Automatic indexing. *Journal of documentation*, 30(4):393–432.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, and 1 others. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.