HYPER: Hypothesis-Guided Planning for Evidence-Seeking Retrieval in Medical Question Answering

Anonymous ACL submission

Abstract

Retrieval-Augmented Generation (RAG) improves Large Language Models (LLMs) by grounding generation in external, non-parametric knowledge. However, when a task requires choosing among competing possibilities, topically relevant context is often insufficient to drive the final decision. Existing RAG methods typically rely on a single initial query, which tends to favor topical relevance over decision-relevant evidence and can retrieve background information that fails to discriminate among plausible answers. To address this issue, we propose **HYPER**: **Hy**pothesis-Guided **P**lanning for **E**vidence-Seeking **R**etrieval, a training-free pre-retrieval planning framework that reorients RAG from topic-oriented retrieval to evidence-oriented retrieval. HYPER uses a provisional answer hypothesis as a planning anchor. It then decomposes retrieval into three evidence roles. SUPPORT seeks evidence for the provisional answer. DISTINCTION retrieves criteria that separate competing alternatives. FEATURE grounds salient question features needed for the decision. This planning step steers retrieval toward evidence near the decision boundary, enabling the generator to answer from context that is more directly aligned with the required choice. Experiments on MedQA and MMLU-Med show that HYPER consistently outperforms single-query RAG and rerank/filter baselines, improving average accuracy over Simple RAG by 5.9 and 3.6 points, respectively. Code is available at `https://anonymous.4open.science/r/HYPER-1047`.

1 Introduction

Retrieval-Augmented Generation (RAG) grounds language-model generation in external knowledge and is now a standard component of many question answering systems (Lewis et al., 2020). However, retrieval is not uniformly helpful. Retrieved documents can contain irrelevant or noisy information that harms downstream generation (Amiraz et al., 2025; Yoran et al., 2023; Du et al., 2025). This issue is especially visible in medical multiple-choice QA, where strong models can often answer from parametric knowledge alone, while simple single-query retrieval can reduce accuracy relative to no-retrieval chain-of-thought prompting.

The core difficulty is not only whether to retrieve, but what to retrieve. In decision-oriented tasks, topically related passages may still be uninformative for choosing among competing answers. Under a bounded context budget, such passages displace evidence that could narrow the decision. We therefore focus on retrieval planning for decision-useful evidence: evidence that supports one answer, distinguishes it from plausible alternatives, or grounds salient clues in the question. This framing follows the long-standing distinction between topical relevance and utility in IR, and recent RAG work that evaluates retrieved context by downstream usefulness (Cooper, 1971; Salemi and Zamani, 2024; Joren et al., 2025; Zhang et al., 2024b,a).

This motivates a pre-retrieval planning problem: the system should decide what evidence would help resolve the answer choice before the candidate pool is formed, rather than relying only on a generic first pass followed by post-hoc selection.

To address this planning problem, we introduce **HYPER**: **Hy**pothesis-Guided **P**lanning for **E**vidence-Seeking **R**etrieval, a training-free pre-retrieval planning method. HYPER uses a provisional answer hypothesis not as a final prediction, but as a planning anchor for retrieval. From the question and candidate answers, it constructs a compact retrieval-planning state and converts it into three targeted queries: SUPPORT for evidence supporting the provisional answer, DISTINCTION for criteria separating competing alternatives, and FEATURE for evidence grounding salient question features. The provisional hypothesis steers retrieval

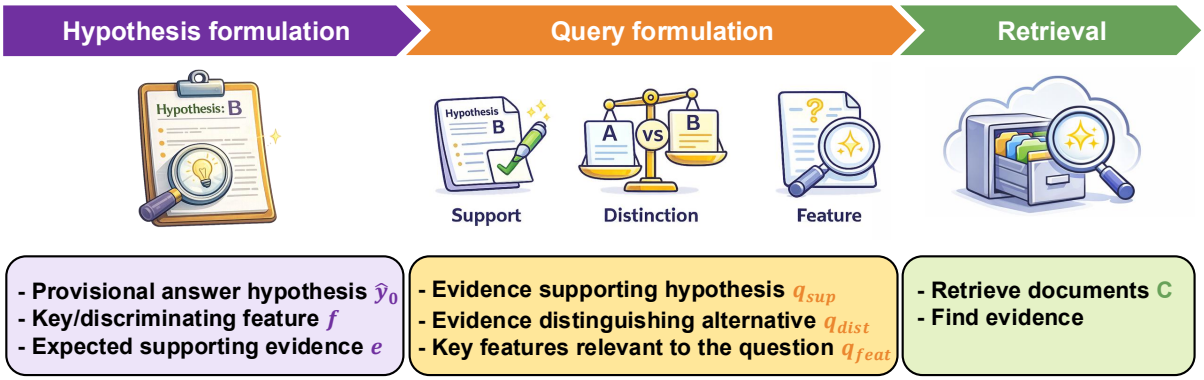

Figure 1: **HYPER pipeline.** A hypothesis formulator proposes a lightweight provisional answer hypothesis together with the evidence expected under that hypothesis. The query formulator turns this state into three verification-oriented queries. The provisional answer hypothesis steers retrieval but is not shown to the generator by default.

but is not shown to the final generator by default.

We evaluate HYPER on MedQA and MMLU-Med, where questions often hinge on diagnostic criteria, differential comparisons, treatment indications, or question-specific clinical clues. Under the same final-context budget, HYPER achieves the best average performance across both datasets and outperforms strong retrieval baselines in nearly all dataset–model settings. It also increases the fraction of retrieved context sets judged useful for the downstream decision.

Our contributions are:

1. We introduce **HYPER**, a training-free pre-retrieval planning method that uses a provisional answer hypothesis to construct three complementary evidence-seeking queries.
2. We analyze retrieved context with the labels ENTAILED, USEFUL-BUT-NOT-ENTAILED, and NOT USEFUL, measuring whether retrieval provides evidence useful for the downstream decision.
3. We show that hypothesis-guided retrieval planning improves medical multiple-choice QA under a shared final-context budget, with gains in both answer accuracy and context utility.

2 Related Work

RAG design and harmful context. RAG systems are commonly organized into pre-retrieval, retrieval, post-retrieval, and generation stages (Fan et al., 2024; Huang and Huang, 2024). HYPER intervenes at the pre-retrieval stage. This focus is motivated by evidence that additional context is not always beneficial: irrelevant passages can distract the model, weakly aligned context reduces robustness, and longer contexts can impair reasoning even when some relevant evidence is present (Amiraz et al., 2025; Yoran et al., 2023; Du et al., 2025). These findings suggest that retrieval should be optimized not only for topical match, but also for whether the retrieved context helps the downstream decision.

Query transformation and multi-query retrieval. A direct way to improve retrieval is to transform the input query before search. Training-based methods learn query rewriters or retrieval-aware generators using supervised data, downstream reader feedback, or process-level rewards (Ma et al., 2023; Chan et al., 2024; Ye et al., 2025; Wang et al., 2025). Training-free methods instead use prompting to rewrite, expand, decompose, or diversify queries. Examples include hypothetical-document retrieval, paraphrase-style multi-query retrieval, and information-diverse rewriting (Gao et al., 2023; Rackauckas, 2024; Li et al., 2024). These methods improve recall, semantic match, or coverage. HYPER also generates multiple queries, but its queries are not generic paraphrases or information-level variants. They are planned around three evidence roles: supporting a provisional answer, distinguishing alternatives, and verifying salient question features.

Reasoning-guided and agentic retrieval. Reasoning-guided RAG methods treat retrieval as part of an adaptive reasoning process. IRCoT interleaves chain-of-thought reasoning with retrieval for multi-hop questions (Trivedi et al., 2023); Self-RAG trains a model to decide when to retrieve and to critique retrieved evidence through self-reflection (Asai et al., 2024); and broader

agentic RAG systems use planning or tool-use policies to decide when and how to retrieve (Liang et al., 2025). These methods mainly study adaptive retrieval control across one or more reasoning steps. HYPER studies a complementary pre-retrieval planning problem: before the first retrieval pass, how can a provisional reasoning state be converted into targeted evidence-seeking queries while keeping the retriever and final generator fixed?

Retrieval utility. Information retrieval has long distinguished topical relevance from utility: a result may be about a query without being useful for satisfying the user's information need (Cooper, 1971). Recent RAG work revisits this distinction by evaluating retrieved context in terms of sufficiency or downstream usefulness rather than surface-level relevance alone (Salemi and Zamani, 2024; Joren et al., 2025; Zhang et al., 2024b,a). HYPER follows this utility-oriented view, focusing on whether retrieved context helps resolve the downstream decision rather than merely matching the question topic.

Post-retrieval selection. Post-retrieval methods assume that a candidate pool has already been retrieved and then decide which passages should enter the final generator context. Some methods use retrieval-side selection criteria: MMR balances query relevance with novelty to reduce redundancy, while reciprocal rank fusion aggregates multiple ranked lists (Carbonell and Goldstein, 1998; Cormack et al., 2009). Others incorporate model feedback: RePLUG, RankRAG, CRAG, MAIN-RAG, and ITEM methods use reranking, correction, filtering, or utility judgments to select evidence before generation (Shi et al., 2024; Yu et al., 2024; Yan et al., 2024; Chang et al., 2025; Zhang et al., 2024a). HYPER intervenes earlier: rather than improving an already retrieved candidate pool, it constructs evidence-seeking queries before that pool is formed.

Medical RAG. Medical QA is a useful setting for studying retrieval quality because correct answers often depend on clinical criteria, differential diagnosis, treatment indications, or a small number of question-specific clues. Prior work shows that medical RAG performance is sensitive to corpus choice, retriever design, and retrieval strategy, and that reformulation or iterative follow-up retrieval can improve performance (Xiong et al., 2024a,b; Kim et al., 2025). We use medical multiple-choice QA because it makes the gap between topical relevance and decision usefulness especially visible, although the HYPER formulation is not limited to medicine.

3 Method

3.1 Problem Setting

We consider multiple-choice decision settings where a model selects an answer from candidates $A = \{a_1, \ldots, a_K\}$ and may use a retrieved context set $C_m(x, A) \subseteq D$. We compare methods under a shared final-context budget,

$$|C_m(x, A)| \leq B_{\max},$$

which limits the evidence shown to the final generator. The budget is not an exact-fill requirement: methods may inspect different intermediate candidate pools, and the realized final context may contain fewer than $B_{\max}$ documents because of deduplication or filtering.

In this setting, retrieval is useful only when the final context helps resolve the answer choice. HYPER therefore treats retrieval as a pre-retrieval planning problem: before forming the candidate pool, the system asks what evidence would support a provisional answer hypothesis, distinguish it from alternatives, or ground salient question features.

3.2 HYPER: Overview

HYPER is a training-free, hypothesis-guided retrieval planning framework for question answering under a bounded final-context budget. Its central idea is to use a tentative answer not as a final prediction, but as a control signal for retrieval: the system first forms a provisional answer hypothesis, then infers what evidence should hold if that hypothesis were correct, and finally retrieves documents that test those expectations.

The framework has three components:

1. **Hypothesis formulator**, which produces a compact retrieval-planning state;
2. **Query formulator**, which converts that hypothesis into three targeted retrieval queries;
3. **Retriever and fusion**, which gather evidence for those queries and merge it into a final context under the shared budget;

More specifically, the hypothesis formulator H first constructs the retrieval-planning state

$$h = H(x, A) = (\hat{y}_0, f, e, r),$$

where $\hat{y}_0$ is the provisional answer hypothesis, f denotes discriminating features, e denotes evidence needs, and r denotes brief reasoning. Conditioned on the original question x and this state h, the query formulator Q produces three queries,

$$(q_{\text{sup}}, q_{\text{dist}}, q_{\text{feat}}) = Q(x, h).$$

The shared retriever R retrieves evidence for these queries, and the retrieved results are fused into a final context C,

$$C = \text{Fuse}\big(R(q_{\text{sup}}), R(q_{\text{dist}}), R(q_{\text{feat}})\big).$$

Based on this context, the downstream generator G predicts the final answer as

$$\hat{y} = G(x, A, C). \tag{1}$$

Crucially, the hypothesis influences retrieval only through the rewritten queries and is not passed directly to the generator in the default setup. Figure 1 provides a visual overview of the main steps and the associated notation.

3.3 Stage 1: Hypothesis Formulator

The first stage is a hypothesis formulator. Given the question stem and answer options, it proposes a provisional answer hypothesis and identifies the evidence needed to verify it. The purpose of this stage is not to commit to a final answer, but to define a concrete verification target for retrieval. The hypothesis formulator outputs a compact structured record containing:

- **provisional answer hypothesis** $\hat{y}_0$: the current best-guess option;
- **discriminating features** f : question-specific clues that separate the leading candidates;
- **evidence needs** e: facts that the hypothesis formulator expects would support the provisional answer and should be sought through retrieval;
- **brief reasoning** r: a concise explanation for why this hypothesis is currently preferred.

We keep this state short and structured because HYPER uses it to steer retrieval rather than to expose a long reasoning trace to the generator. What matters is the verification agenda implied by the tentative answer.

3.4 Stage 2: Query Formulator

Conditioned on the original question, the answer options, and the structured retrieval-planning state $(\hat{y}_0, f, e, r)$, the query formulator generates exactly three high-precision search queries, which we now name as SUPPORT, DISTINCTION, FEATURE:

- **SUPPORT** q_{sup} **: supports the provisional answer hypothesis.** This query uses the hypothesized answer text, evidence needs and brief reasoning to retrieve passages that would directly support the current hypothesis.
- **DISTINCTION** q_{dist} **: distinguishes leading alternatives.** This query uses the discriminating features and candidate answers to retrieve criteria that separate the strongest competing options.
- **FEATURE** q_{feat}: grounds salient question features. This query targets key clues in the question stem and retrieves evidence about their clinical meaning or decision relevance, independent of whether the current hypothesis is correct.

Together, these queries instantiate complementary retrieval roles. SUPPORT seeks evidence for the provisional hypothesis, DISTINCTION seeks criteria that distinguish competing answers, and FEATURE grounds question-specific clues in external evidence. This role separation is intended to make retrieval useful even when the provisional hypothesis is incomplete or partially wrong.

3.5 Retriever, Fusion, and Generation

The three rewritten queries are passed to a shared retriever, which retrieves the top-5 documents for each. We fuse these lists by deduplicating the documents and summing their retrieval scores across the queries, meaning documents retrieved multiple times receive a higher aggregate score. The final context C is then formed by ranking the documents based on this score. This fused context serves as the only information shown to the generator, which predicts the final answer as defined in Eq. (1). Appendix A details an alternative query-order fusion strategy that appends documents sequentially by query and removes later duplicates.

4 Experimental Setup

4.1 Benchmarks

We focus on medical multiple-choice QA because it provides a clean setting for studying decision-oriented retrieval: many questions are answerable from parametric knowledge, but retrieval can help when it surfaces diagnostic criteria, differential comparisons, or question-specific clinical clues. Within this domain, we evaluate HYPER on two

standard benchmarks, **MedQA** (Jin et al., 2021) and **MMLU-Med** (Hendrycks et al., 2020), which provide controlled multiple-choice supervision for retrieval-augmented decision making.

4.2 Models

The main experiments use four generator configurations: Llama3.2-3B-instruct, Llama3.1-8B-instruct, Qwen3-4B-instruct-2507, and Qwen3-30B-A3B-instruct-2507. For each method, all LLM-based components use the same model.

4.3 Compared Methods

We compare HYPER with baselines from three families: no-retrieval or simple retrieval, post-retrieval selection, and query transformation. Detailed implementation choices, prompts, and hyperparameter settings are provided in Appendix B.

No retrieval and simple retrieval. CoT answers directly without retrieval, while SIMPLE RAG retrieves the top-15 documents using the raw question as a single query and passes them to the generator.

Post-retrieval selection. These methods first retrieve candidates with the raw question and then select, filter, or reorder the final context. MMR-RAG retrieves 150 candidates and applies maximal marginal relevance as a diversity-aware selection criterion. RERANK-RAG uses the same 150-candidate pool and reranks it with a BGE-M3 reranker. MAIN-RAG starts from the top-15 retrieved documents and filters them using LLM evaluations of question-document-answer triples (Chang et al., 2025). ITEM-A also starts from the top-15 retrieved documents and iteratively alternates pseudo-answer generation with LLM-based utility judgments to select useful passages (Zhang et al., 2024a). Because MAIN-RAG and ITEM-A filter the initial context, their final context may contain fewer than 15 documents.

Query transformation. These methods rewrite or expand the query before retrieval. RRR applies the training-free Rewrite-Retrieve-Read pipeline, rewriting the input query before retrieval (Ma et al., 2023). LC-MQR is a LangChain-style MultiQueryRetriever baseline that generates paraphrase-style query variants and fuses their retrieved results; we follow the LC-MQR baseline setting used in prior work (Kuo et al., 2025). DMQR generates diverse rewrites using information-level strategies, including general rewriting, keyword rewriting, pseudo-answer rewriting, and core-content extraction (Li et al., 2024). HYDE generates hypothetical documents, encodes them with the retriever, averages their embeddings, and retrieves with the averaged query representation (Gao et al., 2023). For LC-MQR and DMQR, we use the same retrieval-fusion procedure as HYPER, so the final context may contain fewer than 15 unique documents after deduplication.

4.4 Retriever, Corpus, and Budget

All retrieval-based methods use the same retriever and corpus so that performance differences reflect retrieval strategy rather than backend changes. We use MedCPT (Jin et al., 2023) over the medical textbook corpus of Xiong et al. (2024a). The maximum final-context budget is set to $B_{\max} = 15$ documents.

We normalize comparisons by the evidence shown to the final generator, rather than by the number of intermediate candidates inspected. Methods therefore retain their own candidate-pool rules: single-query methods retrieve up to 15 documents directly; multi-query methods retrieve top-5 documents per query and fuse them with deduplication; and post-retrieval selection methods may inspect larger or method-specific candidate pools before producing the final context. Consequently, the realized final context can contain fewer than 15 documents when fusion or filtering removes candidates. Detailed candidate-pool choices for each baseline are provided in Appendix B.

4.5 Prompting and Implementation Details

For final answer generation, all methods use the same MIRAGE prompt (Xiong et al., 2024a). This keeps the generation interface fixed across methods, so differences in performance are attributable primarily to the retrieval strategy. Prompt templates and other implementation details are provided in the appendix.

4.6 Context-Utility Evaluation

Accuracy alone does not distinguish improvements due to better retrieved evidence from improvements caused by other changes in generation behavior. We therefore evaluate retrieved context sets directly using a separate LLM judge. For each question, method, and model setting, the judge first predicts the answer option most strongly supported by the retrieved context alone (`selected_answer`) and

Table 1: Accuracy (%) on MedQA and MMLU-Med for HYPER and baseline methods under the shared 15-document final-context budget. HYPER achieves the best average performance on both datasets.

	MedQA					MMLU-Med				
Method	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average
CoT	58.0	69.0	72.8	84.1	71.0	66.3	75.0	82.9	90.2	78.6
Simple RAG	55.6	66.8	71.3	82.7	69.1	64.5	74.7	81.8	88.9	77.5
Post-retrieval selection										
MMR-RAG	54.4	67.0	71.8	82.8	69.0	**69.0**	75.8	82.2	89.3	79.1
Rerank-RAG	56.8	67.3	72.1	83.3	69.9	67.1	75.8	82.3	90.1	78.8
MAIN-RAG	55.9	65.8	70.9	83.6	69.0	67.0	75.7	81.2	88.2	78.0
ITEM-A	55.5	65.0	69.8	81.8	68.0	63.7	72.5	82.0	89.3	76.9
Query transformation										
RRR	60.1	72.0	73.6	81.9	71.9	68.7	74.0	83.7	89.4	79.0
LC-MQR	58.9	67.5	73.1	83.3	70.7	67.0	74.7	82.5	88.8	78.2
DMQR	58.0	70.3	73.4	84.1	71.4	66.1	76.5	84.5	90.2	79.3
HyDE	61.0	69.3	71.4	82.9	71.1	67.8	76.5	82.6	89.4	79.1
HYPER (ours)	**63.4**	**73.0**	**77.9**	**85.8**	**75.0**	68.6	**79.7**	**85.4**	**90.6**	**81.1**

Table 2: Decision-Useful Rate (ENTAILED + USEFUL) on MedQA and MMLU-Med for HYPER and baseline methods. HYPER yields the highest rate of decision-useful retrieved contexts across all reported dataset-model cells.

	MedQA					MMLU-Med				
Method	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average
Simple RAG	30.0	30.0	30.0	30.0	30.0	48.7	48.7	48.7	48.7	48.7
Post-retrieval selection										
MMR-RAG	26.4	26.4	26.4	26.4	26.4	43.1	43.1	43.1	43.1	43.1
Rerank-RAG	44.1	44.1	44.1	44.1	44.1	56.8	56.8	56.8	56.8	56.8
MAIN-RAG	23.7	26.4	24.7	26.7	25.4	35.6	36.7	33.7	36.5	35.6
ITEM-A	23.5	25.7	25.2	26.9	25.3	41.0	43.4	43.6	45.5	43.4
Query transformation										
RRR	62.0	68.4	65.4	66.8	65.7	48.1	55.5	59.7	63.3	56.6
LC-MQR	45.2	51.2	62.2	65.6	56.1	51.4	53.7	56.7	59.1	55.3
DMQR	52.4	58	62.2	72.1	61.2	52.2	56.2	57.1	59.4	56.2
HyDE	51.6	54.5	52.8	54.6	53.4	43.0	46.2	44.3	46.3	44.9
HYPER (ours)	**73.4**	**81.8**	**83.2**	**89.8**	**82.1**	**65.8**	**69.1**	**72.5**	**73.7**	**70.3**

then assigns a utility label describing how that context relates to the gold answer.

The judge uses three labels. For brevity, we abbreviate the prompt label USEFUL-BUT-NOT-ENTAILED as USEFUL in tables and formulas.

- **ENTAILED**: the gold answer follows decisively from the retrieved context alone, with no missing fact or outside knowledge, and the context-only `selected_answer` matches the gold answer.
- **USEFUL**: the retrieved context provides strong decision-relevant support for the gold answer but lacks one final inferential bridge. This includes cases where the context contains a decisive clue, rules out plausible alternatives, or makes the gold answer reachable with minimal additional knowledge.
- **NOT USEFUL**: the retrieved context does not provide strong decision-relevant support for the gold answer. This includes background-only passages, keyword overlap without meaningful narrowing power, weak associations, and cases where the context supports another answer more strongly than the gold answer.

Specifically, let N_d denote the number of questions in dataset d, and let $N_{\mathrm{E}}(d, m, g)$ and $N_{\mathrm{U}}(d, m, g)$ be the numbers labeled ENTAILED and USEFUL, respectively, under method m and model family g. Then, **Decision-Useful Rate** (DUR) is defined as

$$\mathrm{DUR}(d, m, g) = \frac{N_{\mathrm{E}}(d, m, g) + N_{\mathrm{U}}(d, m, g)}{N_d}.$$

DUR summarizes the fraction of questions for which retrieval yields context that is either decisive or meaningfully helpful for the correct answer. We also report accuracy within each utility category. To test whether these labels correspond to downstream performance differences, we compare each method against no-retrieval CoT on the same

Table 3: Average context-utility distribution across MedQA and MMLU-Med (%). Values are averaged over the two datasets and four model configurations. USEFUL abbreviates USEFUL-BUT-NOT-ENTAILED.

Method	Entailed	Useful	Not Useful
Simple RAG	14.3	25.1	60.7
MMR-RAG	11.6	23.2	65.3
Rerank-RAG	19.9	30.6	49.5
MAIN-RAG	11.6	18.9	69.5
ITEM-A	15.2	19.2	65.6
RRR	24.9	36.3	38.9
LC-MQR	19.6	36.1	44.3
DMQR	22.9	35.8	41.3
HyDE	16.6	32.5	50.8
HYPER (ours)	34.7	41.5	23.8

Table 4: Utility-conditioned accuracy and same-subset gain over no-retrieval CoT. Results are averaged over four models and ten methods under the configurations of Table 2. Each cell is reported as *accuracy* / Δ_{CoT}.

Label	MedQA	MMLU-Med
ENTAILED	94.3 / +8.0	94.7 / +5.1
USEFUL	78.8 / +4.5	83.6 / +1.3
NOT USEFUL	58.3 / -5.8	70.4 / -3.4

subset of questions.:

$$\Delta_{\text{CoT}}(d, m, g \mid \ell) = \text{Acc}(d, m, g \mid \ell) - \text{Acc}(d, \text{CoT}, g \mid \ell).$$

where the second term evaluates the same model's no-retrieval system on the subset of questions whose retrieved context is assigned label ℓ under method m. This same-subset comparison controls for differences in subset composition.

The context-utility evaluation is conducted using GPT-4.1 as the judge model, and the full prompt and implementation details are provided in the Appendix.

5 Results

5.1 Main accuracy results

Table 1 reports answer accuracy under the shared 15-document final-context budget. Simple retrieval with the raw question is not reliably beneficial: SIMPLE RAG underperforms no-retrieval COT in every dataset-model cell. This indicates that adding retrieved context alone does not necessarily improve answer selection when the retrieved evidence is not aligned with the decision.

HYPER gives the strongest overall performance. Averaged across models, it improves over SIMPLE RAG by 5.9 points on MedQA and 3.6 points on MMLU-Med, and it achieves the best average accuracy on both datasets. The gains are especially clear on MedQA, where HYPER outperforms all baselines in every model setting. On MMLU-Med, HYPER also obtains the best average accuracy, although one individual cell is slightly higher for another method. This suggests that HYPER provides a robust overall advantage.

The comparison across baseline families clarifies the source of HYPER's gains. Post-retrieval selection methods provide limited improvements: RERANK-RAG improves over SIMPLE RAG, but remains below HYPER on average, while MAIN-RAG and ITEM-A do not consistently improve accuracy despite filtering the retrieved context. Diversity-oriented selection through MMR-RAG is also insufficient, suggesting that broader coverage or lower redundancy alone does not guarantee decision-useful evidence. Query-transformation baselines such as RRR, LC-MQR, DMQR, and HYDE are more competitive, but still remain below HYPER on average. This suggests that the key factor is not generic query reformulation, broader coverage, or post-hoc selection, but steering first-pass retrieval toward evidence that can resolve the target decision.

5.2 Context-utility analysis

The context-utility analysis shows that HYPER changes the quality of the retrieved context, not only the final answer. Table 2 shows that HYPER achieves the highest DUR in every dataset-model cell, with especially large gains on MedQA. In contrast, post-retrieval filtering methods such as MAIN-RAG and ITEM-A reduce DUR below SIMPLE RAG on average, and MMR-RAG also remains low despite encouraging diversity. Query-transformation methods improve DUR more reliably, but still remain below HYPER. This suggests that query-transformation or diversification helps, but is less effective than an explicit evidence-seeking retrieval plan.

Table 3 further shows that HYPER increases both ENTAILED and USEFUL contexts while sharply reducing NOT USEFUL contexts. Table 4 validates these labels against downstream behavior: ENTAILED contexts yield the highest accuracy and largest gain over no-retrieval COT, USEFUL contexts remain beneficial, and NOT USEFUL contexts fall below the no-retrieval baseline. Together, these results indicate that HYPER improves accu-

Table 5: Diagnostic control for answer-option exposure.

Method	MedQA	MMLU-Med
LC-MQR	70.7	78.2
LC-MQR + options	71.3	79.2
HYPER − options	73.1	79.9
HYPER	75.0	81.1

Table 6: HYPER query-component ablation.

Method	MedQA	MMLU-Med
HYPER	75.0	81.1
HYPER − q_{sup}	72.4	79.0
HYPER − q_{dist}	73.2	80.1
HYPER − q_{feat}	73.0	80.5

racy by retrieving more decision-useful evidence. Appendix A further shows the same pattern in a Qwen3-4B budget-sensitivity analysis, where HYPER maintains higher accuracy and DUR than Simple RAG and LC-MQR across final-context budgets from 3 to 30 documents.

5.3 Controls and ablations

We next examine which components of HYPER are responsible for the observed gains.

Table 5 evaluates whether improvements arise primarily from exposing answer options during retrieval planning. Adding answer options to a generic multi-query formulator yields only a modest gain over LC-MQR without options, whereas HYPER remains clearly stronger. This gap indicates that HYPER's benefit comes not merely from conditioning queries on the answer choices, but from organizing retrieval around a verification agenda induced by the provisional hypothesis. The no-options variant also provides preliminary evidence that the same idea may extend beyond explicitly enumerated answer choices, when a system can first form a plausible provisional hypothesis.

Table 6 evaluates the contribution of the three rewritten queries. Removing any one of the query roles reduces accuracy on both datasets. The largest decrease follows removal of q_{sup}, the answer-support query, but removing q_{dist} or q_{feat} also degrades performance. These results indicate that the three query roles provide complementary contributions.

6 Discussion

The results suggest that HYPER's main advantage comes from steering first-pass retrieval toward evidence needed for the decision, rather than retrieving more broadly, diversifying the result set, or filtering after retrieval. The increase in ENTAILED/USEFUL contexts supports this mechanism. This design also mitigates a potential confirmation-bias concern: only the support query directly follows the provisional hypothesis, while the distinction and feature queries can surface evidence that challenges it by separating alternatives or grounding question-specific clues. Appendix D illustrates a case where the final generator revises an initially plausible but incorrect hypothesis using the retrieved evidence.

The filtering baselines show why post-hoc selection is limited. Since MAIN-RAG and ITEM-A remove documents from the initial context, lower DUR is not surprising by itself. What matters is that this reduction is not accompanied by accuracy gains. This suggests that, in complex medical decision questions, utility is often distributed across partial clues, criteria, and contrastive evidence, making it difficult to separate removable noise from evidence that is not decisive in isolation. Filtering can therefore discard documents that appear weak individually but are needed collectively to resolve the decision.

The diversity and query-transformation baselines further show that coverage and query variation are insufficient without explicit evidence planning. MMR-RAG broadens coverage but does not ensure decision-aligned evidence. Generic query transformation is stronger, but remains below HYPER on average, suggesting that the key is not query variation alone but an explicit plan for what evidence to seek. Overall, bounded-context decision tasks require retrieval planned around evidence that resolves competing hypotheses.

7 Conclusion

We introduced HYPER, a training-free retrieval planning method that converts a provisional answer hypothesis into evidence-seeking queries. Across MedQA and MMLU-Med, HYPER consistently improves over single-query RAG and outperforms strong baselines on average under a shared final-context budget. These results suggest that, for bounded-context decision tasks, retrieval is most effective when it is planned around the evidence needed to resolve competing hypotheses.

Limitations

This study has several limitations.

First, HYPER depends on the quality of the intermediate retrieval-planning state. If the provisional answer hypothesis, discriminating features, or evidence needs are inaccurate or incomplete, the resulting rewritten queries may be misaligned with the true decision boundary. Although the key feature query is intended to mitigate this risk, it does not eliminate it.

Second, our empirical evaluation is concentrated on medical multiple-choice QA. This is a deliberate choice because the distinction between topical relevance and decision-usefulness is particularly clear in this setting. However, it leaves open the extent to which the same design transfers to other domains or to open-ended tasks in which candidate answers are not explicitly given and must instead be hypothesized.

Third, the context-utility analysis relies on an LLM judge. We use this judge as a comparative operational measure and keep the evaluation prompt fixed across methods, but the resulting labels should not be interpreted as human gold annotations. Accordingly, the utility analysis is best understood as supporting evidence for the proposed mechanism rather than as a definitive causal characterization of retrieval quality.

Finally, our experiments use a fixed corpus, a fixed retriever family, and a fixed final-context budget. The comparisons are normalized by the evidence shown to the final generator rather than by total retrieval-time computation, and some baselines inspect larger intermediate candidate pools by design. We therefore do not claim that the reported gains will transfer unchanged to open-web retrieval, substantially larger corpora, or fully agentic multi-turn retrieval settings.

Ethical Considerations

Our primary experiments are conducted in the medical domain, where errors in retrieval or answer generation may have serious real-world consequences. This work is intended as a study of retrieval behavior on benchmark QA tasks and is not designed for direct clinical use. Any real-world medical deployment would require substantially stronger validation, including expert review, provenance-aware interfaces, calibrated uncertainty handling, abstention mechanisms, and systematic monitoring for harmful retrieval failures.

More broadly, methods that condition retrieval on a provisional answer hypothesis may introduce additional risks if the intermediate hypothesis is inaccurate or systematically biased. Our default design partially mitigates this concern by restricting the role of the hypothesis to retrieval planning rather than exposing it directly to the final generator. However, this design choice does not remove the possibility of biased or misdirected retrieval. Careful evaluation remains necessary in any high-stakes application.

References

Chen Amiraz, Florin Cuconasu, Simone Filice, and Zohar Karnin. 2025. The distracting effect: Understanding irrelevant passages in rag. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18228–18258.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International conference on learning representations*, volume 2024, pages 9112–9141.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336.

Chi-Min Chan, Chunpu Xu, Ruibin Yuan, Hongyin Luo, Wei Xue, Yike Guo, and Jie Fu. 2024. Rq-rag: Learning to refine queries for retrieval augmented generation. *arXiv preprint arXiv:2404.00610*.

Chia-Yuan Chang, Zhimeng Jiang, Vineeth Rakesh, Menghai Pan, Chin-Chia Michael Yeh, Guanchu Wang, Mingzhi Hu, Zhichao Xu, Yan Zheng, Mahashweta Das, and 1 others. 2025. Main-rag: Multi-agent filtering retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2607–2622.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335. Association for Computational Linguistics.

William S Cooper. 1971. A definition of relevance for information retrieval. *Information storage and retrieval*, 7(1):19–37.

Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 758–759.

Yufeng Du, Minyang Tian, Srikanth Ronanki, Subendhu Rongali, Sravan Bodapati, Aram Galstyan, Azton Wells, Roy Schwartz, Eliu A Huerta, and Hao Peng. 2025. Context length alone hurts llm performance despite perfect retrieval. *arXiv preprint arXiv:2510.05381*.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 6491–6501.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1762–1777.

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

Yizheng Huang and Jimmy Xiangji Huang. 2024. A survey on retrieval-augmented text generation for large language models. *ACM Computing Surveys*.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*.

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, Won Kim, Qingyu Chen, Donald C Comeau, Lana Yeganova, W John Wilbur, and Zhiyong Lu. 2023. Medcpt: Contrastive pre-trained transformers with large-scale pubmed search logs for zero-shot biomedical information retrieval. *Bioinformatics*, 39(11):btad651.

Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. 2025. Sufficient context: A new lens on retrieval augmented generation systems. In *International Conference on Learning Representations*, volume 2025, pages 20310–20334.

Hyunjae Kim, Jiwoong Sohn, Aidan Gilson, Nicholas Cochran-Caggiano, Serina Applebaum, Heeju Jin, Seihee Park, Yujin Park, Jiyeong Park, Seoyoung Choi, and 1 others. 2025. Rethinking retrieval-augmented generation for medicine: A large-scale, systematic expert evaluation and practical insights. *arXiv preprint arXiv:2511.06738*.

Yuan-Ching Kuo, Yi Yu, Chih-Ming Chen, and Chuan-Ju Wang. 2025. Mmlf: Multi-query multi-passage late fusion retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6587–6598.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Zhicong Li, Jiahao Wang, Zhishu Jiang, Hangyu Mao, Zhongxia Chen, Jiazhen Du, Yuanxing Zhang, Fuzheng Zhang, Di Zhang, and Yong Liu. 2024. Dmqr-rag: Diverse multi-query rewriting for rag. *arXiv preprint arXiv:2411.13154*.

Jintao Liang, Huifeng Lin, You Wu, Rui Zhao, Ziyue Li, and 1 others. 2025. Reasoning rag via system 1 or system 2: A survey on reasoning agentic retrieval-augmented generation for industry challenges. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 1954–1966.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5303–5315.

Zackary Rackauckas. 2024. Rag-fusion: a new take on retrieval-augmented generation. *arXiv preprint arXiv:2402.03367*.

Alireza Salemi and Hamed Zamani. 2024. Evaluating retrieval quality in retrieval-augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2395–2400.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. Replug: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8371–8384.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 10014–10037.

Yujing Wang, Hainan Zhang, Liang Pang, Binghui Guo, Hongwei Zheng, and Zhiming Zheng. 2025. Maferw: Query rewriting with multi-aspect feedbacks for retrieval-augmented large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25434–25442.

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

Guangzhi Xiong, Qiao Jin, Xiao Wang, Minjia Zhang, Zhiyong Lu, and Aidong Zhang. 2024b. Improving retrieval-augmented generation in medicine with iterative follow-up questions. In *Biocomputing 2025: Proceedings of the Pacific Symposium*, pages 199–214. World Scientific.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation.

Xiaopeng Ye, Chen Xu, Chaoliang Zhang, Zhaocheng Du, Jun Xu, Gang Wang, and Zhenhua Dong. 2025. Q-prm: Adaptive query rewriting for retrieval-augmented generation via step-level process supervision. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 15113–15128.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2023. Making retrieval-augmented language models robust to irrelevant context. *arXiv preprint arXiv:2310.01558*.

Yue Yu, Wei Ping, Zihan Liu, Boxin Wang, Jiaxuan You, Chao Zhang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Rankrag: Unifying context ranking with retrieval-augmented generation in llms. *Advances in Neural Information Processing Systems*, 37:121156–121184.

Hengran Zhang, Keping Bi, Jiafeng Guo, and Xueqi Cheng. 2024a. An iterative utility judgment framework inspired by philosophical relevance via llms. *arXiv preprint arXiv:2406.11290*.

Hengran Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024b. Are large language models good at utility judgments? In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1941–1951.

Appendix

A Additional Results

This section provides the full results underlying the summary tables in the main text, together with additional appendix-only analyses for the ablation settings.

A.1 Additional Sensitivity and Robustness Analyses

This subsection provides additional analyses of baseline coverage, context budget, fusion order, corpus choice, and planner–generator coupling. These analyses complement the main results in Table 1 and 2.

Final-context budget sensitivity. Table 7 varies the final-context budget for Qwen3-4B. For $B = 30$, we increase the vLLM context length to 16,384 tokens. HYPER maintains higher accuracy and DUR than Simple RAG and LC-MQR across all tested budgets, suggesting that its advantage is not explained solely by the amount of context shown to the generator.

Table 7: Qwen3-4B budget sensitivity. Each method cell is reported as *accuracy / DUR*.

Dataset	B	Simple RAG	LC-MQR	HYPER
MedQA	3	70.3 / 13.1	71.4 / 31.2	**73.1 / 62.8**
MedQA	6	71.3 / 19.6	71.7 / 44.4	**74.5 / 73.5**
MedQA	9	71.5 / 22.7	73.1 / 50.6	**75.6 / 77.8**
MedQA	15	71.3 / 30.0	73.1 / 62.2	**77.9 / 83.2**
MedQA	30	72.3 / 36.9	73.3 / 68.9	**77.4 / 86.9**
MMLU-Med	3	81.5 / 31.2	81.7 / 35.2	**84.3 / 52.5**
MMLU-Med	6	82.3 / 37.8	83.3 / 46.5	**84.7 / 62.9**
MMLU-Med	9	81.9 / 41.5	82.6 / 50.4	**85.1 / 67.1**
MMLU-Med	15	81.8 / 48.7	82.5 / 56.7	**85.4 / 72.5**
MMLU-Med	30	81.9 / 51.8	83.1 / 61.3	**84.7 / 73.3**

Fusion-order sensitivity. Table 8 compares the default score fusion with a query-order fusion that appends documents by query order and removes later duplicates. HYPER remains stronger than LC-MQR under both ordering rules, indicating that the gains are stable to this ordering choice. The default score-fusion strategy is slightly better on average for HYPER.

Planner–generator coupling. Table 9 treats the hypothesis formulator and query formulator as a retrieval-planning module and swaps the final generator. The results show that HYPER does not require the planner and generator to be the same model. In particular, contexts planned by the stronger Qwen3-30B planner improve the Qwen3-4B and llama3.2 3b generator, consistent with the higher DUR obtained by the stronger planner.

Corpus sensitivity. Table 10 repeats the Qwen3-4B comparison with PubMed retrieval while keeping the retriever and final-context budget fixed. HYPER remains stronger than Simple RAG and LC-MQR in both accuracy and DUR.

A.2 Reasoning-Enhanced Model Comparisons

Table 11 compares HYPER with thinking-mode and search-trained models, including Search-R1 (Jin et al., 2025). These comparisons are diagnostic rather than directly matched baselines, because they change model training or inference behavior, whereas HYPER keeps the base generator fixed and changes the retrieval-planning interface.

For these runs, we increase the vLLM context length and maximum output length by a factor of three, to 24,576 and 6,144 tokens. We use Qwen/Qwen3-4B-Thinking-2507 and Qwen/Qwen3-30B-A3B-Thinking-2507 for the thinking-mode variants, and PeterJinGo/SearchR1-nq_hotpotqa_train-llama3.2-3b-it-em-ppo for the Search-R1 comparison. For Search-R1, we also increase the maximum number of search turns from the default 4 to 12, since the default setting often spent the available turns issuing queries without producing a final answer in our MedQA setting.

The thinking-mode results show that stronger test-time reasoning can improve performance for sufficiently strong models, as in Qwen3-30B on MedQA. However, this improvement is not uniform: Qwen3-4B Thinking is weaker than the corresponding non-thinking CoT and HYPER settings, and Qwen3-30B HYPER remains comparable to Qwen3-30B Thinking overall. These results suggest that longer reasoning is complementary to, rather than a replacement for, retrieval-side evidence planning.

Search-R1 tests a different question: whether search-oriented RL training transfers to this medical QA setting. Despite increasing the turn budget, Search-R1 performs substantially worse than the corresponding Llama3.2-3B CoT and HYPER settings. This suggests that search training from other domains does not necessarily generalize to complex medical evidence-grounding, where retrieval must target clinical criteria, differential comparisons, and question-specific clues.

Table 8: Fusion-order sensitivity. Values are accuracy (%).

		MedQA					MMLU-Med				
Method	Ordering	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average
LC-MQR	score fusion	58.9	67.5	73.1	83.3	70.7	67.0	74.7	82.5	88.8	78.2
LC-MQR	query order	58.2	67.9	73.1	84.1	70.8	68.4	75.1	81.3	89.7	78.6
HYPER	score fusion	63.4	73.0	77.9	85.8	75.0	68.6	79.7	85.4	90.6	81.1
HYPER	query order	62.5	73.1	76.1	85.3	74.3	69.6	78.6	84.2	89.7	80.5

Table 9: Cross-model planner/generator analysis.

Planner	Generator	MedQA	MMLU-Med
Single model			
Llama3.2 3B	Llama3.2 3B	63.4	68.6
Qwen3 4B	Qwen3 4B	77.9	85.4
Cross model			
Llama3.2 3B	Qwen3 4B	74.0	84.2
Qwen3 4B	Llama3.2 3B	63.2	71.5
Strong planner			
Qwen3 30B	Llama3.2 3B	68.7	72.8
Qwen3 30B	Qwen3 4B	81.9	86.4

Table 10: Qwen3-4B PubMed corpus analysis.

Dataset	Method	Acc.	DUR
MedQA	Simple RAG	68.7	21.8
MedQA	LC-MQR	71.6	51.1
MedQA	HYPER	**73.1**	**77.0**
MMLU-Med	Simple RAG	80.8	43.6
MMLU-Med	LC-MQR	81.5	52.6
MMLU-Med	HYPER	**82.6**	**72.9**

Table 11: Reasoning-enhanced model comparisons.

Method	MedQA	MMLU-Med
Llama3.2 model		
Llama3.2 3B CoT	58.0	66.3
Llama3.2 3B HYPER	63.4	68.6
Search-R1 3B PPO	48.0	57.5
Qwen3 model		
Qwen3 4B CoT	72.8	82.9
Qwen3 4B Thinking CoT	65.4	73.9
Qwen3 4B HYPER	77.9	85.4
Qwen3 30B CoT	84.1	90.2
Qwen3 30B Thinking CoT	86.7	90.4
Qwen3 30B HYPER	85.8	90.6

A.3 Full model-by-model values for the diagnostic controls

Tables 12 and 13 provide the full values underlying the dataset-level averages reported in Tables 5 and 6 of the main text. Reporting the per-model numbers is useful for checking whether the average trends are broad or whether they depend on a single model family. The finer-grained pattern is consistent with the main-text discussion. Exposing the answer options to the LC-MQR produces only modest gains relative to LC-MQR from the question alone, whereas HYPER remains stronger across model families. Likewise, in the query-role ablation, removing any one of the three HYPER queries lowers the overall average, indicating that the three retrieval intents are complementary rather than redundant.

Table 12 makes clear that answer-option exposure by itself does not recover the full benefit of HYPER. While the option-aware LC-MQR control often performs somewhat better than the standard LC-MQR baseline, the gains are smaller and less stable than those achieved by adding hypothesis-guided retrieval planning. This reinforces the same conclusion drawn in the main text, now demonstrated at the level of individual generators.

The full query-ablation table leads to the same interpretation as Table 6 in the main text. The support query q_{sup} is the most important component on average, but the distinction query q_{dist} and the feature query q_{feat} also contribute meaningfully. Cell-level variation remains, as expected, yet the average pattern still supports the view that HYPER works by combining multiple evidence-seeking roles rather than by relying on a single rewritten query.

A.4 Additional DUR results for the controls and ablations

Tables 14 and 15 report DUR for the same diagnostic controls. These results complement the accuracy tables by showing how the corresponding variants affect the fraction of retrieved contexts judged decision-useful. In general, variants that are weaker in accuracy are also weaker in DUR, and the gaps are often larger under DUR than under accuracy. This pattern strengthens the interpretation that the main effect of HYPER is improved evidence selection rather than incidental changes in final answer generation.

For the option-exposure control, DUR shows an even clearer separation between generic option-aware rewriting and HYPER. Adding options to

Table 12: Full model-by-model accuracy values underlying the dataset-level averages reported in Table 5.

	MedQA					MMLU-Med				
Method	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average
LC-MQR	58.9	67.5	73.1	83.3	70.7	67.0	74.7	82.5	88.8	78.2
LC-MQR + options	57.0	68.3	75.8	84.1	71.3	67.5	77.2	83.7	88.4	79.2
HYPER − options	61.1	71.6	75.3	84.5	73.1	69.1	77.3	83.7	89.4	79.9
HYPER (ours)	**63.4**	**73.0**	**77.9**	**85.8**	**75.0**	**68.6**	**79.7**	**85.4**	**90.6**	**81.1**

Table 13: Full model-by-model accuracy values underlying the dataset-level averages reported in Table 6.

	MedQA					MMLU-Med				
Method	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average
HYPER (ours)	**63.4**	**73.0**	**77.9**	85.8	**75.0**	68.6	**79.7**	85.4	**90.6**	**81.1**
HYPER − q_{sup}	62.1	70.1	73.8	83.5	72.4	66.9	77.1	83.0	88.9	79.0
HYPER − q_{dist}	61.2	71.5	74.0	**86.3**	73.2	**69.2**	78.7	83.4	89.0	80.1
HYPER − q_{feat}	61.3	70.9	74.3	85.6	73.0	68.2	77.8	**86.0**	90.1	80.5

Table 14: Additional DUR results for the option-exposure control. DUR denotes ENTAILED + USEFUL.

	MedQA					MMLU-Med				
Method	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average
LC-MQR	45.2	51.2	62.2	65.6	56.1	51.4	53.7	56.7	59.1	55.3
LC-MQR + options	51.1	62.9	71.8	76.4	65.6	56.8	61.3	64.8	66.3	62.3
HYPER − options	61.0	70.5	70.5	80.5	70.6	57.0	61.1	64.2	67.5	62.4
HYPER (ours)	**73.4**	**81.8**	**83.2**	**89.8**	**82.1**	**65.8**	**69.1**	**72.5**	**73.7**	**70.3**

a LC-MQR prompt improves the rate of decision-useful retrieval, but the full HYPER pipeline remains substantially higher. This is consistent with the argument that the benefit comes from how HYPER organizes retrieval around a provisional hypothesis, and the key clues in the stem.

The DUR ablation table mirrors the accuracy ablation but also sharpens it. Removing any one of the three query roles lowers the share of decision-useful contexts, again with the largest average drop after removing q_{sup}. At the same time, the drops for q_{dist} and q_{feat} remain substantial, which supports the claim that alternative discrimination and grounding question-specific clues are not peripheral additions; they help determine whether the final context is actually informative for the downstream decision.

A.5 Full context-utility breakdown

Table 16 expands the summary in Table 3 by reporting the full label distribution for each dataset and model configuration. The more detailed view shows that HYPER does not improve only by shifting a small number of cases from NOT USEFUL to USEFUL. It increases both the ENTAILED and USEFUL rates while reducing NOT USEFUL across nearly all settings. This matters because the ENTAILED subset is the one most strongly associated with downstream answer gains in the main text.

The per-model breakdown also suggests a pattern noted in the main text: the HYPER advantage appears somewhat more pronounced on MedQA than on MMLU-Med, with a similar tendency visible not only in overall DUR but also in the ENTAILED portion.

A.6 Utility-conditioned accuracy by method

Table 18 provides the full values underlying Table 4. The main purpose of this table is to verify that the monotonic pattern discussed in the main text is preserved when we disaggregate by model family and dataset. That pattern remains intact overall: ENTAILED subsets have the highest answer accuracy and the largest positive same-subset gains over no-retrieval CoT, USEFUL subsets remain beneficial but to a smaller extent, and NOT USEFUL subsets are typically weakest. Although individual cells vary, the full table supports the operational validity of the judge labels used throughout the paper.

A.7 Representative Runtime Measurement

To provide an implementation-level inference-time estimate, we rerun selected methods solely for runtime measurement. Table 17 reports end-to-end runtime on the full MedQA evaluation set with Qwen3-4B using a single NVIDIA B200 GPU.

Table 15: Additional DUR results for the HYPER query-role ablation. DUR denotes ENTAILED + USEFUL.

Method	MedQA					MMLU-Med				
	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Average
HYPER (ours)	**73.4**	**81.8**	**83.2**	**89.8**	**82.1**	**65.8**	**69.1**	**72.5**	**73.7**	**70.3**
HYPER − q_{sup}	55.0	62.1	66.5	70.9	63.6	48.5	52.3	55.2	55.8	53.0
HYPER − q_{dist}	60.8	71.7	70.1	80.4	70.8	53.5	57.4	61.8	64.2	59.2
HYPER − q_{feat}	63.1	73.0	74.2	82.6	73.2	51.6	57.0	60.8	63.0	58.1

Table 16: Full context-utility breakdown underlying Table 3. For each method, we report the rates (%) of ENTAILED, USEFUL-BUT-NOT-ENTAILED (USEFUL), and NOT USEFUL contexts across all four model configurations and their average.

Method	Label	MedQA					MMLU-Med				
		Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Avg.	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Avg.
Simple RAG	Entailed	6.0	6.0	6.0	6.0	6.0	22.5	22.5	22.5	22.5	22.5
	Useful	24.0	24.0	24.0	24.0	24.0	26.2	26.2	26.2	26.2	26.2
	Not Useful	70.0	70.0	70.0	70.0	70.0	51.3	51.3	51.3	51.3	51.3
MMR-RAG	Entailed	4.2	4.2	4.2	4.2	4.2	18.9	18.9	18.9	18.9	18.9
	Useful	22.2	22.2	22.2	22.2	22.2	24.2	24.2	24.2	24.2	24.2
	Not Useful	73.6	73.6	73.6	73.6	73.6	56.9	56.9	56.9	56.9	56.9
Rerank-RAG	Entailed	10.7	10.7	10.7	10.7	10.7	29.1	29.1	29.1	29.1	29.1
	Useful	33.4	33.4	33.4	33.4	33.4	27.7	27.7	27.7	27.7	27.7
	Not Useful	55.9	55.9	55.9	55.9	55.9	43.2	43.2	43.2	43.2	43.2
MAIN-RAG	Entailed	6.0	6.2	6.4	6.7	6.3	15.7	17.3	17.2	17.7	17.0
	Useful	17.8	20.2	18.2	20.0	19.0	19.9	19.5	16.5	18.7	18.7
	Not Useful	76.3	73.6	75.3	73.3	74.6	64.4	63.3	66.3	63.5	64.4
ITEM-A	Entailed	6.4	7.1	7.9	7.9	7.3	20.6	22.9	24.0	24.6	23.0
	Useful	17.0	18.5	17.4	18.9	18.0	20.5	20.6	19.7	20.9	20.4
	Not Useful	76.5	74.3	74.8	73.1	74.7	59.0	56.6	56.4	54.5	56.6
RRR	Entailed	20.6	25.5	21.9	23.9	23.0	19.9	26.4	29.2	31.8	26.8
	Useful	41.4	43.0	43.4	43.0	42.7	28.2	29.1	30.5	31.5	29.8
	Not Useful	38.0	31.6	34.6	33.2	34.3	51.9	44.5	40.3	36.7	43.4
LC-MQR	Entailed	11.0	12.7	15.5	18.4	14.4	23.0	23.8	24.5	27.8	24.8
	Useful	34.2	38.5	46.7	47.2	41.7	28.4	29.9	32.2	31.3	30.5
	Not Useful	54.8	48.8	37.8	34.4	43.9	48.6	46.3	43.3	40.9	44.7
LC-MQR + options	Entailed	13.4	18.1	20.8	24.3	19.1	26.4	27.4	28.9	30.7	28.3
	Useful	37.8	44.9	51.0	52.1	46.4	30.5	34.0	35.9	35.6	34.0
	Not Useful	48.9	37.1	28.2	23.6	34.4	43.2	38.7	35.2	33.7	37.7
DMQR	Entailed	16.0	16.7	18.2	24.7	18.9	25.4	26.1	27.3	28.7	26.9
	Useful	36.4	41.2	44.0	47.4	42.3	26.7	30.1	29.8	30.8	29.4
	Not Useful	47.6	42.0	37.8	27.9	38.8	47.8	43.8	42.9	40.6	43.8
HyDE	Entailed	13.0	14.0	13.6	14.5	13.7	18.9	19.9	19.5	19.8	19.5
	Useful	38.6	40.5	39.2	40.1	39.6	24.1	26.3	24.8	26.4	25.4
	Not Useful	48.4	45.5	47.2	45.4	46.6	57.0	53.8	55.7	53.7	55.1
HYPER (ours)	Entailed	29.6	32.0	35.3	41.9	34.7	31.9	34.5	35.5	36.7	34.7
	Useful	43.8	49.8	47.8	47.9	47.3	34.0	34.5	37.0	37.0	35.6
	Not Useful	26.6	18.2	16.8	10.2	17.9	34.2	30.9	27.5	26.3	29.7
HYPER − options	Entailed	18.5	24.9	25.1	31.6	25.0	25.2	28.5	29.1	32.4	28.8
	Useful	42.5	45.6	45.4	48.9	45.6	31.9	32.6	35.1	35.1	33.7
	Not Useful	39.0	29.5	29.5	19.5	29.4	43.0	38.9	35.8	32.5	37.6
HYPER - q_{sup}	Entailed	12.5	12.6	15.5	14.0	13.6	16.4	18.6	19.7	17.9	18.1
	Useful	42.5	49.6	51.0	56.9	50.0	32.2	33.8	35.6	37.9	34.8
	Not Useful	45.0	37.9	33.5	29.1	36.4	51.5	47.7	44.8	44.2	47.0
HYPER - q_{dist}	Entailed	18.2	24.0	25.2	31.7	24.8	23.2	25.5	27.4	27.2	25.8
	Useful	42.5	47.7	44.9	48.7	46.0	30.3	31.9	34.4	36.9	33.4
	Not Useful	39.2	28.3	29.9	19.6	29.2	46.5	42.6	38.2	35.8	40.8
HYPER - q_{feat}	Entailed	20.0	22.9	25.1	31.7	24.9	20.1	23.5	22.5	25.1	22.8
	Useful	43.1	50.0	49.1	50.9	48.3	31.4	33.6	38.2	37.9	35.3
	Not Useful	36.9	27.0	25.8	17.4	26.8	48.4	43.0	39.2	37.0	41.9

These runs use the same optimized vLLM batching implementation for the measured methods. Retrieval computation is performed on CPU in our setup, so the numbers should be interpreted as implementation-dependent measurements rather than hardware-independent method costs.

Compared with SIMPLE RAG, HYPER adds 154.81 seconds on the full MedQA evaluation set.

Method	End-to-end runtime (s)
NO-RAG (COT)	125.29
SIMPLE RAG	398.95
LC-MQR	427.25
HYPER	553.76

Table 17: Representative end-to-end runtime on the full MedQA evaluation set using a single NVIDIA B200 GPU.

This added runtime is on the same order as one COT run in the same setup, reflecting the two short pre-retrieval planning calls used for hypothesis formulation and query formulation.

B Experimental Details

This section records the implementation choices used to instantiate HYPER and the comparison baselines. All LLM components within a method use the same model as the final generator for that experimental setting.

B.1 Shared inference settings

All non-judge model inference was run with vLLM. Unless otherwise specified, the context length was set to 8,192 tokens, the maximum output length was set to 2,048 tokens, and decoding used greedy decoding. The only exception among the standard baselines was HyDE, for which hypothetical-document generation used temperature 0.7.

For the $B = 30$ budget-sensitivity setting, we increased the vLLM context length to 16,384 tokens. For thinking-model and Search-R1 comparisons, we increased the context length and maximum output length to 24,576 and 6,144 tokens, respectively.

All retrieval-based methods used the same MedCPT retriever over the same medical textbook corpus and were evaluated under the same final generator-context budget of 15 documents. The comparison is therefore normalized by the evidence shown to the final generator, even when methods differ in how many intermediate candidates they inspect before selecting that final context.

B.2 Computational Budget Accounting

Table 19 reports the analytical inference budget of each method, excluding the shared final answer-generation call. We count pre-final LLM calls and retrieval queries to summarize the main method-level cost differences.

HYPER adds two fixed pre-retrieval LLM calls for hypothesis formulation and query formulation. This makes it more expensive than single-query retrieval, but its LLM-call budget remains smaller than post-retrieval filtering methods such as MAIN-RAG and ITEM-A, which require multiple LLM judgments over candidate documents.

B.3 No-retrieval CoT and Simple RAG

The no-retrieval baseline answers directly from the question and options using the CoT prompt in Figure 3. SIMPLE RAG uses the raw question as a single retrieval query, retrieves the top 15 documents, and passes them directly to the shared final generator prompt in Figure 16. These two baselines define the lower and upper ends of the minimal retrieval setup used throughout the paper: the former removes retrieval entirely, while the latter introduces retrieval without any additional planning, transformation, reranking, or filtering.

B.4 HYPER

HYPER is instantiated as the two-stage pre-retrieval pipeline described in Section 4. The Hypothesis Formulator (Figure 13) reads the question and answer options and returns a structured retrieval-planning state containing discriminating features, a brief rationale, evidence needs, and a provisional answer hypothesis. The Query Formulator (Figure 15) then converts that state into exactly three queries corresponding to the roles described in the main text: support for the provisional answer hypothesis, distinction from competing options, and grounding salient question features.

Each of the three queries retrieves the top 5 documents. The retrieved sets are merged, deduplicated, and truncated only as needed to satisfy the shared final-context budget. By default, the provisional answer hypothesis is used only for retrieval planning and is not shown to the final generator. The HYPER − options control removes the answer options from the hypothesis-formulation prompt while leaving the rest of the pipeline unchanged.

B.5 LC-MQR

LC-MQR baseline follows the standard paraphrase-style multi-query rewriting setup with retrieval fusion. Concretely, we use the same three-query prompt family as the LC-MQR baseline in prior work (Kuo et al., 2025), prompting the model to produce exactly three paraphrase-style queries from the original question. The option-aware control in Figure 5 additionally exposes the answer options but does not introduce any structured retrieval-

Table 18: Full model-by-model values underlying Table 4. Each cell is reported as *accuracy* / Δ_{CoT}.

		MedQA					MMLU-Med				
Method	Label	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Avg.	Llama3.2 3B	Llama3.1 8B	Qwen3 4B	Qwen3 30B	Avg.
Simple RAG	Entailed	93.5 / +9.1	93.5 / +9.1	93.5 / +9.1	93.5 / +9.1	93.5 / +9.1	97.1 / +6.1	97.1 / +6.1	97.1 / +6.1	97.1 / +6.1	97.1 / +6.1
	Useful	85.6 / +5.9	85.6 / +5.9	85.6 / +5.9	85.6 / +5.9	85.6 / +5.9	88.1 / +0.7	88.1 / +0.7	88.1 / +0.7	88.1 / +0.7	88.1 / +0.7
	Not Useful	64.5 / -4.9	64.5 / -4.9	64.5 / -4.9	64.5 / -4.9	64.5 / -4.9	71.9 / -5.2	71.9 / -5.2	71.9 / -5.2	71.9 / -5.2	71.9 / -5.2
MMR-RAG	Entailed	98.1 / +13.0	98.1 / +13.0	98.1 / +13.0	98.1 / +13.0	98.1 / +13.0	95.1 / +4.9	95.1 / +4.9	95.1 / +4.9	95.1 / +4.9	95.1 / +4.9
	Useful	84.8 / +5.0	84.8 / +5.0	84.8 / +5.0	84.8 / +5.0	84.8 / +5.0	86.3 / -1.1	86.3 / -1.1	86.3 / -1.1	86.3 / -1.1	86.3 / -1.1
	Not Useful	66.4 / -3.6	66.4 / -3.6	66.4 / -3.6	66.4 / -3.6	66.4 / -3.6	76.1 / -2.4	76.1 / -2.4	76.1 / -2.4	76.1 / -2.4	76.1 / -2.4
Rerank-RAG	Entailed	97.8 / +8.1	97.8 / +8.1	97.8 / +8.1	97.8 / +8.1	97.8 / +8.1	95.6 / +6.0	95.6 / +6.0	95.6 / +6.0	95.6 / +6.0	95.6 / +6.0
	Useful	81.6 / +5.2	81.6 / +5.2	81.6 / +5.2	81.6 / +5.2	81.6 / +5.2	82.1 / -2.0	82.1 / -2.0	82.1 / -2.0	82.1 / -2.0	82.1 / -2.0
	Not Useful	61.5 / -5.9	61.5 / -5.9	61.5 / -5.9	61.5 / -5.9	61.5 / -5.9	73.4 / -4.3	73.4 / -4.3	73.4 / -4.3	73.4 / -4.3	73.4 / -4.3
MAIN-RAG	Entailed	86.8 / +17.1	91.1 / +10.1	97.6 / +6.1	100.0 / +4.7	93.9 / +9.5	93.6 / +6.4	95.2 / +5.3	97.9 / +2.7	99.0 / -0.5	96.4 / +3.5
	Useful	60.0 / +1.8	70.0 / -1.9	79.7 / +4.7	89.4 / +3.1	74.8 / +1.9	73.0 / +7.4	81.6 / -0.9	91.1 / -1.7	91.7 / +0.5	84.4 / +1.3
	Not Useful	52.8 / -4.0	62.6 / -4.7	66.4 / -4.3	80.6 / -2.1	65.6 / -3.8	59.2 / -2.4	68.7 / -0.3	74.7 / -2.5	84.3 / -3.0	71.7 / -2.1
ITEM-A	Entailed	84.1 / +12.2	94.5 / +11.0	98.0 / +13.0	99.0 / +2.0	93.9 / +9.5	87.1 / +7.6	93.2 / +6.4	96.6 / +5.7	99.6 / +2.6	94.1 / +5.6
	Useful	68.2 / +7.4	80.5 / +4.2	87.8 / +7.7	93.8 / +3.3	82.6 / +5.7	67.7 / -1.3	80.8 / +1.8	87.9 / +1.9	92.5 / +2.2	82.2 / +1.1
	Not Useful	50.3 / -6.6	58.4 / -7.5	62.6 / -7.2	76.7 / -4.5	62.0 / -6.5	54.3 / -6.6	61.1 / -8.0	73.8 / -4.7	83.6 / -3.5	68.2 / -5.7
RRR	Entailed	82.1 / +6.1	93.5 / +7.4	94.3 / +6.5	97.7 / +2.3	91.9 / +5.6	85.3 / +7.8	93.4 / +6.6	95.3 / +7.2	96.8 / +0.3	92.7 / +5.5
	Useful	64.6 / +6.5	73.5 / +5.5	77.4 / +4.3	83.3 / -0.4	74.7 / +4.0	73.6 / +5.5	78.2 / +3.2	86.1 / +0.6	90.1 / +0.3	82.0 / +2.4
	Not Useful	43.6 / -6.0	52.5 / -4.2	55.8 / -7.3	68.6 / -8.3	55.1 / -6.5	59.7 / -1.4	60.4 / -8.0	73.6 / -3.6	82.7 / -2.5	69.1 / -3.9
LC-MQR	Entailed	86.4 / +13.6	94.4 / +11.7	98.5 / +8.6	97.4 / +0.4	94.2 / +8.6	84.9 / +9.6	91.9 / +7.4	95.5 / +6.0	98.7 / +1.7	92.7 / +6.1
	Useful	66.5 / +8.9	74.5 / +2.2	78.5 / +4.2	88.0 / +0.5	76.9 / +4.0	70.6 / +1.6	77.8 / +1.8	88.0 / +0.0	90.0 / +0.6	81.6 / +1.0
	Not Useful	48.6 / -6.6	54.9 / -8.1	55.9 / -8.1	69.3 / -3.7	57.2 / -6.6	56.7 / -4.0	64.0 / -6.0	70.9 / -4.5	81.3 / -5.0	68.2 / -4.8
DMQR	Entailed	79.4 / +4.4	93.9 / +7.0	97.0 / +6.5	98.4 / +1.3	92.2 / +4.8	84.7 / +5.8	93.7 / +7.4	96.0 / +5.7	97.1 / +1.0	92.9 / +5.0
	Useful	63.9 / +5.8	80.6 / +7.4	80.0 / +5.4	87.6 / +0.3	78.0 / +4.7	71.8 / +4.5	79.3 / +3.4	87.7 / +2.2	91.9 / +0.6	82.7 / +2.6
	Not Useful	46.2 / -7.1	50.9 / -6.9	54.3 / -7.9	65.2 / -2.5	54.1 / -6.1	53.5 / -5.8	64.3 / -4.0	74.9 / -1.5	84.1 / -1.1	69.2 / -3.1
HyDE	Entailed	81.8 / +6.7	93.8 / +10.7	94.8 / +4.6	99.5 / +1.6	92.5 / +5.9	91.7 / +9.7	94.9 / +4.6	98.1 / +2.8	98.6 / -0.5	95.8 / +4.2
	Useful	60.7 / +5.1	72.3 / +1.7	76.3 / +3.6	84.1 / +0.2	73.3 / +2.7	71.0 / +3.8	82.1 / +2.1	88.5 / +1.9	95.1 / +2.8	84.2 / +2.6
	Not Useful	49.5 / -5.7	59.4 / -4.2	60.7 / -7.2	77.0 / -3.3	61.7 / -5.1	58.8 / -1.8	67.7 / +0.0	74.8 / -2.0	84.1 / -1.7	71.4 / -1.4
HYPER (ours)	Entailed	89.1 / +14.3	94.3 / +8.8	97.1 / +6.9	98.5 / +3.2	94.8 / +8.3	90.5 / +8.1	92.3 / +5.1	98.2 / +3.4	97.5 / +1.2	94.6 / +4.4
	Useful	64.5 / +6.3	74.6 / +6.8	77.3 / +7.4	84.9 / +2.5	75.3 / +5.7	73.0 / +4.6	81.9 / +7.7	84.6 / +3.7	91.6 / +1.0	82.8 / +4.3
	Not Useful	32.9 / -5.9	31.2 / -12.1	39.3 / -5.1	37.7 / -10.0	35.3 / -8.3	44.1 / -5.4	63.9 / +1.2	69.9 / -0.3	80.0 / -1.4	64.5 / -1.5

planning state. In both cases, we retrieve the top 5 documents per query, take the union of the retrieved sets, remove duplicates, and pass the resulting context to the same final generator prompt used by all methods. These controls are therefore matched to HYPER in the number of rewritten queries and in the final evidence budget; what differs is how retrieval intent is constructed.

B.6 HyDE

HyDE uses the prompt in Figure 6 to generate hypothetical documents from the input question. In our implementation, we generate 8 hypothetical passages per question. The original query embedding and the 8 hypothetical-passage embeddings are then averaged to form the retrieval representation used by MedCPT. As noted above, HyDE is the only method that does not use greedy decoding for its intermediate generation step: we use temperature 0.7 so that the synthetic passages are not collapsed into near-duplicates.

B.7 Rerank-RAG

RERANK-RAG first retrieves 150 candidate documents with the raw question using MedCPT and then reranks those candidates with the BGE reranker v2-m3 cross-encoder (Chen et al., 2024). Reranking is performed on the question–document pair, with a batch size of 16 and a maximum sequence length of 512 tokens. The top 15 reranked documents are then passed to the shared final generator. This baseline therefore tests whether stronger post-retrieval selection can compensate for starting from a raw-question first pass.

B.8 MAIN-RAG

MAIN-RAG follows the retrieve-then-filter design described in the main text. It begins with 15 documents retrieved from the raw question. For each candidate document, Agent-1 (Figure 7) produces a document-conditioned answer choice, and Agent-2 (Figure 8) judges whether the document contains sufficient information to support that an-

Method	LLM calls	Retrieval queries
CoT	0	0
SIMPLE RAG	0	1
Post-retrieval selection		
MMR-RAG	0	1
RERANK-RAG	0	1
MAIN-RAG	30	1
ITEM-A	4-6	1
Query transformation		
RRR	1	1
LC-MQR	1	3
DMQR	1	5
HYDE	8	1
HYPER	2	3

Table 19: Analytical per-question computational budget. Counts exclude the shared final answer-generation call.

swer. We quantify the Agent-2 judgment using the log-probability difference between the generated `Yes` and `No` tokens and retain documents whose scores are at least the mean score of the candidate set. The surviving documents constitute the final context shown to the shared generator.

B.9 MMR-RAG

MMR-RAG is a post-retrieval diversity selection baseline based on maximal marginal relevance (Carbonell and Goldstein, 1998). We first retrieve 150 candidate documents using the raw question and the shared MedCPT retriever. From this pool, we greedily select 15 documents by balancing similarity to the question and dissimilarity to the documents already selected. We use inner product similarity over the stored MedCPT embeddings and set the diversity weight to $\lambda = 0.5$. The selected documents are passed to the final generator in their MMR selection order.

B.10 RRR

RRR follows the training-free query-rewriting setting of Rewrite-Retrieve-Read (Ma et al., 2023). We use an LLM as a frozen rewriter that receives the input question and answer options and produces a single rewritten search query before retrieval. We use zero-shot prompting for this step to avoid adding task-specific demonstrations. The rewritten query is then used to retrieve the top 15 documents with MedCPT, and the retrieved documents are passed to the shared final generator. Thus, this baseline differs from SIMPLE RAG only in replacing the raw question with a single rewritten query. The exact prompt used for query rewriting is provided in Figure 9.

B.11 DMQR

DMQR is adapted from DMQR-RAG (Li et al., 2024) as an information-based multi-query rewriting baseline. We use a fixed strategy set rather than adaptive strategy selection. Given the input question, the rewriter generates up to four rewrites corresponding to general query rewriting, keyword rewriting, pseudo-answer rewriting, and core-content extraction. Together with the original question, this yields up to five retrieval queries after removing duplicate rewrites. The exact prompt used to generate the strategy-specific rewrites is provided in Figure 10.

To keep the final-context budget matched to the rest of our experiments, we allocate a total retrieval budget of 15 documents across the available queries. This corresponds to retrieving the top 3 documents per query when all five queries are available. The resulting lists are fused using the same score-fusion rule as HYPER: duplicate documents are merged, retrieval scores are summed across lists, and the final context is ranked by the aggregate score. We do not use the original DMQR-RAG web-search retriever, BGE reranking stage, or adaptive strategy selector, since our goal is to compare the effect of information-diverse query rewriting under the same retriever, corpus, and generator interface as the other methods.

B.12 ITEM-A

ITEM-A is a post-retrieval utility-selection baseline adapted from the ITEM framework (Zhang et al., 2024a). We use the answer-in-the-loop variant without the additional relevance-reranking loop, so that the baseline tests utility-based filtering rather than reranking. For each question, we first retrieve the top 15 documents using the raw question. The LLM then alternates between generating a pseudo-answer from the currently selected evidence and selecting documents from the original candidate set that are useful for answering or verifying the question. The prompts used for pseudo-answer generation and utility selection are provided in Figures 11 and 12, respectively.

We run at most three iterations and stop early when the selected document set does not change. The final selected documents are passed to the same final generator prompt used by the other retrieval-based methods. Since ITEM-A filters the initial raw-question context, its final context may contain

fewer than 15 documents. In our runs, ITEM-A returned an empty selected set for a small fraction of examples, with an average rate of 2.4% across datasets. These cases are passed to the generator with an empty retrieved context and are counted as NOT USEFUL in the context-utility evaluation.

B.13 Context-utility judge

The context-utility analysis uses the judge prompt in Figure 2. Given the question, options, gold answer, and the retrieved context set produced by a method, the judge first identifies the answer option most strongly supported by the context alone (`selected_answer`) and then assigns one of the three labels used in the paper: ENTAILED, USEFUL-BUT-NOT-ENTAILED, or NOT USEFUL. The prompt is intentionally conservative about ENTAILED and USEFUL-BUT-NOT-ENTAILED; background-only passages, weak topical relevance, and contexts that support a competing option are labeled NOT USEFUL. We keep this prompt fixed across all methods so that the utility analysis reflects differences in retrieved evidence rather than differences in the judging procedure.

C Prompt Templates

This section reproduces the prompt templates used in the experiments. Figures 3 and 16 show the answer-generation prompts for the no-retrieval and retrieval-based settings, respectively. Figures 4–12 show the prompts used by the comparison baselines, Figure 2 shows the context-utility judge, and Figures 13–15 show the HYPER prompts, including the no-options variant used in the diagnostic control. We include the exact templates because several of the compared methods differ primarily in how they formulate retrieval intent.

D Illustrative Example

Figures 17–21 present a representative MedQA example that makes the logic of HYPER concrete. The example is useful precisely because the intermediate hypothesis is provisional rather than guaranteed to be correct. As in the main method, the formulator is asked to propose a provisional answer hypothesis that can organize retrieval, not to lock in the final answer. The support query follows that provisional guess, while the distinction and feature queries are intended to surface the evidence that can either confirm it or overturn it.

Example walkthrough. In this example, the hypothesis formulator initially leans toward a clubfoot-style interpretation and therefore proposes a corresponding management hypothesis. HYPER does not treat that intermediate state as authoritative. Instead, it uses the intermediate state to generate one query that follows the current hypothesis and two additional queries that probe the nearby decision boundary. The resulting retrieved set contains both hypothesis-supporting and hypothesis-challenging evidence, including passages that distinguish a flexible forefoot deformity with a neutral heel from true clubfoot. The final generator then produces its answer from the retrieved evidence and the original question, rather than from the provisional hypothesis itself. As shown by the raw output in Figure 21, the model can revise the initial guess when the retrieved evidence points elsewhere. This is the intended role of hypothesis conditioning throughout the paper: it sharpens first-pass retrieval while preserving the ability of the downstream generator to correct an initially plausible but imperfect hypothesis.

GPT Judge Prompt

System:
You evaluate the relationship between retrieved context and the gold answer for a multiple-choice question.

User:
QUESTION
{question}

OPTIONS
{options}

GOLD ANSWER
{gold_line}

RETRIEVED CONTEXT
{chunks}

TASK
Given the retrieved context, evaluate its relationship to the gold answer.
1. Identify the answer option most strongly supported by the retrieved context alone. Call this śelected_answer.
2. Then classify how the retrieved context relates to the gold answer: whether it entails the gold answer, provides useful but incomplete support for it, or is not useful for supporting it.

LABELS

1. ENTAILED
Definition:
The gold answer is entailed by the retrieved context alone.
The reasoning chain from the retrieved context to the gold answer is complete within the provided context.

Required conditions:
- The gold answer follows from the retrieved context alone.
- No outside knowledge, missing fact, or unstated assumption is needed.
- The conclusion is decisive, not merely plausible.
- 'selected_answer' matches the gold answer.

2. USEFUL_BUT_NOT_ENTAILED
Definition:
The retrieved context provides strong decision-relevant support for the gold answer, but the gold answer is not fully entailed by the retrieved context alone. A small additional reasoning step, minor outside knowledge, or a single missing bridge may still be needed, but the retrieved context already contains evidence that substantially narrows the decision toward the gold answer.

This label should be used only when the retrieved context contains at least one of the following:
- a decisive clue, rule, condition, or relation that strongly supports the gold answer, even if one final bridge is missing
- evidence that rules out or strongly weakens one or more plausible alternative answers, leaving the gold answer as the clearly favored option
- a partial but highly informative connection between the question and the gold answer that would make the correct answer reasonably reachable with only minimal additional knowledge

Do not use this label for context that is merely relevant, broadly related, or weakly suggestive.

3. NOT_USEFUL
Definition:
The retrieved context does not provide strong decision-relevant support for the gold answer.
It may be topically related, mention similar concepts, or provide background information, but it does not contain a decisive clue, a strong exclusion of alternatives, or a core connection that materially helps determine the gold answer.

RULES
- Evaluate only the retrieved context.
- Evaluate the label relative to the gold answer.
- Infer 'selected_answer' from the retrieved context alone; do not force it to equal the gold answer.
- Be conservative about ENTAILED.

Output only JSON on the last line:
{{"selected_answer": "<A|B|C|D|UNSURE>", "label": "<ENTAILED|USEFUL_BUT_NOT_ENTAILED|NOT_USEFUL>"}}

Figure 2: Prompt used for the context-utility judge.

COT Prompt

System:
You are a helpful medical expert, and your task is to answer a multi-choice medical question. Please first think step-by-step and then choose the answer from the provided options. Organize your output in a json formatted as Dict{"step_by_step_thinking": Str(explanation), "answer_choice": Str{A/B/C/...}}. Your responses will be used for research purposes only, so please have a definite answer.

User:
Here is the question:
{question}

Here are the potential choices:
{options}

Please think step-by-step and generate your output in json:

Figure 3: Prompt used for the no-retrieval CoT baseline.

LC-MQR Prompt

User:
You are an AI language model assistant. Your task is to generate exactly three different versions of the given user question to retrieve relevant documents from a vector database. By generating multiple perspectives on the user question, your goal is to help the user overcome some of the limitations of the distance-based similarity search.

Original question: {query}

Format your response in plain text as:

Sub-query 1:

Sub-query 2:

Sub-query 3:

Figure 4: The prompt used for LC-MQR baseline.

LC-MQR with Options Prompt

User:
You are an AI language model assistant. Your task is to generate exactly three different versions of the given user question to retrieve relevant documents from a vector database. By generating multiple perspectives on the user question, your goal is to help the user overcome some of the limitations of the distance-based similarity search.

Original question: {query}

Options:
{options}

Format your response in plain text as:

Sub-query 1:

Sub-query 2:

Sub-query 3:

Figure 5: The prompt used for LC-MQR with options ablation.

HyDE Prompt

User:
Please write a passage to answer the question.
Question: {question}
Passage:

Figure 6: Prompt used for generating hypothetical documents in the HyDE baseline.

MAIN-RAG: Agent-1 (Predictor) Prompt

System:
You are an accurate and reliable AI assistant that can answer questions with the help of external documents. You should only provide the correct answer without repeating the question and instruction.

User:
Document:
{context}

Question:
{question}

Options:
{options}

Based ONLY on the provided document, what is the best answer to the question? Output only the answer choice.

Figure 7: The prompt used for Agent-1 (Predictor) in the MAIN-RAG baseline to generate document-conditioned answers for the Document-Query-Answer triplet.

MAIN-RAG: Agent-2 (Judge) Prompt

System:
You are a noisy document evaluator that can judge if the external document is noisy for the query with unrelated or misleading information. Given a retrieved Document, a Question, and an Answer generated by an LLM (LLM Answer), you should judge whether both the following two conditions are reached: (1) the Document provides specific information for answering the Question; (2) the LLM Answer directly answers the question based on the retrieved Document. Please note that external documents may contain noisy or factually incorrect information. If the information in the document does not contain the answer, you should point it out with evidence. You should answer with 'Yes' or 'No' with evidence of your judgment, where 'No' means one of the conditions (1) and (2) are unreached and indicates it is a noisy document.

User:
Document:
{context}

Question with Options:
{question}
{options}

LLM Answer:
{llm_answer}

Does the document contain enough specific information to answer the question, and is the LLM Answer directly supported by the document? Answer with exactly one token: Yes or No.

Figure 8: The prompt used for Agent-2 (Judge) in the MAIN-RAG baseline to evaluate the decision-usefulness and reliability of the retrieved document.

RRR Prompt

User:
Provide a better search query for web search engine to answer the given question, end the query with '**'. Question: {question} Options:{options}

Query:

Figure 9: The prompt used for RRR baseline.

DMQR Prompt

User:
Instruction
You will receive a user's question that requires retrieving relevant content through internet search to provide an answer. There are now the following four rewriting methods, General Search Rewriting, Keyword Rewriting, Pseudo-Answer Rewriting, Core Content Extraction.
Please apply four rewriting methods to rewrite the question.
General Search Rewriting
Rewrite the question into a general query for internet search.
Keyword Rewriting
Extract all keywords from the question and separate them with commas, preserving the amount of information as in the original question.
Pseudo-Answer Rewriting
Generate an answer for the question, and use the answer to match the real answers from the search engine.
Core Content Extraction
Reduce the amount of information in the original question, only extracting the most core content.
The rewritten query should be more brief than Keyword Rewriting.
Example
Question: Which city was the site where the armistice agreement officially ending World War I was signed?
Output:
General Search Rewriting: City where World War I armistice agreement was signed
Keyword Rewriting: World War I, Armistice, Signing Location
Pseudo-Answer Rewriting: The armistice that ended World War I was signed in the city of Compiegne.
Core Content Extraction: World War I armistice signing city
Begin! Only output the final result without any additional content. Do not generate any other unrelated content.
Question: {query}
Output:

Figure 10: The prompt used for DMQR baseline.

ITEM-A: Generating pseudo answer prompt

User:
You are a faithful question and answer assistant. Answer the question based on the given information with one or few sentences without the source.
assistant:
Yes, i am the faithful question and answer assistant.
User:
Given the information:
{Documents}
Answer the following question based on the given information with one or few sentences without the source.
Question: {question}

Answer:

Figure 11: The prompt used for generating pseudo answer in ITEM-A baseline.

ITEM-A: Utility selection prompt

User:
You are the utility judger, an intelligent assistant that can select the passages that have utility in answering the question.
assistant:
Yes, i am the utility judger.
User:
I will provide you with {num} passages, each indicated by number identifier [].
I will also provide you with a reference answer to the question.
Select the passages that have utility in generating the reference answer to the following question from the {num} passages: {query}.
assistant:
Okay, please provide the passages and the reference answer. **User:**
[1] {passage_1}
assistant:
Received passage [1].
User:
[2] {passage_2}
assistant:
Received passage [2].
.....
User:
Question: {query}.
Reference answer: {pseudo answer}.

The requirements for judging whether a passage has utility in answering the question are: The passage has utility in answering the question, meaning that the passage not only be relevant to the question, but also be useful in generating a correct, reasonable and perfect answer to the question.
Directly output the passages you selected that have utility in generating the reference answer to the question. The format of the output is: 'My selection:[[i],[j],...].'. Only response the selection results, do not say any word or explain.

Figure 12: The prompt used for utility selection in ITEM-A baseline.

HYPER: Hypothesis Formulator Prompt

System:
You are an expert analyst taking an exam.

User:
Question: {question}

Options:
{options}

Analyze this question carefully. Think step-by-step about each option, considering the information given in the question and relevant knowledge. Reason through the elimination analysis before making your final assessment.

After your analysis, provide your final assessment in JSON:
```
{
    "discriminating_features": ["2-3 features that distinguish between options"],
    "reasoning": "brief explanation why this is the best answer",
    "confirming_evidence": ["1-3 specific facts that would confirm this answer"],
    "best_guess": "A/B/C/D",
    "best_guess_text": "«copy the chosen option text verbatim»"
}
```

Figure 13: Prompt used for the HYPER hypothesis-formulation stage.

HYPER: Hypothesis Formulator without Options Prompt

System:
You are an expert analyst taking an exam.

User:
Question: {question}

Analyze this question carefully. Think step-by-step about each option, considering the information given in the question and relevant knowledge. Reason through the elimination analysis before making your final assessment.

After your analysis, provide your final assessment in JSON:
```
{
    "discriminating_features": ["2-3 features that distinguish between options"],
    "reasoning": "brief explanation why this is the best answer",
    "confirming_evidence": ["1-3 specific facts that would confirm this answer"],
    "best_guess": "write your best guess for the answer"
}
```

Figure 14: The prompt used for the Hypothesis Formulator (Stage 1) without options for ablation.

HYPER: Query Formulator Prompt

System:
You are a search query expert. Generate precise, targeted search queries. Output ONLY the 3 queries in the exact format requested.

User:
Generate 3 highly targeted search queries to find evidence for this question.

Question: {question}
Best Guess Answer: {best_guess_text}
Reasoning: {reasoning}
Evidence Needed: {confirming_evidence}
Key Features: {discriminating_features}

Generate 3 SPECIFIC queries:
Query 1: Find evidence supporting {best_guess_text} - focus on the main reasoning
Query 2: Find distinguishing criteria between the top candidate answers
Query 3: Find specific key features or facts

Format:
Query 1: [query]
Query 2: [query]
Query 3: [query]

Figure 15: Prompt used for the HYPER query-formulation stage.

Final Generator Prompt

System:
You are a helpful medical expert, and your task is to answer a multi-choice medical question using the relevant documents. Please first think step-by-step and then choose the answer from the provided options. Organize your output in a json formatted as Dict{"step_by_step_thinking": Str(explanation), "answer_choice": Str{A/B/C/...}}. Your responses will be used for research purposes only, so please have a definite answer.

User:
Here are the relevant documents:
{context}

Here is the question:
{question}

Here are the potential choices:
{options}

Please think step-by-step and generate your output in json:

Figure 16: The standard generator prompt (MIRAGE template) used uniformly across all evaluated methods, including Simple RAG, all baselines, and our proposed HYPER, to strictly control the post-retrieval generation phase.

Example Question (MedQA)

Question:
A 3-week-old male newborn is brought to the physician because of an inward turning of his left forefoot. He was born at 38 weeks' gestation by cesarean section because of breech presentation. The pregnancy was complicated by oligohydramnios. Examination shows concavity of the medial border of the left foot with a skin crease just below the ball of the great toe. The lateral border of the left foot is convex. The heel is in neutral position. Tickling the lateral border of the foot leads to correction of the deformity. The remainder of the examination shows no abnormalities. X-ray of the left foot shows an increased angle between the 1st and 2nd metatarsal bones.
Which of the following is the most appropriate next step in the management of this patient?
Options
A: Foot abduction brace
B: Arthrodesis of the forefoot
C: Reassurance
D: Tarsometatarsal capsulotomy

Figure 17: Representative MedQA example used to illustrate the end-to-end HYPER workflow.

Hypothesis Formulation

Discriminating Features:
The clinical presentation includes a medial concavity of the foot with a skin crease and correction with tickling of the lateral border—classic signs of *talipes equinovarus* (congenital clubfoot) in the early stages.

Confirming Evidence:
The patient is 3 weeks old, which falls within the optimal age range for early intervention in congenital clubfoot.

Reasoning:
The clinical findings are consistent with congenital clubfoot. In infants younger than 6 months with a flexible deformity, the appropriate initial management is typically non-surgical correction using a foot abduction brace.

Provisional Answer Hypothesis:
Foot abduction brace

Figure 18: Illustrative provisional output of the Hypothesis Formulator. The Provisional Answer hypothesis is used to steer retrieval and may later be confirmed or overturned by the retrieved evidence.

Query Formulator Output

Support Query
Foot abduction brace for congenital clubfoot in infants under 6 months with flexible deformity and response to tickling: evidence for first-line non-surgical management in early-stage talipes equinovarus

Distinction Query
Distinguishing features between foot abduction brace, arthrodesis, tarsometatarsal capsulotomy, and reassurance in the management of early congenital clubfoot in neonates under 3 months of age

Feature Query
Clinical and radiographic key features of congenital clubfoot in 3-week-old infants: medial concavity, lateral convexity, skin crease below great toe, correction with lateral tickling, and increased 1st–2nd metatarsal angle on X-ray

Figure 19: Three intent-aligned queries generated from the provisional hypothesis: answer support, alternative discrimination, and grounding of question-specific clues.

Retrieved Evidence Examples

Support / Distinction Evidence
"Treatment of In-toeing. The mainstay of management is to identify patients who have pathologic reasons for in-toeing and provide reassurance and follow-up to document improvement. It can take until 7 to 8 years of age for correction. Fewer than 1% of patients with in-toeing will require surgical intervention due to functional disability or cosmetic concerns."

Distinction / Key Feature Evidence
"Family history is important. Clubfoot can be congenital, teratologic, or positional. Congenital clubfoot (approximately 75% of cases) is typically isolated, although infants should be evaluated for developmental dysplasia of the hip. The infant will have hindfoot equinus and varus, forefoot adduction, and varying degrees of rigidity. These findings arise from abnormalities in the talonavicular joint."

Figure 20: Representative passages retrieved by the generated queries. The retrieved set can contain both supporting and corrective evidence, allowing the final generator to revise the provisional hypothesis when necessary.

Generator Raw Output Examples

The patient is a 3-week-old male with an inward turning of the left forefoot, described as concavity of the medial border of the foot with a skin crease just below the ball of the great toe and a convex lateral border. The heel is in neutral position, and tickling the lateral border leads to correction of the deformity. This indicates that the deformity is not fixed and can be passively corrected — a key feature of a positional deformity rather than a rigid congenital deformity like clubfoot.
The x-ray shows an increased angle between the 1st and 2nd metatarsal bones, which is consistent with metatarsus adductus — a condition where the forefoot is adducted (turned inward), often due to in utero positioning. This is a common, benign condition in newborns, especially in the context of a breech presentation and oligohydramnios, which may have contributed to abnormal positioning in utero.
Document [4] (Surgery_Schwartz) states: 'Metatarsus adductus in infants will also resolve spontaneously in most cases.' Similarly, Document [8] notes that the calcaneovalgus foot (a different condition) resolves by 2 years with reassurance and passive stretching. The key point is that the deformity is not rigid and responds to passive manipulation (tickling corrects it), which supports a positional, non-pathologic origin.
Therefore, the most appropriate next step is reassurance, with monitoring and possibly passive stretching exercises, as the condition is expected to resolve spontaneously. No bracing, surgery, or arthrodesis is indicated at this stage.
Option A (foot abduction brace) is not indicated — bracing is not recommended for metatarsus adductus or intoeing in infants (Document [3]). Option B (arthrodesis) is invasive and inappropriate for a benign, self-resolving condition. Option D (tarsometatarsal capsulotomy) is a surgical procedure for a rigid deformity and not indicated in a condition that is passively correctable. Thus, the best choice is C: Reassurance.
answer_choice: C

Figure 21: Raw generator output for the representative MedQA example. The output shows that the final generator answers from the retrieved evidence and the original question rather than from the provisional hypothesis alone, and in this case correctly arrives at the gold answer after revising the initial guess. The text is reproduced verbatim except for formatting.