SFQ: A Sentence-level Framework for Medical Follow-up Questioning

Anonymous ACL submission

Abstract

Follow-up questioning is essential for safe medical decision-making but remains poorly evaluated in large language models (LLMs). In real clinical settings, patients provide incomplete information, requiring clinicians to actively elicit missing evidence before diagnosis. However, existing medical LLM benchmarks primarily evaluate diagnosis accuracy under complete or fixed case presentations, making it unclear whether models genuinely seek missing information. We propose *Sentence-level Follow-up Questioning (SFQ)*, a framework that formulates follow-up behavior as an explicit *sentence-level recall* task. Medical cases are decomposed into atomic clinical sentences, and incomplete cases are constructed by selectively hiding clinically relevant facts. Models must recover these facts through follow-up questions before committing to a diagnosis, enabling fine-grained and interpretable evaluation beyond diagnosis accuracy. Experiments on controlled sentence-abstracted benchmarks and a realistic external follow-up dataset show that recovering missing clinical information is associated with higher diagnostic accuracy. Recall-aware post-training improves both information recovery and diagnostic performance under incomplete inputs, demonstrating that explicitly modeling *follow-up questioning* can lead to safer and more reliable medical reasoning[1].

1 Introduction

Large language models (LLMs) have achieved strong performance in reasoning and text generation, particularly on question-answering tasks (Brown et al., 2020; Ouyang et al., 2022; Berger et al., 2025). However, many real-world applications are inherently interactive and require multi-turn conversations rather than single-shot responses (Chen et al., 2017; Biancofiore et al., 2024). In such settings, models must not only answer questions, but also identify missing information and actively elicit it through follow-up questions. This information-seeking capability is critical when user inputs are partial, ambiguous, or underspecified; without it, models are forced to rely on incomplete evidence, increasing the risk of overconfident or unreliable outputs.

The limitations of follow-up questioning are particularly serious in medical decision-making (Bender et al., 2021; Yu et al., 2025). In real clinical settings, patients rarely describe their conditions completely or clearly. Initial conversations are often incomplete, messy, and informal. Clinicians must ask targeted follow-up questions to gradually gather the information needed for diagnosis. Making diagnosis decisions without sufficient inquiry is unsafe in human medicine, and this risk is amplified when AI systems are used to support clinical decisions. Therefore, a competent medical assistant should first identify missing information and actively elicit it, rather than immediately producing a diagnosis.

Despite its importance, follow-up questioning remains a major weakness of current LLMs (Zhang et al., 2024; Meng et al., 2023; Huang et al., 2025b). Beyond architectural or inference-related limitations, a key issue lies in prevailing training paradigms. Most supervision is outcome-oriented, relying on single-turn question-answer pairs where all relevant information is already provided. In contrast, data that explicitly demonstrates how and when to ask effective follow-up questions is scarce. As a result, LLMs often fail to treat follow-up questioning as a distinct and necessary capability (Zhang et al., 2024; Huang et al., 2025b).

Although research on medical LLMs has grown rapidly, existing benchmarks and methods do not adequately evaluate follow-up questioning behavior (Liu et al., 2024; Yu et al., 2025). Most benchmarks assess diagnostic accuracy under complete

[1] Code will be released upon publication.

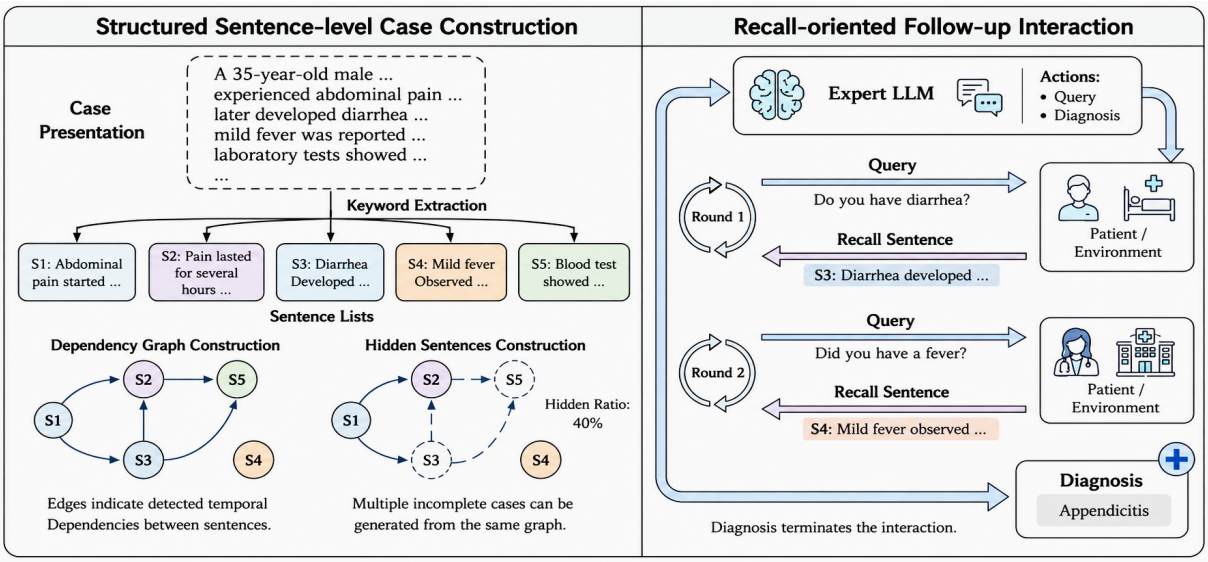

Figure 1: Our **sentence-level** formulation of medical follow-up questioning. Incomplete cases are constructed by selectively **hiding atomic clinical sentences** (left), and follow-up questioning is framed as a **recall-oriented process** (right). Diagnosis is treated as a terminal decision following multi-rounds informative interaction.

or fixed case presentations (Williams et al., 2024; Luk et al., 2024). Reasoning-enhanced or dialogue-based approaches may generate follow-up questions, but these are often implicit, prompt-driven, or external to the model's core behavior, making them difficult to control and reproduce (Wei et al., 2023; Gendron et al., 2024). More recently, follow-up questioning in medical reasoning has begun to receive more explicit attention and have achieved some satisfactory results. For example, Sun et al. (2026) propose KG-Followup, which augments LLMs with medical knowledge graphs and active in-context learning to generate clinically relevant follow-up questions. Wong et al. (2026) introduce MedClarify, an information-seeking diagnostic agent that selects case-specific follow-up questions by reducing uncertainty over differential diagnoses. However, these studies primarily focus on improving question generation or test-time diagnostic information seeking, while a controllable and fine-grained framework for evaluating whether models can actively recover missing clinical facts remains underexplored. Detailed discussions of these related work are provided in Appendix A.

To address this gap, we propose *Sentence-level Follow-up Questioning* (SFQ), a framework for defining and evaluating medical follow-up questioning using a sentence-level abstraction. Instead of treating medical cases as unstructured text, we decompose case presentations into *atomic clinical sentences*, each representing a single factual unit. As illustrated in Figure 1, this abstraction enables controllable construction of incomplete cases by selectively hiding subsets of sentences while preserving their causal or temporal relationships. Within this formulation, follow-up questioning is reframed as a *recall-oriented process*, requiring models to actively retrieve missing clinical facts through targeted queries before committing to a diagnosis.

Building on this formulation, we further optimize follow-up behavior through post-training. We apply supervised fine-tuning to expose models to examples of effective follow-up questions under incomplete information, and employ Group Relative Policy Optimization (GRPO, Shao et al. (2024)) to explicitly model follow-up questioning and diagnosis as distinct actions. Queries are rewarded for retrieving missing clinical facts, while redundant or uninformative questions are penalized, encouraging the model to balance continued inquiry with timely diagnosis.

We evaluate the proposed framework on sentence-abstracted medical cases and existing follow-up benchmarks, demonstrating consistent improvements in sentence-level recall and diagnostic performance under incomplete information. In summary, our contributions are as follows:

- We introduce a *sentence-level abstraction* of medical case presentations that enables controllable construction of incomplete cases and reframes follow-up questioning as a *recall-*

oriented evaluation problem beyond final diagnosis accuracy.

- We propose a post-training framework combining supervised fine-tuning and Group Relative Policy Optimization (GRPO) to explicitly optimize information-seeking behavior, modeling *follow-up questioning* and *diagnosis* as distinct actions. The model's ability of follow-up questioning has been improved after the training.

2 Framework

2.1 Sentence-level Case Abstraction

As illustrated in Figure 1 (left), real-world medical case presentations are typically provided as free-text descriptions, where multiple clinical facts—such as symptoms, temporal progression, and examination results—are interwoven in an unstructured manner. To enable controllable evaluation of follow-up questioning, we introduce a sentence-level abstraction that decomposes a case presentation into atomic and interpretable clinical units. Formally, a medical case is represented as:

$$\mathcal{C} = (P, Y), \tag{1}$$

where P denotes the original case presentation and Y is the ground-truth final diagnosis. Given P, we construct a set of sentence-level clinical facts

$$\mathcal{S} = \{s_1, s_2, ..., s_n\}, \tag{2}$$

where each sentence s_i corresponds to an atomic piece of clinically meaningful information, such as a symptom description, a temporal event, or a test finding. These sentence units are designed to be minimal and self-contained, serving as the basic elements for information recall.

The sentence set S is generated using an LLM with domain-specific prompting, which rewrites the original presentation into a list of concise, patient-style sentences while preserving the underlying clinical content. Compared to paragraph-level or dialogue-level abstractions, sentence-level decomposition provides finer granularity and clearer attribution of information, enabling explicit control over what clinical facts are observed or hidden.

Given the full sentence set S, we construct an **incomplete case** by selecting a subset $S_{\text{init}} \subset S$, while the remaining sentences $S_{\text{hid}} = S \backslash S_{\text{init}}$ represent clinically relevant but unobserved information to be elicited through follow-up questioning. By varying the ratio $|S_{\text{init}}|/|S|$, multiple incomplete cases can be generated from the same underlying presentation.

2.2 Sentence Dependency Graph

While sentence-level abstraction enables controllable information hiding, clinical sentences are not independent. Medical case presentations often contain temporal or causal dependencies between sentences, and ignoring such structure may result in incoherent or clinically implausible incomplete cases.

For example, consider the following two sentences:

- (1)*"I had abdominal pain yesterday."*
- (2)*"After several hours of abdominal pain, I developed diarrhea."*

Sentence (2) is temporally and causally dependent on sentence (1). If sentence (1) is observed while sentence (2) is hidden, the incomplete case remains coherent and reflects a realistic patient description. In contrast, hiding sentence (1) while exposing sentence (2) leads to an invalid setting, as the latter presupposes the former. This motivates a structure-aware hiding strategy rather than fully random sentence masking.

To capture such dependencies, we construct a **sentence dependency graph**

$$\mathcal{G} = (S, \mathcal{E}), \tag{3}$$

where each node corresponds to a sentence $s_i \in S$, and directed edges $(s_i \rightarrow s_j) \in \mathcal{E}$ indicate detected temporal or causal dependencies between sentences. The resulting graph is a directed acyclic graph (DAG), reflecting the progression and logical structure of the clinical narrative, as shown in Figure 1 (left). A simplified selection procedure is shown below as Algorithm 1.

By incorporating sentence dependencies into the hiding process, our framework generates incomplete cases that are both controllable and clinically consistent. This structure-aware construction forms the basis for meaningful evaluation of follow-up questioning behavior, which we describe next.

We further validate the proposed dependency-graph construction from two complementary perspectives: (*i*) logical effectiveness and (*ii*) structural consistency. Both evaluations compare DG-selected sentence subsets against randomly selected subsets of equal size (50% pruning ratio,

Algorithm 1 Sentences Dependency Graph

Require: Sentence graph $\mathcal{G} = (S, \mathcal{E})$, target ratio r

Ensure: Observed sentence set S_{init}

```
Initialize S_init ← ∅
Initialize queue Q with root nodes of G
while |S_init|/|S| < r and Q not empty do
    s ← Q.pop()
    Add s to S_init
    for each child c of s in G do
        Q.push(c)
    end for
end while
return S_init
```

half-sentences) on the 1,450 test cases in MedSentences.

2.2.1 Dependency Graph Analysis

Logical effectiveness. We first compare diagnostic accuracy when the LLM receives only the DG-selected half (`dg_half_sentences`) versus the randomly selected half (`random_half_sentences`). For each subset, an LLM produces top-3 diagnosis candidates and a judge model evaluates them against the ground truth.

Selection Method	Top-1 Acc.	Top-3 Acc.
Random (50%)	20.3%	34.8%
DG (50%)	22.4%	37.0%
Δ (DG − Random)	**2.1%**↑	**2.2%**↑

Table 1: Diagnostic accuracy comparison between DG-selected and randomly selected sentence subsets (50% pruning). Results evaluated using `GPT-5 mini` as the diagnosis and judge model.

Beyond accuracy, we validate whether DG edges reflect clinically meaningful dependencies. We sample dependency pairs and non-dependency control pairs from `dependency_pair`, then judge whether the relation $A \rightarrow B$ is clinically valid.

Pair Type	Judgment: YES (%)	Count
Dependency pairs	85.7%	83
Non-dependency pairs	3.5%	1767
Balanced accuracy	91.1%	

Table 2: Validation of DG-inferred dependency relationships. "YES" indicates the annotator confirmed that $A \rightarrow B$ is clinically valid.

In a supplementary run over 100 cases (`pair-count=50`), we evaluated 1,850 pairs in total. Given the class imbalance between dependency and non-dependency pairs, we focus on dependency sensitivity (85.7%) and non-dependency specificity (96.5%), which together yield a balanced accuracy of 91.1%. These results support the reliability of the dependency edges used in our construction.

Structural consistency. We quantify how well each selection method preserves graph structure using three metrics on the induced subgraph of selected sentences: root coverage, dependency conflict rate, and ancestor retention. Let $G = (V, E)$ be the full dependency graph, $V' \subseteq V$ be the selected nodes, and $G' = (V', E')$ be the induced subgraph:

$$\text{RootCov} = \frac{|\{v \in V' : \text{in-deg}_G(v) = 0\}|}{|\{v \in V : \text{in-deg}_G(v) = 0\}|}, \tag{4}$$

$$\text{DepConflict} = \frac{|\{(u, v) \in E : v \in V',\, u \notin V'\}|}{|\{(u, v) \in E : v \in V'\}|}, \tag{5}$$

$$\text{AncRetain} = \frac{1}{|V'|} \sum_{v \in V'} \frac{|\text{Anc}(v) \cap V'|}{|\text{Anc}(v)|}, \tag{6}$$

where $\text{Anc}(v)$ denotes the set of all ancestors of v in G.

Method	Root Cov. ↑	Dep. Conf. ↓	Anc. Ret. ↑
Random (50%)	0.491	0.427	0.579
DG (50%)	0.629	0.236	0.795
Δ	**0.137**↑	**-0.191**↓	**0.215**↑

Table 3: Structural comparison between DG-selected and random sentence subsets.

Overall, these results indicate that DG-based selection preserves diagnostically informative and structurally coherent sentence subsets better than random masking.

2.3 Follow-up Interaction Protocol

As illustrated in Figure 1 (right), we model medical follow-up as a multi-round interaction process between an expert agent and a patient (or environment) agent. Following prior work on interactive and multi-agent diagnosis systems, we adopt a follow-up interaction protocol and adapt it to our sentence-level case abstraction, enabling explicit modeling and evaluation of information recall during diagnosis. Under this protocol, the expert agent interacts with the patient agent over multiple rounds. At each round t, the expert agent selects

one of two actions,

$$a_t \in \{\text{query}, \text{diagnosis}\}, \tag{7}$$

where the `query` action corresponds to asking a follow-up question to acquire missing clinical information, and the `diagnosis` action outputs a final diagnostic prediction and terminates the interaction.

The patient agent has access to the full sentence set S and responds to each query by returning the subset of hidden sentences that semantically match the question, if any; otherwise, it returns an empty response. The patient agent only reveals facts present in the original case presentation, ensuring that follow-up questioning corresponds to recalling existing sentence-level information.

The interaction proceeds until the expert agent outputs a diagnosis. The final diagnosis is evaluated against the ground-truth label Y, while recalled sentences are recorded throughout the interaction. This protocol grounds follow-up questioning in a sentence-level information space, enabling explicit analysis of information-seeking behavior under incomplete clinical evidence.

3 Model Optimization

3.1 Supervised Fine-tuning

We first apply supervised fine-tuning (SFT) to teach the model what to ask given an incomplete case presentation. Under the proposed sentence-level framework, SFT is formulated as a conditional generation task that maps observed sentences to follow-up questions targeting hidden clinical facts.

Formally, for each case with sentence set S, we construct an incomplete case by selecting an observed subset $S_{\text{init}} \subset S$ and a hidden subset $S_{\text{hid}} = S \setminus S_{\text{init}}$. During data preprocessing, each hidden sentence $s \in S_{\text{hid}}$ is paired with a canonical follow-up question q_s that would elicit the corresponding information. The supervision signal for a case is thus defined as the set of target questions:

$$Q_{\text{hid}} = \{q_s \mid s \in S_{\text{hid}}\}. \tag{8}$$

Given the observed sentences S_{obs} as input, the model is trained to generate the corresponding follow-up questions Q_{hid}. As illustrated in Figure 1, SFT is implemented as a single-round generation process without interaction, where the model outputs a sequence (or list) of follow-up questions conditioned on the incomplete case. This formulation avoids reliance on multi-turn dialogue during training and provides a stable learning signal aligned with sentence-level information recall.

We fine-tune a pretrained large language model using standard maximum likelihood estimation, minimizing the negative log-likelihood of the target questions given the observed sentences:

$$\mathcal{L}_{\text{SFT}} = -\sum_{q \in Q_{\text{hid}}} \log p_\theta\big(q \mid S_{\text{init}}\big). \tag{9}$$

During training, loss is computed only over the generated follow-up questions, while the input context is masked to prevent spurious supervision. SFT equips the model with a basic understanding of which follow-up questions are clinically relevant under incomplete information. However, it does not explicitly model when to ask questions or when to stop and make a diagnosis. We address these limitations by introducing reinforcement-based optimization in the next section.

3.2 Reinforcement Optimization with GRPO

While supervised fine-tuning teaches the model what follow-up questions are relevant, it does not explicitly optimize the sequential decision process of interactive diagnosis (when to ask, what to ask next, and when to stop). We therefore apply Group Relative Policy Optimization (GRPO) (Shao et al., 2024) on top of SFT under the proposed follow-up interaction protocol.

In GRPO, each interaction trajectory receives recall-aware query rewards and a terminal diagnosis reward, so optimization directly favors informative questioning and timely, correct termination. Compared with SFT alone, this stage improves action-level decision quality under incomplete clinical evidence. Full formulation details (state/action definition, return function, relative-return optimization, and training configuration) are provided in Appendix E.

4 Experimental Setup

4.1 Datasets

We evaluate our framework on two medical follow-up benchmarks that reflect complementary interaction settings.

Med-Sentences We construct a sentence-level follow-up benchmark by transforming cases from *MedCaseReasoning* (Wu et al., 2025) into atomic clinical sentences as described in Section 2. Given

Model	Med-Sentences (Quarter)		Med-Sentences (Half)		Follow-up Text	
	IRR	HKR	IRR	HKR	IRR	HKR
GPT-5 mini	28.89%	23.1%	53.3%	37.9%	73.1%	61.2%
DeepSeek-V3.2	27.59%	22.5%	54.2%	38.1%	73.3%	63.8%
Llama3.1-8B	27.38%	22.0%	51.9%	37.4%	72.5%	60.6%
Qwen-2.5-7B-Instruct	27.42%	21.6%	53.1%	37.6%	72.3%	60.5%
Ours (SFT)	**44.30%**	**27.8%**	57.2%	40.3%	76.8%	67.7%
Ours (SFT+GRPO)	42.80%	27.0%	**57.7%**	**40.8%**	**77.8%**	**67.9%**

Table 4: Sentence-level information recall rate (IRR) and hit keyword rate (HKR) on sentence-abstracted medical cases with different observation ratios and on follow-up text. Our models (SFT and SFT+GRPO) are post-trained on `Qwen2.5-7B-Instruct`. The results clearly demonstrate that although current LLMs generally do not possess the ability to actively conduct follow-up questioning, with the appropriate training data for post-training, this capability can be enhanced.

a case presentation represented as a sentence set S, we generate incomplete cases by selectively hiding a subset $S_{\text{hid}} \subset S$, while exposing the remaining sentences S_{init}. We consider two incompleteness ratios: *Quarter* (25%) and *Half* (50%). Each hidden sentence is associated with a canonical follow-up question generated during preprocessing, forming a controllable and reproducible evaluation setting for follow-up questioning.

Follow-up Text To assess external validity, we additionally evaluate on the dataset introduced by (Li et al., 2023). This dataset provides an initial case description followed by expert-defined follow-up texts (e.g., additional examinations or findings), which serve as a natural form of information supplementation. Unlike Med-Sentences, this benchmark does not explicitly define sentence-level hiding, but reflects realistic follow-up information acquisition in clinical diagnosis.

Together, these two datasets allow us to evaluate both *controlled sentence-level recall* and realistic *follow-up information utilization*.

4.2 Models

We evaluate a range of representative large language models, including `Qwen2.5-7B-Instruct` (`Qwen2.5-7B`), `GPT-5`, `GPT-5 mini`, `Llama3.1-8B` and `DeepSeek-V3.2` (Qwen et al., 2025; Singh et al., 2026; Grattafiori et al., 2024; DeepSeek-AI et al., 2025). We report results for our fine-tuned models `Ours (SFT)` and `Ours (SFT+GRPO)`. All models follow the same interaction protocol and prompt format to ensure fair comparison.

4.3 Interaction Protocol

All evaluations follow the recall-oriented follow-up interaction protocol described in Section 2.3. At each turn, the model selects one of two actions: (1) issuing a follow-up query to request additional information, or (2) producing a final diagnosis, which terminates the interaction. For Med-Sentences, the environment returns hidden sentences whose content matches the model's query; for Follow-up Text, relevant follow-up descriptions are returned. A maximum number of interaction rounds is enforced to prevent unbounded querying.

4.4 Evaluation Metrics

We evaluate models from both information acquisition and diagnostic performance perspectives.

Information Recall Rate (IRR). IRR measures the proportion of case information recovered during interaction:

$$\text{IRR} = \frac{|S_{\text{init}} \cup \hat{S}_{\text{rec}}|}{|S|}, \tag{10}$$

where $\hat{S}_{\text{rec}}$ denotes the set of sentences recalled through follow-up questioning.

Hit Keyword Rate (HKR). To assess the quality of recalled information, we define HKR as the fraction of diagnosis-relevant keywords covered by recalled sentences. Keywords are generated during preprocessing by LLMs with access to the ground-truth diagnosis:

$$\text{HKR} = \frac{|\text{KW}(S_{\text{init}} \cup \hat{S}_{\text{rec}})|}{|\text{KW}(S)|}. \tag{11}$$

Diagnosis Accuracy (Acc). We report Top-1 and Top-3 diagnostic accuracy. Due to the inherent variability in medical diagnosis phrasing, we adopt a judge-based evaluation scheme using `Qwen2.5-7B` and `GPT-5` as diagnosis judges to determine semantic equivalence between predicted and ground-truth diagnoses (see LLM-as-a-judge prompt in Appendix G).

5 Results

5.1 Overall Performance

Table 4 shows the main results on Med-Sentences (Quarter / Half) and Follow-up Text. Across all settings, our approach consistently improves sentence-level information acquisition while maintaining or improving diagnostic performance.

On *Med-Sentences (Quarter)*, our SFT model achieves an IRR of 44.30%, substantially outperforming all base models, indicating a stronger ability to actively recover hidden clinical information under highly incomplete inputs. Incorporating GRPO further stabilizes performance, yielding comparable IRR (42.80%) while maintaining competitive HKR. Similar trends are observed under the Half setting, where our models achieve the highest IRR (57.7%) and HKR (40.8%), demonstrating robustness as case completeness increases.

On the *Follow-up Text benchmark*, which reflects more realistic follow-up information acquisition, our GRPO-optimized model achieves the strongest overall performance, with IRR reaching 77.8% and HKR 67.9%, outperforming both general-purpose and domain-specific baselines. These results suggest that explicit optimization for follow-up questioning generalizes beyond controlled sentence-level settings.

5.2 Follow-up Recall on Diagnostic Accuracy

Tables 5-7 present diagnosis accuracy evaluated under different recall agents and judge models. Across all datasets, introducing an explicit recall agent consistently improves Top-1 and Top-3 diagnostic accuracy.

Med-Sentences (Half)			
Judge	Recall Agent	Top-1	Top-3
Qwen2.5 -7B	None	9.96%	16.8%
	Qwen2.5-7B	10.7%	17.2%
	GPT-5 mini	10.8%	17.2%
	Ours (SFT+GRPO)	**12.3%**	**21.4%**
GPT-5	None	22.2%	37.1%
	Qwen2.5-7B	24.1%	38.2%
	GPT-5 mini	24.7%	38.7%
	Ours (SFT+GRPO)	**26.1%**	**41.2%**

Table 5: Top-1 and Top-3 diagnostic accuracy with different recall agents on Med-Sentences (Half). Specifically, the *recall agent* acts as an Expert LLM (as shown on the right side of Figure 1), extracting follow-up information from the incomplete input. The *judge model* makes the final diagnosis based on the information recalled by the recall agent.

On Med-Sentences (Half) (Table 5), when evaluated by a GPT-5 judge, diagnosis accuracy improves from 22.2% (no recall agent) to 26.1% with our GRPO-trained recall agent, with Top-3 accuracy increasing to 41.2%. Similar gains are observed when using `Qwen2.5-7B` as the judge, indicating that improvements are not judge-specific.

Med-Sentences (Quarter)			
Judge	Recall Agent	Top-1	Top-3
Qwen2.5 -7B	None	8.45%	14.1%
	Qwen2.5-7B	8.83%	14.4%
	GPT-5 mini	8.75%	15.2%
	Ours (SFT+GRPO)	**9.84%**	**17.2%**
GPT-5	None	13.3%	22.9%
	Qwen2.5-7B	15.1%	25.7%
	GPT-5 mini	15.4%	26.3%
	Ours (SFT+GRPO)	**19.6%**	**31.7%**

Table 6: Top-1 and Top-3 diagnostic accuracy with different recall agents on Med-Sentences (Quarter).

Results on Med-Sentences (Quarter) (Table 6) further highlight the importance of follow-up under more severe information sparsity. Our model achieves the highest Top-1 accuracy (9.84% under `Qwen2.5-7B` and 19.6% under `GPT-5`), outperforming baselines by a clear margin, despite the increased difficulty of the setting.

Follow-up Text			
Judge	Recall Agent	Top-1	Top-3
Qwen2.5 -7B	None	16.1%	22.5%
	Qwen2.5-7B	16.3%	22.9%
	GPT-5 mini	18.1%	25.1%
	Ours (SFT+GRPO)	**18.9%**	**26.8%**
GPT-5	None	31.5%	40.6%
	Qwen2.5-7B	32.8%	40.9%
	GPT-5 mini	33.3%	42.1%
	Ours (SFT+GRPO)	**36.1%**	**45.8%**

Table 7: Top-1 and Top-3 diagnostic accuracy with different recall agents on Follow-up Text, evaluated using `Qwen2.5-7B` and `GPT-5` as diagnosis judges.

On Follow-up Text (Table 7), the benefits of recall-oriented interaction are most pronounced. Under GPT-5 evaluation, our GRPO-trained model reaches 36.1% Top-1 accuracy and 45.8% Top-3 accuracy, compared to 31.5% / 40.6% without follow-up reasoning. This confirms that improved information recall directly translates into more reliable final diagnoses in realistic follow-up scenarios.

Overall, these results demonstrate that:

- Explicitly modeling follow-up questioning as a recall-oriented process improves information acquisition (IRR, HKR).
- Improvements in sentence-level recall consistently lead to higher diagnostic accuracy

across datasets and judges.

- Reinforcement optimization with GRPO further enhances robustness and generalization, particularly in realistic follow-up settings.

Together, these findings validate sentence-level recall as a meaningful and actionable evaluation signal for medical follow-up questioning.

5.3 Ablation Experiments and Analysis

We further perform ablation experiments to isolate the contribution of key design choices. The ablation suite contains two parts: (1) **key-factor ablations**, including *w/o DG*, *w/o keyword*, and *w/o sentence abstraction*; and (2) **interaction-budget analysis**, which varies the maximum number of follow-up rounds.

Key-factor ablations. We evaluate two model backbones (`GPT-5 mini` and `Qwen2.5-7B`) under the same training and evaluation protocol.

Model	Setting	Top-1	Top-3
GPT-5 mini	Full method	22.4%	37.0%
	w/o DG	20.3%	34.8%
	w/o keyword	22.0%	36.2%
	w/o sentence abstraction	17.3%	31.1%
Qwen2.5-7B	Full method	10.0%	16.8%
	w/o DG	8.2%	12.5%
	w/o keyword	9.9%	16.8%
	w/o sentence abstraction	7.1%	11.1%

Table 8: Key-factor ablations evaluated by Top-1 and Top-3 diagnostic accuracy.

Removing DG consistently reduces performance on both models, indicating that structure-aware sentence selection improves the quality of observed evidence. Removing sentence abstraction causes the largest degradation, suggesting that atomic clinical decomposition is central to controllable follow-up learning. In contrast, removing keyword generation yields relatively small changes, implying that keyword signals are complementary but not the primary source of gain.

Interaction-budget analysis. To study budget sensitivity, we vary the maximum follow-up rounds $n \in \{0, 3, 5, 10, 15, 30\}$ on the first 100 Med-Sentences (Quarter) cases and report IRR/HKR trends.

Overall, increasing interaction rounds mainly improves information acquisition in the low-to-medium budget regime, while gains beyond $n = 10$ are limited. This supports using $n = 10$ as a practical trade-off between recall effectiveness and interaction cost.

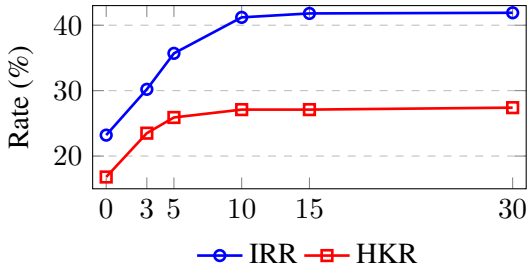

Figure 2: Interaction-budget analysis for `Our (SFT)` on Med-Sentences (Quarter). IRR and HKR improve with larger budgets and then show diminishing returns.

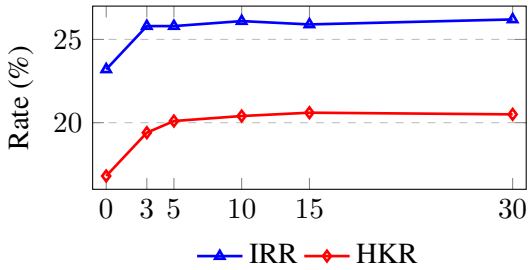

Figure 3: Interaction-budget analysis for `Qwen2.5-7B` on Med-Sentences (Quarter). Gains are smaller and quickly approach a plateau.

5.4 Case Study and Error Analysis

To better understand model behavior beyond aggregate metrics, we conduct a qualitative case study based on *a rare disease case* and our SFT model. Overall, we observe that effective sentence-level recall often leads to more informative follow-up interactions, but correct recall alone does not always guarantee correct diagnosis, particularly for rare diseases with overlapping clinical features. A detailed analysis of a representative case is provided in Appendix H.

6 Conclusion

Our results highlight the importance of explicitly modeling follow-up questioning as an information acquisition problem rather than treating it as a by-product of diagnosis-oriented reasoning. Across both controlled sentence-level benchmarks and realistic follow-up settings, we observe a consistent correlation between improved information recall and diagnostic accuracy, suggesting that diagnostic errors often stem from missing evidence rather than insufficient inference.

Limitations

This work has several limitations that point to directions for future research:

- First, while our framework provides a controlled and interpretable abstraction for evaluating follow-up questioning, it focuses on sentence-level factual information and does not aim to fully capture all aspects of real-world clinical decision-making.
- Second, our evaluation is conducted in a simulated interaction setting, which allows for systematic comparison but may differ from open-ended patient–clinician conversations in practice.
- Finally, we primarily study a limited set of model backbones and optimization strategies, and exploring broader model families and training paradigms may further improve follow-up reasoning capabilities.

References

Emily Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? pages 610–623.

Armin Berger, Sarthak Khanna, David Berghaus, and Rafet Sifa. 2025. Reasoning llms in the medical domain: A literature survey. *arXiv preprint arXiv:2508.19097*.

Giovanni Maria Biancofiore, Yashar Deldjoo, Tommaso Di Noia, Eugenio Di Sciascio, and Fedelucio Narducci. 2024. Interactive question answering systems: Literature review. *ACM Computing Surveys*, 56(9):1–38.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *ACM SIGKDD Explorations Newsletter*, 19(2):25–35.

Junying Chen, Chi Gui, Anningzhe Gao, Ke Ji, Xidong Wang, Xiang Wan, and Benyou Wang. 2024. Cod, towards an interpretable medical agent using chain of diagnosis. *Preprint*, arXiv:2407.13301.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. 2025. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Leon Garza, Anantaa Kotal, Michael A. Grasso, and Emre Umucu. 2025. Retrieval-augmented framework for llm-based clinical decision support. *Preprint*, arXiv:2510.01363.

Joseph Gatto, Parker Seegmiller, Timothy Burdick, Inas S. Khayal, Sarah DeLozier, and Sarah M. Preum. 2025. Follow-up question generation for enhanced patient-provider conversations. *Preprint*, arXiv:2503.17509.

Gaël Gendron, Qiming Bao, Michael Witbrock, and Gillian Dobbie. 2024. Large language models are not strong abstract reasoners. *Preprint*, arXiv:2305.19555.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Tenghao Huang, Sihao Chen, Muhao Chen, Jonathan May, Longqi Yang, Mengting Wan, and Pei Zhou. 2025a. Teaching language models to gather information proactively. *Preprint*, arXiv:2507.21389.

Youcheng Huang, Bowen Qin, Chen Huang, Duanyu Feng, Xi Yang, and Wenqiang Lei. 2025b. Beyond solving math quiz: Evaluating the ability of large reasoning models to ask for information. *Preprint*, arXiv:2508.11252.

Jian Li, Xi Chen, Weizhi Liu, Li Wang, Yingman Guo, Mingke You, Gang Chen, and Kang Li. 2023. One is not enough: Multi-agent conversation framework enhances rare disease diagnostic capabilities of large language models.

Shuyue Stella Li, Vidhisha Balachandran, Shangbin Feng, Jonathan S. Ilgen, Emma Pierson, Pang Wei Koh, and Yulia Tsvetkov. 2024. Mediq: Question-asking llms and a benchmark for reliable interactive clinical reasoning. *Preprint*, arXiv:2406.00922.

Mianxin Liu, Jinru Ding, Jie Xu, Weiguo Hu, Xiaoyang Li, Lifeng Zhu, Zhian Bai, Xiaoming Shi, Benyou Wang, Haitao Song, Pengfei Liu, Xiaofan Zhang, Shanshan Wang, Kang Li, Haofen Wang, Tong Ruan, Xuanjing Huang, Xin Sun, and Shaoting Zhang. 2024. Medbench: A comprehensive, standardized, and reliable benchmarking system for evaluating chinese medical large language models. *Preprint*, arXiv:2407.10990.

D. W. A. Luk, W. C. T. Ip, and Y. F. Shea. 2024. Performance of GPT-4 and GPT-3.5 in generating accurate and comprehensive diagnoses across medical subspecialties. *Journal of the Chinese Medical Association*, 87(3):259–260. Epub 2024 Feb 2.

Yan Meng, Liangming Pan, Yixin Cao, and Min-Yen Kan. 2023. FollowupQG: Towards information-seeking follow-up question generation. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 252–271, Nusa Dua, Bali. Association for Computational Linguistics.

Annika Meyer, Janik Riese, and Thomas Streichert. 2024. Comparison of the performance of gpt-3.5 and gpt-4 with that of medical students on the written german medical licensing examination: Observational study. *JMIR Medical Education*, 10.

George Michalopoulos, Michal Malyska, Nicola Sahar, Alexander Wong, and Helen Chen. 2022. Icdbigbird: A contextual embedding model for icd code classification. *Preprint*, arXiv:2204.10408.

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

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 467 others. 2026. Openai gpt-5 system card. *Preprint*, arXiv:2601.03267.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Scharli, Aakanksha Chowdhery, Philip Mansfield, Blaise Aguera y Arcas, Dale Webster, and 11 others. 2022. Large language models encode clinical knowledge. *Preprint*, arXiv:2212.13138.

Liwen Sun, Xiang Yu, Ming Tan, Zhuohao Chen, Anqi Cheng, Ashutosh Joshi, and Chenyan Xiong. 2026. Linking knowledge to care: Knowledge graph-augmented medical follow-up question generation. *Preprint*, arXiv:2603.01252.

Tao Tu, Anil Palepu, Mike Schaekermann, Khaled Saab, Jan Freyberg, Ryutaro Tanno, Amy Wang, Brenna Li, Mohamed Amin, Nenad Tomasev, Shekoofeh Azizi, Karan Singhal, Yong Cheng, Le Hou, Albert Webson, Kavita Kulkarni, S Sara Mahdavi, Christopher Semturs, Juraj Gottweis, and 6 others. 2024. Towards conversational diagnostic ai. *Preprint*, arXiv:2401.05654.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Christopher YK Williams, Brenda Y Miao, Aaron E Kornblith, and Atul J Butte. 2024. Evaluating the use of large language models to provide clinical recommendations in the emergency department. *Nature communications*, 15(1):8236.

Hui Min Wong, Philip Heesen, Pascal Janetzky, Martin Bendszus, and Stefan Feuerriegel. 2026. Medclarify: An information-seeking ai agent for medical diagnosis with case-specific follow-up questions. *Preprint*, arXiv:2602.17308.

Kevin Wu, Eric Wu, Rahul Thapa, Kevin Wei, Angela Zhang, Arvind Suresh, Jacqueline J. Tao, Min Woo Sun, Alejandro Lozano, and James Zou. 2025. Medcasereasoning: Evaluating and learning diagnostic reasoning from clinical case reports. *Preprint*, arXiv:2505.11733.

Archana Yadav, Harshvivek Kashid, Medchalimi Sruthi, B JayaPrakash, Chintalapalli Raja Kullayappa, Mandala Jagadeesh Reddy, and Pushpak Bhattacharyya. 2025. From recall to creation: Generating follow-up questions using bloom's taxonomy and Grice's maxims. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pages 1322–1338, Vienna, Austria. Association for Computational Linguistics.

E. Yu, X. Chu, W. Zhang, X. Meng, Y. Yang, X. Ji, and C. Wu. 2025. Large language models in medicine: Applications, challenges, and future directions. *International Journal of Medical Sciences*, 22(11):2792–2801.

Guangtao Zeng, Wenmian Yang, Zeqian Ju, Yue Yang, Sicheng Wang, Ruisi Zhang, Meng Zhou, Jiaqi Zeng, Xiangyu Dong, Ruoyu Zhang, Hongchao Fang, Penghui Zhu, Shu Chen, and Pengtao Xie. 2020. MedDialog: Large-scale medical dialogue datasets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9241–9250, Online. Association for Computational Linguistics.

Tong Zhang, Peixin Qin, Yang Deng, Chen Huang, Wenqiang Lei, Junhong Liu, Dingnan Jin, Hongru Liang, and Tat-Seng Chua. 2024. Clamber: A benchmark of identifying and clarifying ambiguous information needs in large language models. *Preprint*, arXiv:2405.12063.

Xuejiao Zhao, Siyan Liu, Su-Yin Yang, and Chunyan Miao. 2025a. Medrag: Enhancing retrieval-augmented generation with knowledge graph-elicited reasoning for healthcare copilot. *Preprint*, arXiv:2502.04413.

Yuze Zhao, Jintao Huang, Jinghan Hu, Xingjun Wang, Yunlin Mao, Daoze Zhang, Zeyinzi Jiang, Zhikai Wu, Baole Ai, Ang Wang, Wenmeng Zhou, and Yingda Chen. 2024a. Swift:a scalable lightweight infrastructure for fine-tuning. *Preprint*, arXiv:2408.05517.

Ziliang Zhao, Zhicheng Dou, and Yujia Zhou. 2024b. Generating intent-aware clarifying questions in conversational information retrieval systems. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 3384–3394, New York, NY, USA. Association for Computing Machinery.

Ziliang Zhao, Shiren Song, and Zhicheng Dou. 2025b. Followgpt: A framework of follow-up question generation for large language models via conversation log mining. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, CIKM '25, page 4412–4422, New York, NY, USA. Association for Computing Machinery.

Zhanke Zhou, Xiao Feng, Zhaocheng Zhu, Jiangchao Yao, Sanmi Koyejo, and Bo Han. 2025. From passive to active reasoning: Can large language models ask the right questions under incomplete information? *Preprint*, arXiv:2506.08295.

A Related Works

A.1 LLM Follow-up Questioning

Recent studies have explored follow-up questioning as a means for LLMs to acquire missing information under underspecified inputs. Meng et al. (2023) formalizes information-seeking follow-up question generation and introduces a benchmark emphasizing diverse questioning strategies, while Zhao et al. (2025b) leverages real user–LLM conversation logs to mine follow-up intents for more realistic question generation. Related work also studies structured forms of questioning, such as organizing follow-up questions according to Bloom's taxonomy (Yadav et al., 2025). Beyond explicit question generation, recent efforts investigate proactive information acquisition under incomplete problem settings, evaluating whether models choose to ask for missing information rather than directly answering (Huang et al., 2025b), and proposing training or benchmarking frameworks that encourage evidence gathering before decision making (Huang et al., 2025a; Zhou et al., 2025). Similar ideas appear in conversational information retrieval, where intent-aware clarification and iterative clarification–rewriting are used to progressively refine user queries (Zhao et al., 2024b).

In the medical domain, follow-up questioning has been studied to support patient–provider interactions and interactive clinical reasoning. Prior work proposes benchmarks and systems for question-asking in clinical settings (Li et al., 2024), while multi-agent conversational frameworks demonstrate that structured follow-up interactions can improve diagnostic performance in complex cases (Gatto et al., 2025; Li et al., 2023).

A.2 Medical Case Diagnosis Prediction

Medical case diagnosis prediction has been extensively studied with large language models, primarily under the assumption that complete or fixed case presentations are provided as input (Singhal et al., 2022; Meyer et al., 2024). Recent benchmarks evaluate diagnostic accuracy by mapping structured or semi-structured clinical descriptions to diagnoses, ICD codes, or clinical labels (Michalopoulos et al., 2022; Nori et al., 2023). Subsequent work incorporates reasoning-oriented techniques, such as chain-of-thought prompting, retrieval-augmented generation, or tool use, to improve diagnostic performance and robustness (Chen et al., 2024; Zhao et al., 2025a; Garza et al., 2025).

More recent studies explore interactive or dialogue-based medical assistants, allowing models to engage in multi-turn conversations with patients or simulated users to gather additional information (Zeng et al., 2020; Tu et al., 2024). While these approaches move closer to real clinical interactions, evaluations typically focus on final diagnostic outcomes or conversation-level success, with limited control over what information is missing or how it is acquired. As a result, the abilities of LLMs to actively seek and recall specific missing clinical facts remain less evaluated.

B Artifact License and Usage

We use publicly available research datasets and models following their respective licenses and terms of use. MedCaseReasoning and Follow-up Text are used for academic research purposes, and open-source model checkpoints are accessed according to their original distribution terms. Our released code and derived artifacts will include license information upon publication.

Sentence Construction Prompt

You are a clinical text processing assistant. Your task is to transform a medical case presentation into two outputs:
1. A list of extracted clinical keywords.
2. A list of short, patient-style sentences ("I have...", "I am...", "My ...")
where each sentence contains exactly ONE clinical fact.

TASK INSTRUCTIONS

Given an input case presentation, perform the following steps:

(1) KEYWORD EXTRACTION
- Extract minimal clinical units.
- Include demographics, duration, symptoms, characteristics, radiation, severity, aggravating/relieving factors, relevant history.
- Do NOT include function words or unnecessary phrases.
- Keep keywords short (1–5 words each).

(2) SENTENCE ATOMIZATION
- Convert the case into patient-style first-person statements.
- Each sentence must contain exactly ONE clinical fact.
- Use simple structures: "I am...", "I have...", "My symptoms..."
- No merging of multiple facts into one sentence, unless they share a core keyword.
- No extra explanations or new clinical information.
- If two sentences have a very clear sequence relationship, place them in one sentence.
- Minimize the number of sentences as much as possible

(3) Question Generation
- Generate a sample question for each sentence, focusing on the clinical fact it contains.
- For example, sentence "I have had a fever for 3 days." could generate the question "Do you have a fever and how long it takes?"
- Each question should be concise and directly related to the sentence.
- Each question should be in the same index with its corresponding sentence.

(4) JSON OUTPUT FORMAT
Return ONLY a JSON object with fields:
{
"keywords": ["...", "..."],
"sentences": ["...", "..."],
"questions": ["...", "..."]
}

INPUT CASE PRESENTATION

{CASE-TEXT}

C Sentence Construction and Details

C.1 Sentence Extraction Prompt

To enable controllable and fine-grained evaluation of follow-up questioning, we first transform free-text medical case presentations into a structured sentence-level representation. This process decomposes each case into atomic clinical sentences, associated keywords, and sentence-level follow-up questions, which together serve as the foundation for subsequent incomplete case construction and evaluation.

Given a raw case presentation C, we use a large language model to generate:

1. a set of extracted clinical keywords $\mathcal{K}$;
2. a list of atomic, patient-style sentences $\mathcal{S} = \{s_1, \ldots, s_n\}$, where each sentence contains exactly one clinical fact; and
3. a corresponding list of follow-up questions $\mathcal{Q} = \{q_1, \ldots, q_n\}$, each aligned with one sentence.

The transformation is performed using the following prompt:

C.2 Keyword Generation

For each medical case, clinical keywords are generated jointly with sentence construction as described in Appendix C.1. Keywords are intended to represent minimal semantic units corresponding to the core clinical facts expressed in each atomic sentence, including symptoms, temporal descriptors, and relevant contextual attributes.

During evaluation, keyword matching is used to compute the hit keyword rate (HKR), which measures whether information elicited through follow-up questioning captures clinically relevant concepts. Keyword matching is performed by comparing recalled sentences against the keyword set associated with the original case. Minor lexical variations are allowed through normalization and relaxed matching rules, while function words and non-clinical terms are ignored.

This keyword-based signal complements sentence-level recall by providing a coarse-grained measure of semantic coverage, without relying on exact sentence matching.

C.3 Dataset Statistics and Splits

Med-Sentences			
Type	%	Avg. Initial	Avg. Full
Quarter	25%	5.89	25.01
Half	50%	12.27	25.01

Table 9: The average length of sentences-list for different division methods.

We report basic statistics of the sentence-level abstraction in Table 9. The table summarizes the average number of observed sentences in the initial presentation and the total number of atomic sentences per case under different observation ratios (25% and 50%). These statistics reflect the granularity of sentence decomposition and ensure that incomplete cases are constructed with comparable levels of information across experimental settings.

For the sentence-abstracted version of the MedCase-Reasoning dataset, we randomly split the data into training, validation, and test sets with a ratio of 8:1:1. All models are trained on the training set, and final results are reported on the held-out test set.

D Recall-oriented Follow-up Interaction

D.1 Expert Prompt for Follow-up Questioning

In our recall-oriented interaction, the expert agent acts as a clinician who iteratively decides whether to ask a targeted follow-up question or to stop and output a final diagnosis. At each turn, the expert observes the currently known patient statements, the dialogue history, and the remaining turn budget, and must output exactly one action in a strict JSON format.

D.2 Patient / Environment Prompt

The patient (environment) agent serves as a non-generative information source. It holds the complete set of atomic clinical sentences and responds to the expert's follow-up questions by returning only matched sentences, without introducing new facts or performing additional reasoning.

Patient / Environment Prompt

You are a patient who knows a set of true statements about yourself.

You will receive:
- A list of ALL-SENTENCES, each describing one true fact about you.
- A doctor's question.

Your task is to answer ONLY based on ALL-SENTENCES.

Rules:
- If the question is clearly asking about one or more facts that are present in ALL-SENTENCES, you should return those sentences.
- If multiple sentences are relevant, you may return more than one.
- If there is no relevant information, answer that you don't have this information.
- Do NOT invent new facts.

You MUST output ONLY a JSON object with the following format:

If one or more sentences match the doctor's question:

```
{ "matched": true,
  "sentences": [ "<one matching sentence>",
                 "<another matching sentence>" ] }
```

If no sentence matches:

```
{ "matched": false,
  "sentences": [] }
```

Expert Prompt

You are an experienced clinician evaluating a patient case.
You will see:
- A list of patient statements that are currently KNOWN.
- A log of previous questions you asked and the patient's answers.
- The maximum number of turns you are allowed to use.

Your task is to decide, at each step, whether to:
1) ASK a focused follow-up question, or
2) DIAGNOSE the most likely primary diagnosis.

Rules:
- You MAY ONLY ask questions about potential symptoms, history, or risk factors that could plausibly appear as short patient-style sentences (e.g., "I have chest pain", "I have a fever", "I am a smoker", etc.).
- When you ASK, your question must have a single answer, such as: "Do you have chest tightness?", "What is your age?", "Do your symptoms radiate to your left arm?"
- When you DIAGNOSE, provide your best single main diagnosis.
- Be concise and clinically focused.
- Do NOT repeat the same question.
- Stop asking questions once you believe you have enough information to make diagnosis.

You MUST respond ONLY in valid JSON with the following format:

If you choose to ask a question:
`{ "action": "query", "question": "Do you have ... ?" }`

If you choose to provide a final diagnosis:
`{ "action": "diagnosis", "diagnosis": "<your final diagnosis>" }`

Do not include any other fields or free text.

D.3 Multi-round Interaction Protocol

Following prior work on interactive medical reasoning and multi-turn diagnosis, we adopt a structured multi-round interaction protocol within our sentence-level recall framework. Each episode proceeds as a sequence of turns between the expert agent and the patient (environment) agent.

At each turn t, the expert observes the currently known patient statements and the dialogue history, and outputs exactly one action: either a follow-up query or a final diagnosis. If a query is issued, the patient agent returns all matched sentences from the full sentence set, or an empty response if no relevant information is available. To ensure bounded interaction, the maximum number of rounds is fixed to $T = 10$ for all experiments. If no diagnosis is produced within T rounds, the expert is forced to output a diagnosis at the final step.

This protocol ensures that follow-up questioning is explicit, bounded, and fully observable, enabling precise measurement of information recall and its impact on downstream diagnostic decisions. For experiments involving local model inference, including `Qwen2.5-7B-Instruct`, `Llama3.1-8B`, and our post-trained models, all interactions are executed on a single NVIDIA A40 GPU.

E Optimization with GRPO

To further improve follow-up questioning behavior beyond supervised fine-tuning, we apply reinforcement optimization using Group Relative Policy Optimization (GRPO). This section provides the detailed formulation and implementation corresponding to the brief description in Section 3.2.

E.1 Policy and Trajectory Formulation

We formulate follow-up questioning as a sequential decision problem over interaction trajectories. At each round t, the expert agent observes the current state

$$s_t = \big(S_{\text{init}}^{(t)}, h_t\big), \tag{12}$$

where $S_{\text{init}}^{(t)}$ is the currently observed sentence set and h_t is the dialogue history. The policy samples an action

$$a_t \sim \pi_\theta(a_t \mid s_t), \quad a_t \in \{\texttt{query}, \texttt{diagnosis}\}. \tag{13}$$

When $a_t = \texttt{query}$, the model issues a follow-up question and receives matched hidden sentences (if any); when $a_t = \texttt{diagnosis}$, the episode terminates with predicted diagnosis $\hat{Y}$.

The total return of a trajectory $\tau = (s_1, a_1, \ldots, s_T, a_T)$ is

$$R(\tau) = \sum_{t=1}^{T-1} r_{\text{query}}(s_t, a_t) + r_{\text{diag}}(s_T, a_T), \tag{14}$$

where r_{query} and r_{diag} denote query-level and terminal diagnosis rewards.

E.2 Action and Episode Definition

Each training sample corresponds to a static incomplete case constructed at the sentence level. An episode consists of a bounded sequence of expert actions

$$a_t \in \{\texttt{query}, \texttt{diagnosis}\}, \quad t = 1, \ldots, T, \tag{15}$$

where $T = 10$ is the maximum number of allowed rounds. The episode terminates when a diagnosis action is produced or when the round limit is reached, in which case a diagnosis is forced.

E.3 Recall-aware Reward Design

We design a recall-aware reward function that explicitly balances information acquisition efficiency and diagnostic correctness.

1. Query reward. For a follow-up query issued at round t, the reward depends on whether the query successfully recalls at least one hidden sentence:

$$r_{\text{query}}(t) = \begin{cases} \dfrac{\alpha}{1+\lambda t}, & \text{if the query matches,} \\ -\beta(1+\lambda t), & \text{otherwise,} \end{cases} \tag{16}$$

where $\alpha > 0$ and $\beta > 0$ control the relative reward and penalty, and λ governs time-dependent decay. This formulation encourages early, informative questions while discouraging prolonged or unproductive querying.

2. Diagnosis reward. When a diagnosis action is produced, a terminal reward is assigned:

$$r_{\text{diag}} = \begin{cases} +\gamma, & \text{if the diagnosis is correct,} \\ -\gamma, & \text{otherwise,} \end{cases} \tag{17}$$

where γ is set to dominate individual query rewards, incentivizing the model to stop querying once sufficient information has been gathered.

E.4 Relative-return Optimization in GRPO

After supervised fine-tuning, GRPO is applied as a lightweight reinforcement step. For each initial case, multiple trajectories are sampled and compared by relative return; trajectories with better information-seeking and diagnosis outcomes receive stronger policy updates. This relative comparison mechanism improves question selection and stopping decisions without requiring a separate value model.

E.5 GRPO Training Details

GRPO Training Configuration

```
train_type = lora
lora_rank = 8
lora_alpha = 16
learning_rate = 2e-5
max_length = 2048
num_train_epochs = 5
per_device_train_batch_size = 1
gradient_accumulation_steps = 8
eval_strategy = steps
eval_steps = 50
logging_steps = 20
num_generations = 4
```

GRPO training is initialized from the supervised fine-tuned (SFT) model described in Section 4.1, with `Qwen2.5-7B-Instruct` serving as the base model.

All GRPO training is conducted using the **ms-swift** framework (Zhao et al., 2024a) with LoRA-based parameter-efficient updates. We use 1,400 sentence-abstracted Med-Sentences training cases for reinforcement learning. Training is executed on a single NVIDIA A100 GPU. Unless otherwise specified, other optimization hyperparameters follow the default GRPO configuration provided by ms-swift.

F Supervised Fine-tuning Details

F.1 Training Configuration

All supervised fine-tuning experiments are conducted on top of the `Qwen2.5-7B-Instruct` base model. We fine-tune the model using a standard causal language modeling objective on the constructed sentence-level follow-up question data. Training is performed on a single NVIDIA A40 GPU.

We adopt a small per-device batch size combined with gradient accumulation to balance memory efficiency and optimization stability. Mixed-precision training with bfloat16 is enabled, together with gradient checkpointing, to reduce memory consumption. Model checkpoints are saved periodically, and only a limited number of recent checkpoints are retained.

The key training hyperparameters are summarized in the configuration box below.

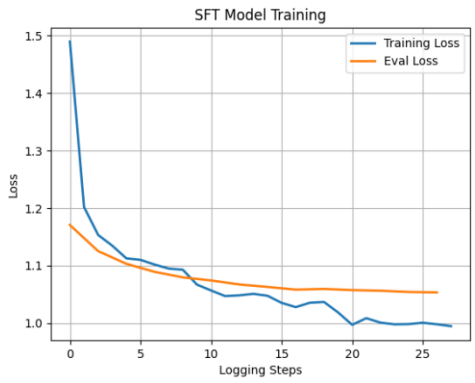

Figure 4: The training loss curve and the evaluation loss curve in the supervised fine-tuning (SFT) method

Supervised Fine-tuning Configuration

```
CONFIG =
per_device_train_batch_size = 2
per_device_eval_batch_size = 2
gradient_accumulation_steps = 8
num_train_epochs = 3
learning_rate = 5e-5
logging_steps = 50
eval_strategy = "steps"
eval_steps = 100
bf16 = True
gradient_checkpointing = True
```

F.2 Training Performance

Figure 4 illustrates the training and evaluation loss curves during supervised fine-tuning. The model is trained for three epochs, and both training and evaluation losses decrease steadily over the course of optimization.

The training loss shows a rapid decline in the early stage, followed by a more gradual convergence, indicating stable optimization behavior. The evaluation loss closely tracks the training loss without divergence, suggesting limited overfitting under the chosen training configuration. Overall, the loss trends demonstrate that the supervised fine-tuning process converges smoothly and yields a well-trained initialization for subsequent reinforcement optimization.

G LLM-based Diagnosis Evaluation

Final diagnosis correctness is evaluated using an *LLM-as-a-Judge* (Wu et al., 2025) protocol to account for synonymous expressions and minor lexical variations in free-form diagnostic outputs. Specifically, we use `GPT-5 mini` as a lightweight judge model that receives the predicted diagnosis and the ground-truth label and outputs a binary decision (yes or no) indicating whether they refer to the same primary diagnosis. The judge is instructed to be concise and does not generate explanations. This evaluation protocol is applied consistently across all models and experimental settings and is used solely for evaluation, not for training or inference.

LLM-as-a-Judge

Is this predicted final diagnosis correct (yes/no)?
Predicted final diagnosis: `{prediction}`
True final diagnosis: `{truth}`
Answer [yes/no].

We emphasize that the judge model is only used for evaluation and does not participate in training or inference of the proposed models. Since the same judge and prompt are applied consistently, relative comparisons across models remain fair and controlled.

H Detailed Case Study

H.1 Case Overview

Case Overview

- **Case ID:** PMC8561223
- **Ground-truth diagnosis:** Pituicytoma
- **Predicted diagnosis:** Cystic pituitary adenoma
- **Recall rate:** 23.1% → 61.5%
- **Category:** Correct follow-up, incorrect diagnosis

H.2 Initial Observed and Hidden Sentences

Initial Observed and Hidden Sentences

Initially Observed Sentences (S_{init}):

- MRI of my brain showed a sellar cystic lesion ...
- The cyst was only partially removed ...
- My headaches worsened after a thunderclap episode ...

Hidden Sentences (S_{hid}):

- I am a 33-year-old soldier ...
- I have had intermittent headaches for the ...
- I had no focal neurologic deficits ...
- The cyst wall enhanced with gadolinium ...
- I had low thyroid-stimulating hormone ...
- I have low testosterone ...
- I did not have diabetes insipidus ...
- An endoscopic transsphenoidal approach was ...
- A yellow capsule released clear, straw-colored ...
- The pulsating cyst capsule was tightly adherent ...

H.3 Follow-up Interaction (Dialog)

Follow-up Interaction (Dialog)

Doctor Query: What type of surgery did you undergo and under what anesthesia?
Patient Answer: I don't have this information.

Doctor Query: What symptoms do you have related to vision?
Patient Answer: I have had intermittent headaches for the past year ...
Patient Answer: My headaches worsened after a thunderclap episode ...
Patient Answer: MRI of my brain showed a sellar cystic lesion ...

Doctor Query: What was found during the surgery regarding the cystic lesion?
Patient Answer: An endoscopic transsphenoidal approach was undertaken ...
Patient Answer: A yellow capsule released clear, straw-colored fluid ...
Patient Answer: The pulsating cyst capsule was tightly adherent ...
Patient Answer: The cyst was only partially removed ...

Doctor Query: What were the findings from the MRI of your brain?
Patient Answer: MRI of my brain showed a sellar cystic lesion ...
Patient Answer: The cyst wall enhanced with gadolinium ...

Final Diagnosis: *Cystic pituitary adenoma*

H.4 Analysis and Discussion

This case illustrates that the model can effectively recover a substantial portion of diagnostically relevant information through follow-up questioning. However, despite high-quality recall, the final diagnosis remains incorrect due to the rarity of the true condition and its substantial overlap with more common pituitary lesions. This suggests that while sentence-level recall provides a meaningful signal for follow-up ability, diagnosis accuracy remains constrained by *prior biases* and *disease prevalence*.