GraphMed-LT: Patient-Specific Graph Memory with Latent Clinical Thought Refinement for Multi-Turn Medical Conversations

Anonymous ACL submission

Abstract

Multi-turn medical question answering (QA) aims to model realistic clinical diagnosis, where a doctor gathers patient information across multiple turns of conversation. Existing multi-turn medical conversation systems have shown promising progress, but they often rely on accumulated conversation histories as memory, leaving clinical evidence fragmented across turns. We propose **GraphMed-LT**, a patient-specific graph memory approach with latent clinical thought refinement for multi-turn medical conversations. GraphMed-LT extracts patient-specific clinical triplets from patient responses, retrieves relevant knowledge triplets, and organises them into an incrementally updated graph memory. The graph memory is projected into graph-conditioned evidence tokens and refined inside a trainable doctor agent through hidden-state feedback, enabling the agent to update its internal clinical context before asking follow-up questions or producing the final answer. Experiments on three multi-turn medical QA benchmarks show that GraphMed-LT consistently outperforms existing multi-turn medical conversation baselines across multiple LLM backbones, achieving up to a 6.3 percentage-point absolute improvement over the strongest baseline. Further analyses show that GraphMed-LT asks more answerable follow-up questions and provides consistent gains across medical specialties.[1]

1 Introduction

Traditional medical question answering (QA) is typically formulated as a single-turn task, where all patient information is provided at once (Jin et al., 2019; Pal et al., 2022). As illustrated in Fig. 1, this differs from multi-turn medical conversations, where patient information is collected progressively through doctor–patient interaction. Although existing approaches have achieved strong performance in single-turn QA, they do not fully reflect real-world clinical practice, where diagnoses are formed through iterative questioning and progressive evidence accumulation (Hendrycks et al., 2021). A central challenge in multi-turn medical conversations is that clinically relevant evidence is distributed across turns, requiring effective integration for reliable multi-hop reasoning (Jin et al., 2024; Wang et al., 2023). This motivates structured mechanisms for organising and integrating clinical evidence acquired across turns.

Existing methods for multi-turn medical reasoning have achieved promising performance by optimising information-seeking behaviour, enabling doctor agents to ask follow-up questions and acquire missing patient evidence before making a decision (Hu et al., 2024; Chen et al., 2025; Jia et al., 2025). MEDIQ (Li et al., 2024) established a doctor–patient conversation setting in which a doctor agent actively queries a patient simulator and reasons over accumulated conversation logs. Building on this interactive setting, ProMed (Ding et al., 2025) further encouraged proactive questioning through reinforcement learning with Shapley Information Gain (Winter, 2002) rewards. MEDDxAgent (Rose et al., 2025) introduced a modular differential-diagnosis agent that combines history taking and knowledge retrieval for interactive clinical reasoning. These methods demonstrate the value of active information seeking in multi-turn medical conversations. However, patient evidence is still largely maintained as dialogue history rather than organised into a structured cross-turn memory. Therefore, clinical facts acquired in separate turns remain fragmented, limiting the multi-turn dependencies required for effective multi-hop reasoning.

To address this challenge, we propose GraphMed-LT, a patient-specific graph memory approach with latent clinical thought refinement for multi-turn medical conversations. GraphMed-LT

[1]The source code for GraphMed-LT is publicly available at https://anonymous.4open.science/r/GraphMed-LT-D52C.

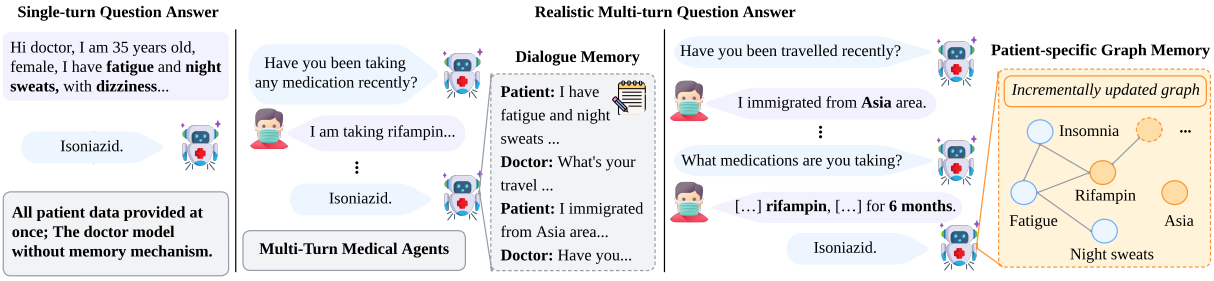

Figure 1: **Single-turn QA vs. multi-turn QA. Left:** In single-turn medical QA, all patient information is provided at once without asking follow-up questions. **Middle:** In multi-turn medical QA, patient information is collected progressively and stored as dialogue history. **Right:** GraphMed-LT organises the acquired patient information into a patient-specific graph memory with latent clinical thought refinement for multi-turn reasoning.

extracts patient-specific triplets from patient responses, retrieves relevant knowledge triplets, and organises them into an incrementally updated graph memory. Knowledge triplets provide a suitable representation for complex biomedical knowledge by decomposing clinical evidence into explicit entity–relation–entity structures, making the underlying reasoning chains easier to retrieve, trace, and verify than in free-form text. By accumulating these triplets in a patient-specific graph memory, GraphMed-LT connects evidence acquired across turns rather than treating it as a dialogue history. The graph memory is projected into graph-conditioned evidence tokens, which are in turn injected into a trainable doctor large language model (LLM) and refined through continuous-space hidden-state feedback before follow-up questioning or final answer prediction.

To assess whether this design improves both final diagnosis and the intermediate information-seeking process, we evaluate GraphMed-LT on three multi-turn medical conversation datasets: iCRAFT-MD, iMedQA, and iMedMCQA (Johri et al., 2023, 2024; Pal et al., 2022). Experimental results show that GraphMed-LT consistently outperforms seven strong baselines across multiple LLM backbones. Further analyses show that GraphMed-LT asks more answerable follow-up questions and provides consistent gains across medical specialties. These findings indicate that combining patient-specific graph memory with latent clinical thought refinement enables more reliable multi-hop reasoning in multi-turn medical conversations. Our contributions can be summarised as follows:

- We propose GraphMed-LT, a patient-specific graph memory approach that incrementally organises extracted patient triplets and retrieved knowledge triplets for multi-turn clinical evidence modelling.
- We introduce latent clinical thought refinement, a continuous-space reasoning mechanism that refines graph-conditioned evidence tokens inside a trainable doctor agent through hidden-state feedback.
- We conduct extensive experiments across three multi-turn medical conversation datasets, showing that GraphMed-LT consistently outperforms strong baselines across multiple LLM backbones.

2 Methodology

In the following, we first define the task and introduce GraphMed-LT, which consists of a patient agent (Sec. 2.2), triplet agent (Sec. 2.3), patient-specific graph memory (Sec. 2.4), and a trainable doctor agent (Sec. 2.5). Fig. 2 shows how GraphMed-LT organises patient evidence across turns into graph-conditioned evidence tokens for latent clinical thought refinement in the doctor agent.

2.1 Task Definition

Multi-turn medical conversations simulate an iterative interaction between a doctor agent and a patient agent. The doctor agent is initially given a question, a set of candidate answers $\mathcal{A} = \{a_1, \ldots, a_C\}$, and an initial patient description p_0. The patient agent has access to the complete patient record $\mathcal{P} = \{p_0, p_1, \ldots, p_n\}$ and reveals relevant information in response to the doctor's follow-up questions. At each turn t, the doctor agent either asks a follow-up question q_t or outputs a final answer $\hat{a} \in \mathcal{A}$. Given q_t, the patient agent returns a response r_t based on $\mathcal{P}$, and the observed conversation history is updated as follows:

$$\mathcal{C}_t = \mathcal{C}_{t-1} \cup \{(q_t, r_t)\}. \quad (1)$$

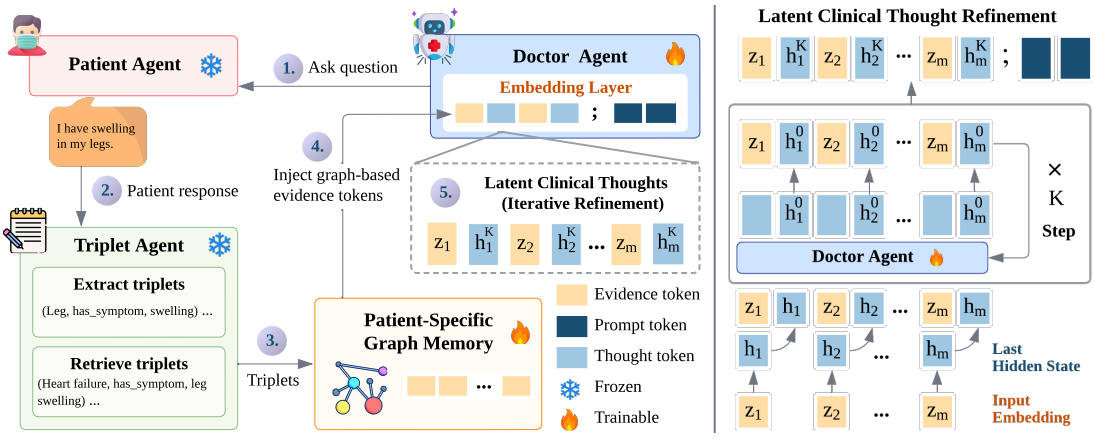

Figure 2: **Left:** The overview of GraphMed-LT follows steps 1–5: the doctor asks a follow-up question, the patient responds, the triplet agent extracts and retrieves triplets, and the patient-specific graph memory projects them into graph-based evidence tokens for the doctor agent. **Right:** (1) graph-based evidence tokens z_i are paired with thought tokens h_i; (2) during the K-step loop, z_i remains fixed while h_i is updated through hidden-state feedback; (3) the final sequence $[z_1, h_1^K, \ldots, z_m, h_m^K]$ is concatenated with prompt for follow-up questioning or answer prediction.

The interaction terminates when the doctor agent outputs a final answer or reaches the maximum number of turns. The goal is to select the correct answer by integrating patient evidence accumulated over multiple turns.

2.2 The Patient Agent

The patient agent simulates the patient in a multi-turn medical conversation. Given the doctor agent's follow-up question q_t and the complete patient record $\mathcal{P}$, it returns a response r_t grounded in the available patient information. The patient agent does not perform diagnosis or decision-making; it only reveals patient information in response to the doctor's questions. To avoid unsupported clinical facts, the patient agent is constrained to answer using information contained in the patient record. If no relevant sentence exists for the given query, it returns a fixed response: "*The patient cannot answer this question, please do not ask this question again*", indicating that the question cannot be answered. This constraint ensures that subsequent triplet extraction is based on dataset-supported patient evidence rather than speculative content, and also provides the basis for measuring answerable question rate. The prompt for the patient agent is provided in Appendix B.

2.3 Triplet Agent

The triplet agent converts patient responses into structured clinical triplets and retrieves relevant knowledge triplets. Given the patient response r_t at turn t, it extracts patient-specific triplets that are explicitly expressed in the response. In addition, the triplet agent retrieves relevant knowledge triplets from PrimeKG (Chandak et al., 2023), a broad-coverage biomedical knowledge graph, using similarity matching. Specifically, the patient response and candidate knowledge triplets are encoded with the biomedical contrastive encoder BiCA (Sinha et al., 2026), and the top-3 most relevant triplets are retained as auxiliary background knowledge. The extracted and retrieved triplets are used to update the patient-specific graph memory. Details of the triplet extraction and retrieval are provided in Appendix A.3.

2.4 Patient-Specific Graph Memory

The patient-specific graph memory module incrementally organises triplets acquired across conversation turns into a patient-specific graph, encodes the graph with a graph attention network (GAT) (Veličković et al., 2017), and projects the resulting representation into graph-conditioned evidence tokens for the doctor agent.

Graph Construction. The graph memory is initialised as $\mathcal{G}_0$ using triplets extracted from the initial patient description p_0. At each turn t, the triplet agent provides a triplet set $\mathcal{T}_t$. The graph memory is updated from $\mathcal{G}_{t-1}$ to $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ by adding the clinical entities and relation-labelled edges in $\mathcal{T}_t$. Each node in $\mathcal{V}_t$ represents a clinical entity, and each edge in $\mathcal{E}_t$ represents a relation-labelled triplet. Source-aware edge types distinguish patient-observed triplets from retrieved knowledge. Details of graph construction are provided in Appendix A.4.

Graph Encoder. The graph encoder converts the patient-specific graph memory into a graph-level representation. Clinical entities are initialised using BiCA embeddings computed from their textual forms. We then use a GAT to propagate information over the graph and aggregate the final node representations into a graph-level representation:

$$\mathbf{g}_t = \mathrm{Pool}\left(\mathrm{GAT}_{\theta_{\mathrm{graph}}}\left(\mathcal{G}_t\right)\right), \tag{2}$$

where θ_{graph} denotes the trainable parameters of the graph encoder and $\mathrm{Pool}(\cdot)$ denotes attention pooling over node representations. Details of the node initialisation, GAT propagation, and graph-level pooling are provided in Appendix A.4.

Projector. The graph-level representation $\mathbf{g}_t$ is mapped into the input embedding space of the doctor agent through a trainable graph-to-token projector:

$$\mathbf{Z}_t = \mathrm{Proj}_{\theta_{\mathrm{proj}}}\left(\mathbf{g}_t\right) = [z_1, \ldots, z_m] \in \mathbb{R}^{m \times d}, \tag{3}$$

where θ_{proj} denotes the trainable projector parameters, m is the number of graph-conditioned evidence tokens, and d is the hidden dimension of the doctor agent. These evidence tokens summarise the current patient-specific graph memory and serve as fixed evidence anchors during latent clinical thought refinement.

2.5 Doctor Agent

The doctor agent is a trainable LLM for follow-up questioning and answer prediction. At turn t, the doctor prompt embedding $\mathbf{P}_t$ contains the question, candidate answers, observed conversation history $\mathcal{C}_{t-1}$, and task instruction. The graph-conditioned evidence tokens $\mathbf{Z}_t = [z_1, \ldots, z_m]$ produced by the patient-specific graph memory module are incorporated into the doctor agent through latent clinical thought refinement.

Latent Clinical Thought Refinement. Using graph-conditioned evidence tokens only as a static prefix may not be sufficient for the doctor agent to internalise the patient-specific graph memory, because the graph evidence is read only once before generation and may not be iteratively reconciled with the evolving clinical context. We therefore introduce latent clinical thought refinement, where graph-conditioned evidence tokens serve as fixed anchors and thought tokens serve as mutable reasoning states. Specifically, each evidence token z_i is paired with one thought token h_i^k, providing a local refinement state for each graph-derived evidence anchor while keeping the refinement length fixed. At refinement step k, the input sequence is constructed as follows:

$$\mathbf{S}^k = [z_1, h_1^k, z_2, h_2^k, \ldots, z_m, h_m^k]. \tag{4}$$

The doctor agent processes this interleaved sequence and produces hidden states at all positions. Only the hidden states at the thought-token positions are used to update the thought tokens for the next refinement step, while the evidence tokens remain unchanged within the refinement loop as graph anchors. This process is repeated for K refinement steps, yielding the final graph-conditioned clinical context $\mathbf{S}^K$. The sequence length remains fixed during the refinement because each iteration replaces the previous thought states rather than appending new tokens.

The refined clinical context is prepended to the doctor prompt:

$$\mathbf{X}_t = [\mathbf{S}^K; \mathbf{P}_t]. \tag{5}$$

Given $\mathbf{X}_t$, the doctor agent either asks another follow-up question or outputs a final answer. If a follow-up question is asked, the patient response is processed by the triplet agent and used to update the graph memory for the next turn. If a final answer is produced, the interaction terminates.

Training Objective. GraphMed-LT is trained with answer prediction supervision after graph encoding, graph-to-token projection, and latent clinical thought refinement. The trainable parameters are $\Theta = \{\theta_{\mathrm{graph}}, \theta_{\mathrm{proj}}, \theta_{\mathrm{doc}}\}$, corresponding to the graph encoder, projector, and doctor agent. Latent clinical thought refinement is implemented through the hidden-state feedback procedure of the doctor agent and does not introduce a separate parameter set. Given the final input $\mathbf{X}_t$, the model predicts a distribution $\hat{\mathbf{y}}$ over candidate answers and is optimised with the cross-entropy loss:

$$\mathcal{L}_{\mathrm{ans}}(\Theta) = -\sum_{c=1}^{C} y_c \log \hat{y}_c. \tag{6}$$

The patient agent and triplet agent are frozen and are not optimised by this loss, serving only as auxiliary interaction and evidence-extraction modules. Additional details of the hidden-state refinement procedure are provided in Appendix A.5.

Method	iCRAFT-MD				iMedQA				iMedMCQA			
	L8	Q7	L70	Q72	L8	Q7	L70	Q72	L8	Q7	L70	Q72
UoT	45.5	45.1	63.0	62.7	40.0	39.8	54.2	55.0	41.7	41.5	55.5	56.1
CoD	46.2	45.6	63.8	63.5	40.7	40.2	55.0	55.6	42.4	42.1	56.2	56.8
DDO	46.0	45.9	64.1	63.4	41.2	40.5	55.6	56.1	42.1	42.4	56.8	57.3
MEDIQ	50.0	49.4	72.1	71.6	45.0	44.8	60.9	61.5	47.5	47.0	62.0	62.7
MedAgentSim	51.8	50.7	71.8	72.4	46.6	46.5	62.8	63.1	49.2	48.6	63.8	64.0
MEDDxAgent	52.7	52.2	73.7	72.9	48.6	47.8	64.1	65.2	50.4	49.9	65.4	65.7
ProMed	55.2	54.0	74.6	74.1	50.3	49.7	66.8	67.5	52.0	51.2	67.9	68.6
GraphMed-LT	**61.5**†	**60.2**†	**78.0**†	**79.1**†	**56.4**†	**55.8**†	**72.7**†	**73.0**†	**57.8**†	**57.0**†	**71.2**†	**72.0**†

Table 1: Accuracy (%) comparison with baselines across datasets and doctor backbones. L8, Q7, L70, and Q72 denote LLaMA-3-8B, Qwen-2.5-7B, LLaMA-3-70B, and Qwen-2.5-72B, respectively. Best results are shown in **bold**. † indicates a significant improvement over the strongest baseline (paired t-test, $p < 0.05$).

3 Experimental Setup

Datasets We evaluate GraphMed-LT on three multi-turn medical conversation datasets. iMedQA (Li et al., 2024) is used for training and in-domain testing, with 10,178 training cases, 1,272 validation cases, and 1,273 test cases. iCRAFT-MD (Johri et al., 2024) is a dermatology-focused zero-shot test set with 140 clinically realistic patient–doctor interactions. iMedMCQA is derived from MedMCQA (Pal et al., 2022) by converting 200 questions into multi-turn conversations, and is used as an additional zero-shot test set. Details of the datasets are provided in Appendix A.6.

Baselines We compare GraphMed-LT with seven multi-turn medical conversation baselines: UoT (Hu et al., 2024), CoD (Chen et al., 2025), DDO (Jia et al., 2025), MEDIQ (Li et al., 2024), MedAgentSim (Almansoori et al., 2025), MEDDxAgent (Rose et al., 2025), and ProMed (Ding et al., 2025).All baselines use the same patient agent, candidate answer set, turn budget, stopping rule, conversation-history access, and answer-selection protocol. For DDO, MedAgentSim, and MEDDxAgent, which were not originally designed for candidate-answer selection, we preserve their information-seeking strategy and adapt only the final prediction interface to the given candidates. This keeps the interaction protocol identical while comparing how each method reasons over acquired evidence. Details are provided in Appendix A.7.

Evaluation We use diagnostic accuracy, measuring whether the doctor agent selects the correct answer after the multi-turn conversation. In addition, we report interaction efficiency, measured by the average number of follow-up questions asked before termination. To assess question answerability, we report the answerable question rate (AQR), defined as the proportion of follow-up questions that can be answered under the patient-record-grounded simulator protocol.

Implementation Details All experiments follow the same multi-turn medical conversation protocol. The patient agent and triplet agent are both instantiated with Qwen-2.5-72B for stable instruction-following, and kept frozen. The patient agent is shared across all methods, ensuring that every method interacts with the same patient-response mechanism. The triplet agent is used by GraphMed-LT and its ablation variants; it is fixed so that differences among these variants reflect how the same structured evidence is organised and integrated. Additional details are provided in Appendix A.2.

4 Results

In this section, we answer five Research Questions (RQ) about GraphMed-LT: its overall effectiveness (Sec. 4.1), component contributions (Sec. 4.2), accuracy–efficiency trade-off (Sec. 4.3), robustness under different evidence settings (Sec. 4.4), and consistency across medical specialties (Sec. 4.5).

4.1 RQ1: How effective is GraphMed-LT for multi-turn medical conversations?

We first evaluate GraphMed-LT from two complementary perspectives: final diagnostic accuracy and question answerability. Diagnostic accuracy measures whether the doctor agent selects the correct answer after the multi-turn conversation, while AQR measures whether follow-up ques-

Method	iCRAFT-MD				iMedQA				iMedMCQA			
	L8	Q7	L70	Q72	L8	Q7	L70	Q72	L8	Q7	L70	Q72
Conversation History	50.0	49.4	72.1	71.6	45.0	44.8	60.9	61.5	47.5	47.0	62.0	62.7
Triplets as Text	53.7	53.1	72.4	72.6	49.8	48.9	63.2	64.5	51.0	50.6	62.9	63.6
w/o Retrieved Triplets	60.6	59.5	77.1	78.2	55.5	55.1	71.8	72.3	56.9	56.3	70.3	71.1
w/o Graph Memory	57.4	56.0	75.2	76.0	52.1	51.6	68.4	69.2	53.8	53.1	67.0	67.8
w/o Latent Thoughts	55.8	54.6	73.8	74.5	50.9	50.2	66.0	66.8	52.4	51.7	65.3	66.0
GraphMed-LT	**61.5**	**60.2**	**78.0**	**79.1**	**56.4**	**55.8**	**72.7**	**73.0**	**57.8**	**57.0**	**71.2**	**72.0**

Table 2: Ablation study across datasets and doctor backbones. L8, Q7, L70, and Q72 denote LLaMA-3-8B, Qwen-2.5-7B, LLaMA-3-70B, and Qwen-2.5-72B, respectively. Best results are shown in **bold**.

tions can be answered under our patient-record-grounded simulator protocol. As shown in Table 1, GraphMed-LT achieves the best accuracy in all twelve dataset–backbone settings. Compared with the strongest baseline in each setting, GraphMed-LT provides consistent gains of 3.3–6.3 percentage points. For example, on iCRAFT-MD, GraphMed-LT improves accuracy over ProMed from 55.2 to 61.5 with LLaMA-3-8B and from 74.1 to 79.1 with Qwen-2.5-72B. On iMedQA, GraphMed-LT reaches 73.0 with Qwen-2.5-72B, compared with 67.5 for ProMed. On iMedMCQA, GraphMed-LT also achieves the highest accuracy, reaching 72.0 with Qwen-2.5-72B compared with 68.6 for the strongest baseline.

Fig. 3 further reports AQR using Qwen-2.5-72B as a representative strong doctor backbone, while Table 1 shows that the accuracy trend is consistent across all evaluated backbones. GraphMed-LT obtains 62.9 AQR on iCRAFT-MD, 60.4 on iMedQA, and 61.8 on iMedMCQA, improving over the strongest baseline by 5.1, 4.4, and 5.1 percentage points, respectively. This indicates that GraphMed-LT asks follow-up questions that are more often answerable under the benchmark protocol, rather than improving accuracy through indiscriminate information seeking. Together, the accuracy and AQR results across the three datasets suggest that GraphMed-LT improves both final answer selection and protocol-level information seeking in multi-turn medical conversations.

4.2 RQ2: How does each component affect GraphMed-LT?

We conduct component ablations to examine whether GraphMed-LT's gains come from structured triplet preprocessing, patient-specific graph memory, retrieved knowledge, or latent clinical thought refinement. To isolate upstream triplet preprocessing from evidence integration, all triplet-based variants reuse the same extracted triplets and differ only in how they are organised and injected into the doctor agent; in particular, *w/o Graph Memory* keeps the same extracted and retrieved triplets but removes graph construction and GAT propagation, aggregating triplet representations into a non-graph evidence representation before projection. As shown in Table 2, *Conversation History*, which provides the accumulated dialogue history directly to the doctor agent without triplet extraction or graph memory, performs weakest in most settings, while *Triplets as Text* improves over it, suggesting that structured clinical facts provide a clearer evidence interface than raw dialogue history. Removing latent thoughts causes the largest accuracy drop, from 73.0 to 66.8 on iMedQA with Qwen-2.5-72B, showing that hidden-state refinement is central to internalising graph-conditioned evidence. Removing graph memory leads to the second-largest accuracy drop, from 73.0 to 69.2, indicating that graph-based organisation is more effective than non-graph evidence aggregation for connecting patient evidence across turns. Removing the retrieved triplets causes a smaller accuracy

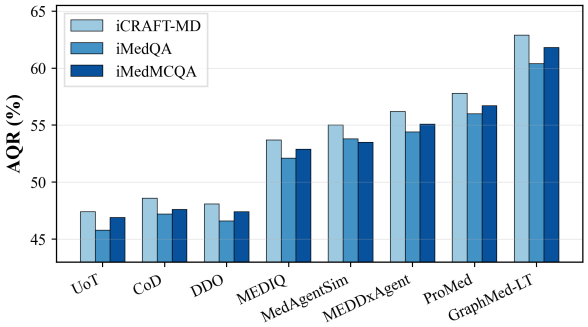

Figure 3: Answerable question rate (%) comparison with multi-turn medical conversation baselines across datasets. Qwen-2.5-72B is used as the doctor backbone.

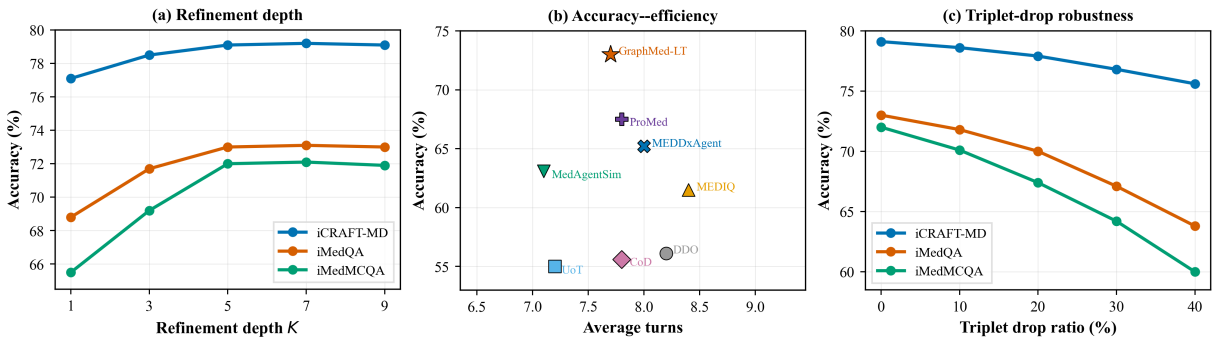

Figure 4: **(a)** Accuracy under different latent refinement depths. **(b)** Accuracy–efficiency trade-off on iMedQA. **(c)** Robustness to incomplete triplet extraction.

decrease, from 73.0 to 72.3, suggesting that retrieved knowledge is useful but not the primary source of improvement. Overall, these ablations show that GraphMed-LT's gains are not solely due to triplet preprocessing; the main improvements come from latent clinical thought refinement and patient-specific graph memory.

4.3 RQ3: Does GraphMed-LT improve the accuracy–efficiency trade-off?

We further examine whether GraphMed-LT improves diagnostic accuracy without substantially increasing the number of interaction turns. We conduct this analysis on iMedQA with Qwen-2.5-72B as a representative setting, and report accuracy against the average number of follow-up questions. As shown in Fig. 4(b), GraphMed-LT achieves 73.0 accuracy with 7.7 average turns, compared with 67.5 for ProMed at 7.8 turns and 65.2 for MEDDxAgent at 8.0 turns. Although MedAgentSim uses slightly fewer turns, its accuracy is substantially lower than GraphMed-LT. These results show that GraphMed-LT achieves higher accuracy than the strongest baselines while using a comparable number of follow-up questions. This suggests that our proposed approach does not necessarily rely on longer conversations; instead, patient-specific graph memory can help the doctor agent use acquired evidence more effectively within a compact interaction process.

4.4 RQ4: Is GraphMed-LT stable under different evidence settings?

We analyse GraphMed-LT's sensitivity to latent refinement depth and simulated triplet missingness, where extracted triplets are randomly dropped. As shown in Fig. 4(a), increasing the number of latent refinement steps K improves accuracy and then saturates: iMedQA rises from 68.8 at $K = 1$ to 73.0 at $K = 5$, iMedMCQA from 65.5 to 72.0, and iCRAFT-MD from 77.1 to 79.1, with little gain beyond $K = 5$. Fig. 4(c) shows that accuracy decreases as more triplets are dropped. With 20% of triplets removed, GraphMed-LT still reaches 77.9 on iCRAFT-MD, 70.0 on iMedQA, and 67.4 on iMedMCQA. These results support $K = 5$ as a stable and efficient choice and suggest that GraphMed-LT is robust to moderate triplet missingness.

4.5 RQ5: Are GraphMed-LT's gains consistent across medical specialties?

We further examine whether GraphMed-LT's gains are consistent across different clinical domains, represented by medical specialty groups within the benchmark setting. As shown in Fig. 5, GraphMed-LT achieves the highest accuracy across all eight specialties, ranging from 72.4 on Dermatology to 77.1 on Otolaryngology. Compared with representative baselines, GraphMed-LT improves consistently across specialties. For example, it improves over MEDDxAgent by 7.6 points on Genomics and 8.6 points on Dermatology, and over ProMed by 5.7 and 5.5 points on the same specialties. On average, GraphMed-LT improves over ProMed, the strongest overall baseline in Table 1, from 69.2 to 74.4, suggesting that its gains are not confined to a single specialty group.

5 Related Work

In the following, we describe key related work and position our contributions in the existing literature.

5.1 Multi-Turn Medical Conversation Agents

Multi-turn medical conversations require a doctor agent to collect patient information across dialogue turns and make diagnostic decisions from progressively acquired evidence (Rezaei et al., 2025; Zhu et al., 2025; Wang et al., 2025). Early automatic

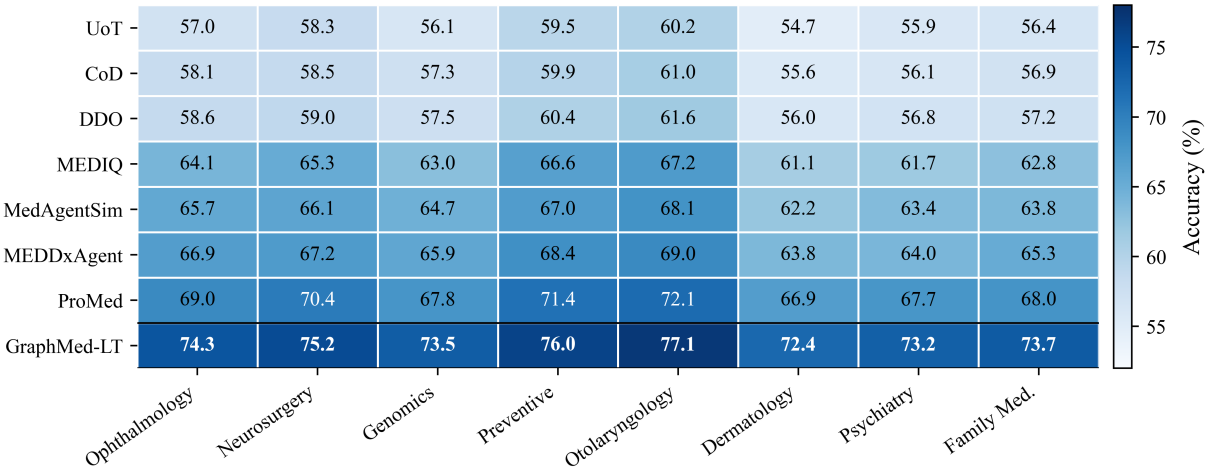

Figure 5: Accuracy (%) comparison across eight medical specialty groups on iMedQA.

diagnosis systems formulate this process as sequential decision-making, where the agent asks about additional symptoms before predicting the final disease. Recent generation-based models further improve symptom inquiry and diagnosis through sequence modelling and attention mechanisms. Diaformer (Chen et al., 2022) formulated automatic diagnosis as symptom sequence generation, MTDiag (Hou et al., 2023) jointly modelled symptom inquiry and disease prediction, and HAIformer (Zhao et al., 2024) incorporated human–AI collaboration for diagnostic decision-making. More recent LLM-based agents have improved active information seeking and diagnostic reasoning in multi-turn settings (Zhou et al., 2026; Feng et al., 2026). These methods show the value of question-asking medical agents, but patient evidence is still largely maintained as dialogue history for multi-turn reasoning. Our proposed GraphMed-LT allows cross-turn patient evidence to be explicitly organised and refined for doctor-agent reasoning.

5.2 Knowledge Integration Medical Models

Knowledge integration has become an important direction for improving medical QA, especially when answers require external evidence or multi-step reasoning (Varshney et al., 2023; Yu et al., 2025). MedRAG (Xiong et al., 2024) evaluated how medical corpora, retrievers, and LLM backbones affect evidence-enhanced medical QA. RAG^2 (Sohn et al., 2025) used model-generated rationales to guide biomedical evidence retrieval and filter noisy snippets. MedGraphRAG (Wu et al., 2025) organised medical documents into graph structures for evidence-grounded generation, while RGAR (Liang et al., 2025) incorporated patient-specific factual knowledge from electronic health records. Although these studies have achieved strong performance by retrieving external medical evidence, they mainly organise evidence for a given query. GraphMed-LT instead updates a patient-specific graph memory throughout the conversation and refines graph-conditioned evidence inside the doctor agent.

6 Conclusions

Multi-turn medical conversations require doctor agents to integrate patient evidence progressively acquired across dialogue turns. Existing methods often rely on accumulated conversation histories, leaving clinical evidence fragmented for multi-turn reasoning. We proposed GraphMed-LT, a novel patient-specific graph memory approach with latent clinical thought refinement. GraphMed-LT extracts patient-specific clinical triplets, retrieves auxiliary knowledge triplets, organises them into an incrementally updated graph memory, and refines graph-conditioned evidence tokens inside a trainable doctor agent through hidden-state feedback. Experiments on three multi-turn medical datasets show that GraphMed-LT consistently outperforms strong baselines across multiple LLM backbones, achieving up to a 6.3 percentage-point absolute improvement. Further analyses showed that GraphMed-LT asks more answerable follow-up questions, improves the accuracy–efficiency trade-off and provides consistent gains across medical specialties. These findings suggest that patient-specific graph memory with latent clinical thought refinement provides an effective mechanism for reliable multi-turn medical reasoning, thereby supporting more robust multi-turn medical conversations.

7 Limitations

This work has several limitations. (1) Our evaluation focuses on benchmark-style multi-turn medical conversations with candidate answers, which enables a controlled and reproducible comparison. Future work can extend GraphMed-LT to open-ended diagnostic generation, longitudinal patient histories, and clinician-authored conversations. (2) GraphMed-LT is evaluated with patient simulators constructed from structured patient records. While this setting ensures that patient responses are grounded in dataset-supported evidence, future studies can examine more natural patient interactions, including incomplete, ambiguous, or linguistically diverse responses. (3) Our current graph memory uses extracted patient-specific triplets and retrieved background triplets under a predefined relation schema. Future work can enrich this schema with broader clinical ontologies, uncertainty-aware extraction, and clinician-validated knowledge sources, further improving graph construction and evidence grounding in real-world medical settings.

References

Mohammad Almansoori, Komal Kumar, and Hisham Cholakkal. 2025. Medagentsim: Self-evolving multi-agent simulations for realistic clinical interactions. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 362–372. Springer.

Payal Chandak, Kexin Huang, and Marinka Zitnik. 2023. Building a knowledge graph to enable precision medicine. *Scientific data*, 10(1):67.

Junying Chen, Chi Gui, Anningzhe Gao, Ke Ji, Xidong Wang, Xiang Wan, and Benyou Wang. 2025. Cod, towards an interpretable medical agent using chain of diagnosis. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14345–14368.

Junying Chen, Dongfang Li, Qingcai Chen, Wenxiu Zhou, and Xin Liu. 2022. Diaformer: Automatic diagnosis via symptoms sequence generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 4432–4440.

Hongxin Ding, Baixiang Huang, Yue Fang, Weibin Liao, Xinke Jiang, Zheng Li, Junfeng Zhao, and Yasha Wang. 2025. Promed: Shapley information gain guided reinforcement learning for proactive medical llms. *arXiv preprint arXiv:2508.13514*, pages 1120–1129.

Yichun Feng, Jiawei Wang, Lu Zhou, Zhen Lei, and Yixue Li. 2026. Doctoragent-rl: A multi-agent collaborative reinforcement learning system for multi-turn clinical dialogue. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 16952–16956. IEEE.

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

Zhenyu Hou, Yukuo Cen, Ziding Liu, Dongxue Wu, Baoyan Wang, Xuanhe Li, Lei Hong, and Jie Tang. 2023. Mtdiag: an effective multi-task framework for automatic diagnosis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 14241–14248.

Zhiyuan Hu, Chumin Liu, Xidong Feng, Yilun Zhao, See-Kiong Ng, Anh T Luu, Junxian He, Pang W Koh, and Bryan Hooi. 2024. Uncertainty of thoughts: Uncertainty-aware planning enhances information seeking in llms. *Advances in Neural Information Processing Systems*, 37:24181–24215.

Zhihao Jia, Mingyi Jia, Junwen Duan, and Jianxin Wang. 2025. Ddo: Dual-decision optimization for llm-based medical consultation via multi-agent collaboration. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 26380–26397.

Bowen Jin, Gang Liu, Chi Han, Meng Jiang, Heng Ji, and Jiawei Han. 2024. Large language models on graphs: A comprehensive survey. *IEEE Transactions on Knowledge and Data Engineering*, 36(12):8622–8642.

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

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Shreya Johri, Jaehwan Jeong, Benjamin A Tran, Daniel I Schlessinger, Shannon Wongvibulsin, Zhuo Ran Cai, Roxana Daneshjou, and Pranav Rajpurkar. 2023. Testing the limits of language models: a conversational framework for medical ai assessment. *medRxiv*.

Shreya Johri, Jaehwan Jeong, Benjamin A Tran, Daniel I Schlessinger, Shannon Wongvibulsin, Zhuo Ran Cai, Roxana Daneshjou, and Pranav Rajpurkar. 2024. Craft-md: A conversational evaluation framework

for comprehensive assessment of clinical llms. In *AAAI 2024 Spring Symposium on Clinical Foundation 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. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, pages 1120–1129.

Sichu Liang, Linhai Zhang, Hongyu Zhu, Wenwen Wang, Yulan He, and Deyu Zhou. 2025. RGAR: Recurrence generation-augmented retrieval for factual-aware medical question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, Suzhou, China. Association for Computational Linguistics.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR.

Mohammad Reza Rezaei, Reza Saadati Fard, Jayson Lee Parker, Rahul G Krishnan, and Milad Lankarany. 2025. Agentic medical knowledge graphs enhance medical question answering: Bridging the gap between LLMs and evolving medical knowledge. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 12682–12701, Suzhou, China. Association for Computational Linguistics.

Daniel Philip Rose, Chia-Chien Hung, Marco Lepri, Israa Alqassem, Kiril Gashteovski, and Carolin Lawrence. 2025. Meddxagent: A unified modular agent framework for explainable automatic differential diagnosis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13803–13826.

Aarush Sinha, Roshan Balaji, Nirav Pravinbhai Bhatt, and 1 others. 2026. Bica: Effective biomedical dense retrieval with citation-aware hard negatives. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 33010–33018.

Jiwoong Sohn, Yein Park, Chanwoong Yoon, Sihyeon Park, Hyeon Hwang, Mujeen Sung, Hyunjae Kim, and Jaewoo Kang. 2025. Rationale-guided retrieval augmented generation for medical question answering. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 12739–12753.

Deeksha Varshney, Aizan Zafar, Niranshu Kumar Behera, and Asif Ekbal. 2023. Knowledge grounded medical dialogue generation using augmented graphs. *Scientific Reports*, 13(1):3310.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv*.

Heng Wang, Shangbin Feng, Tianxing He, Zhaoxuan Tan, Xiaochuang Han, and Yulia Tsvetkov. 2023. Can language models solve graph problems in natural language? *Advances in Neural Information Processing Systems*, 36:30840–30861.

Qipeng Wang, Rui Sheng, Yafei Li, Huamin Qu, Yushi Sun, and Min Zhu. 2025. Medkgi: Iterative differential diagnosis with medical knowledge graphs and information-guided inquiring. *arXiv preprint arXiv:2512.24181*.

Eyal Winter. 2002. The shapley value. *Handbook of game theory with economic applications*, 3:2025–2054.

Junde Wu, Jiayuan Zhu, Yunli Qi, Jingkun Chen, Min Xu, Filippo Menolascina, Yueming Jin, and Vicente Grau. 2025. Medical graph rag: Evidence-based medical large language model via graph retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28443–28467.

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

Yuechun Yu, Han Ying, Haoan Jin, Wenjian Jiang, Dong Xian, Binghao Wang, Zhou Yang, and Mengyue Wu. 2025. Medkgeval: A knowledge graph-based multi-turn evaluation framework for open-ended patient interactions with clinical llms. *arXiv preprint arXiv:2510.12224*.

Xuehan Zhao, Jiaqi Liu, Yao Zhang, Zhiwen Yu, and Bin Guo. 2024. Haiformer: Human-ai collaboration framework for disease diagnosis via doctor-enhanced transformer. In *ECAI 2024: 27th European Conference on Artificial Intelligence, 19–24 October 2024, Santiago de Compostela, Spain–Including 13th Conference on Prestigious Applications of Intelligent Systems (PAIS 2024)*, pages 1495–1502. SAGE Publications Pvt. Ltd 1 Oliver's Yard, 55 City Road, London, EC1Y 1SP.

Yang Zhou, Zhenting Sheng, Mingrui Tan, Yuting Song, Jun Zhou, Yu Heng Kwan, Lian Leng Low, Yang Bai, and Yong Liu. 2026. Note2chat: Improving llms for multi-turn clinical history taking using medical notes. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Jiayuan Zhu, Jiazhen Pan, Yuyuan Liu, Fenglin Liu, and Junde Wu. 2025. Ask patients with patience: Enabling llms for human-centric medical dialogue with grounded reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2846–2857.

A Appendix

A.1 Computing Hardware

All experiments were conducted on GPU nodes equipped with 8 NVIDIA GH200 GPUs. These resources provide sufficient memory and computational capacity to support large-scale model inference and training under the experimental settings considered in this work.

A.2 Implementation Details

Hyperparameter sensitivity. We further vary the number of retrieved triplets and the number of graph-conditioned evidence tokens. As shown in Fig. 6 and Fig. 7, top-3 retrieval and $m = 20$ evidence tokens provide a strong and compact setting for the main experiments. We select hyperparameters using iMedQA development-set accuracy and keep the selected settings fixed for iMedQA test, iCRAFT-MD, and iMedMCQA. During evaluation, all methods use the same decoding setting, with temperature 0.2, top-$p = 0.9$.

Follow-up question generation and stopping. For all methods, the doctor agent generates follow-up questions under the same prompt format and decoding setting. We do not use a predefined question pool; instead, each method generates its own follow-up questions during the conversation. A conversation stops once the doctor agent gives a final answer or reaches the maximum budget of 10 turns. The final answer is always selected from the given candidate answers.

For GraphMed-LT, the triplet agent retrieves the top-3 knowledge triplets at each turn. The graph-to-token projector produces $m = 20$ graph-conditioned evidence tokens from the current patient-specific graph. Each evidence token is paired with one latent thought token, and the refinement loop is run for $K = 5$ steps unless otherwise specified. Table 3 lists the full setting.

Frozen agent configuration and quality checks. GraphMed-LT uses two frozen auxiliary agents. The patient agent is shared across all methods and answers follow-up questions only from atomic facts derived from the complete patient record; unsupported questions trigger a fixed unanswerable response. This keeps patient-response generation identical across methods and prevents unsupported clinical information from entering the conversation. The triplet agent is used only by GraphMed-LT and its ablation variants. It extracts patient-specific triplets from the current patient response and retrieves auxiliary knowledge triplets from PrimeKG using BiCA-based similarity, without performing answer prediction or diagnostic reasoning.

We manually inspect 100 randomly sampled iMedQA development examples to estimate the quality of the frozen auxiliary agents. As shown in Table 4, 98.0% of extracted triplets are supported by the patient response, 78.0% of retrieved triplets are clinically relevant to the response, and 89.0% of unanswerable cases correctly trigger the fixed unanswerable response. These results suggest that the auxiliary agents provide reliable but non-oracle

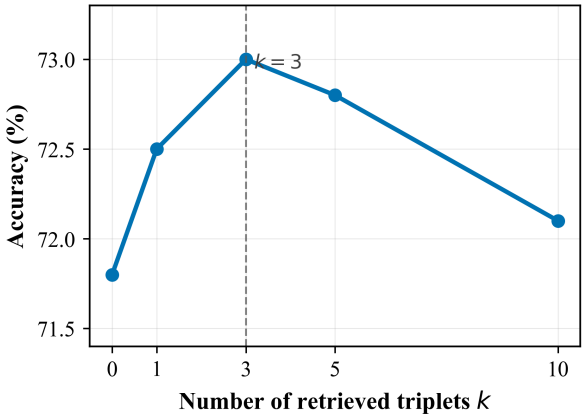

Figure 6: **Effect of retrieved knowledge triplets.** We vary the number of retrieved knowledge triplets k on iMedQA using Qwen-2.5-72B as the doctor backbone. Accuracy improves when a small number of knowledge triplets is retrieved, peaks at $k = 3$, and decreases when more triplets are added, suggesting that excessive retrieval can introduce irrelevant background knowledge.

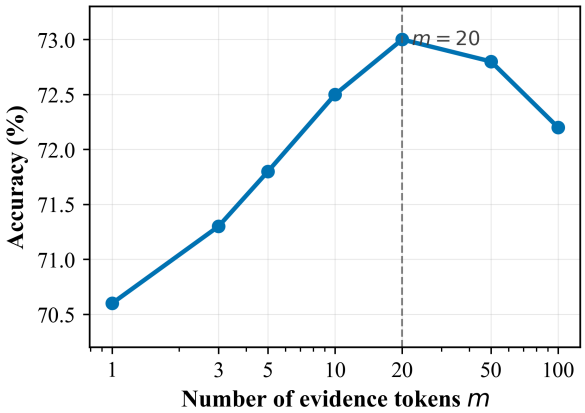

Figure 7: **Effect of graph-conditioned evidence-token budget.** We vary the number of graph-conditioned evidence tokens m on iMedQA using Qwen-2.5-72B as the doctor backbone. Accuracy increases with larger evidence-token budgets and peaks at $m = 20$, while larger budgets provide no further gain.

Hyperparameter	Value
Maximum interaction turns	10
Maximum generation tokens	256 per response
Retrieved knowledge triplets	3
Graph encoder	GAT
GAT layers	2
GAT hidden dimension	256
GAT attention heads	4
Graph pooling	Attention pooling
Projector type	Two-layer MLP
Projector activation	SiLU
Evidence tokens m	20
Refinement steps K	5
Optimiser	AdamW
Learning rate	1×10^{-5}
Weight decay	0.01
Batch size	64
Training epochs	5
Evaluation temperature	0.2
Evaluation top-p	0.9
Doctor-agent tuning	Full fine-tuning
Patient agent	Frozen
Triplet agent	Frozen
Patient/triplet agent backbone	Qwen-2.5-72B

Table 3: Hyperparameter settings for GraphMed-LT.

Quality check	Rate (%)
Extracted triplets supported by patient response	98.0
Retrieved triplets clinically relevant to response	78.0
Unanswerable responses correctly triggered	89.0

Table 4: Manual quality check of the frozen auxiliary agents on 100 randomly sampled iMedQA development examples.

upstream evidence.

Potential risks and intended use. Although GraphMed-LT is evaluated only on benchmark-style multi-turn medical conversation tasks, incorrect diagnoses, unsupported follow-up questions, or incomplete patient-evidence integration could be harmful if the system were used directly in biomedical or clinical decision-making. GraphMed-LT should therefore be treated as a research system for evaluating evidence-grounded multi-turn medical reasoning, rather than as a tool for clinical diagnosis, treatment recommendation, triage, or patient-facing deployment.

Artifact licences and intended use. We use publicly available datasets, models, and knowledge resources in accordance with their stated research-use terms and licences. iCRAFT-MD, iMedQA, iMedMCQA, MedMCQA, PrimeKG, BiCA, LLaMA, Qwen, and the compared baseline systems are used only for research evaluation. The derived multi-turn conversation settings and patient-specific graph memories are constructed for benchmark evaluation and are not intended for clinical deployment or patient-facing use. The released GraphMed-LT code will be distributed for research.

A.3 Triplet Retrieval Details

This section provides additional details for retrieving auxiliary knowledge triplets. At turn t, the triplet agent extracts patient-specific triplets from the patient response and retrieves relevant knowledge triplets from PrimeKG (Chandak et al., 2023). The extracted triplets represent patient-observed evidence, while the retrieved triplets provide auxiliary background knowledge. Each candidate triplet $\tau = (h, \rho, o)$ is linearised into a textual sequence containing its head entity, relation, and tail entity. We encode both the patient response and each linearised candidate triplet using BiCA, and compute their cosine similarity as the relevance score. The top-3 triplets with the highest relevance scores are retained as retrieved knowledge triplets.

A.4 Patient-Specific Graph Memory Details

This section provides implementation details for the patient-specific graph memory module. We initialise $\mathcal{G}_0$ by extracting triplets from the initial patient description p_0. At turn t, the graph memory is updated using patient-specific triplets extracted from the patient response and relevant knowledge triplets retrieved from the external corpus. Each clinical entity is represented as a node, and each triplet (u, ρ, v) is represented as a directed relation-labelled edge from u to v.

Source-aware edge typing. Extracted and retrieved triplets are assigned different edge source types:

$$\sigma(e) \in \{\text{patient}, \text{retrieved}\}. \tag{7}$$

Edges with $\sigma(e) = \text{patient}$ are extracted from patient responses, while edges with $\sigma(e) = \text{retrieved}$ are retrieved from the external triplet corpus. This preserves the distinction between patient-observed evidence and auxiliary background knowledge. The relation label and source type are represented by trainable embeddings and used in the graph encoder attention score.

Node initialisation. Each clinical entity is represented using a BiCA embedding computed from its textual form:

$$\mathbf{h}_i^{(0)} = \mathrm{BiCA}(\mathrm{Text}(v_i)), \tag{8}$$

where $\mathrm{Text}(v_i)$ denotes the textual form of node v_i.

GAT-based graph encoder. The graph encoder is implemented using a graph attention network (GAT). At layer ℓ, node i attends to its neighbours $\mathcal{N}(i)$ and updates its representation as:

$$\mathbf{h}_i^{(\ell+1)} = \sigma\left(\sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(\ell)} \mathbf{W}^{(\ell)} \mathbf{h}_j^{(\ell)}\right), \tag{9}$$

where $\mathbf{W}^{(\ell)}$ is a trainable transformation matrix, $\alpha_{ij}^{(\ell)}$ is the attention weight from node i to node j, and $\sigma(\cdot)$ is a non-linear activation function. The attention weight $\alpha_{ij}^{(\ell)}$ is computed from the transformed node representations together with trainable embeddings of the relation label and source type of the edge between nodes i and j.

Graph-to-token projector. After graph encoding and attention pooling, the graph-level representation $\mathbf{g}_t$ is mapped into graph-conditioned evidence tokens through a trainable projector:

$$\begin{aligned}\mathbf{Z}_t &= \mathrm{Reshape}\left(\mathrm{MLP}_{\theta_{\mathrm{proj}}}(\mathbf{g}_t)\right)\\ &= [z_1, \ldots, z_m] \in \mathbb{R}^{m\times d},\end{aligned} \tag{10}$$

where θ_{proj} denotes the projector parameters, m is the number of graph-conditioned evidence tokens, and d is the hidden dimension of the doctor agent.

A.5 Latent Clinical Thought Refinement Details

This section provides implementation details for latent clinical thought refinement inside the doctor agent. The refinement procedure keeps graph-conditioned evidence tokens fixed and iteratively updates their corresponding thought tokens in continuous hidden space.

Thought initialisation. Given graph-conditioned evidence tokens $\mathbf{Z} = [z_1, \ldots, z_m] \in \mathbb{R}^{m\times d}$, we initialise one thought token for each evidence token. In our implementation, the initial thought states are obtained by applying the doctor agent to the evidence tokens and reading the corresponding hidden states:

$$\mathbf{H}^0 = [h_1^0, \ldots, h_m^0] = \mathrm{Doctor}_{\theta_{\mathrm{doc}}}(\mathbf{Z})_{1:m}. \tag{11}$$

Interleaved hidden-state refinement. At refinement step k, each fixed evidence token is interleaved with its corresponding thought token:

$$\mathbf{S}^k = [z_1, h_1^k, z_2, h_2^k, \ldots, z_m, h_m^k] \in \mathbb{R}^{2m\times d}. \tag{12}$$

The doctor agent processes this interleaved sequence and returns hidden states for all $2m$ positions:

$$\begin{aligned}\widetilde{\mathbf{H}}^{k+1} &= [\widetilde{\mathbf{H}}_1^{k+1}, \widetilde{\mathbf{H}}_2^{k+1}, \ldots, \widetilde{\mathbf{H}}_{2m}^{k+1}]\\ &= \mathrm{Doctor}_{\theta_{\mathrm{doc}}}(\mathbf{S}^k).\end{aligned} \tag{13}$$

Because the odd positions correspond to evidence tokens and the even positions correspond to thought tokens, only the even-position hidden states are used to form the next-step thought states:

$$h_i^{k+1} = \widetilde{\mathbf{H}}_{2i}^{k+1}, \qquad i = 1, \ldots, m, \tag{14}$$

or equivalently,

$$\mathbf{H}^{k+1} = [\widetilde{\mathbf{H}}_2^{k+1}, \widetilde{\mathbf{H}}_4^{k+1}, \ldots, \widetilde{\mathbf{H}}_{2m}^{k+1}]. \tag{15}$$

The graph-conditioned evidence tokens z_i are reused unchanged at every refinement step, while only the thought tokens are updated:

$$\mathbf{S}^{k+1} = [z_1, h_1^{k+1}, z_2, h_2^{k+1}, \ldots, z_m, h_m^{k+1}]. \tag{16}$$

Thus, the refinement length remains fixed at $2m$ and does not grow with the number of refinement steps.

A.6 Datasets Description

iMedQA. iMedQA is reformulated from MedQA (Jin et al., 2021) into a multi-turn medical conversation setting. Each case contains an initial patient description, a complete patient record accessible only to the patient agent, an MCQ, candidate answers, and the gold answer. The doctor agent initially observes only partial patient information and must ask follow-up questions to acquire additional evidence before making a final answer prediction.

iCRAFT-MD. iCRAFT-MD is a dermatology-focused multi-turn medical conversation dataset (Johri et al., 2023, 2024). It is used as an external evaluation set to assess whether the model generalises to clinically realistic patient–doctor interactions beyond the main training benchmark.

Table 5: Dataset statistics. iMedQA provides train/dev/test splits, while iCRAFT-MD and iMedMCQA are used as test-only evaluation sets.

Dataset	Train	Dev	Test	Source
iMedQA	10,178	1,272	1,273	MedQA (Jin et al., 2021)
iCRAFT-MD	–	–	140	CRAFT-MD (Johri et al., 2023, 2024)
iMedMCQA	–	–	200	MedMCQA (Pal et al., 2022)

iMedMCQA. We construct iMedMCQA from MedMCQA (Pal et al., 2022) as an additional test-only dataset. We sample 200 medical MCQs and convert them into multi-turn medical conversations. For each case, the original answer option is preserved as the gold label. The question stem is decomposed into patient-specific atomic facts. Only partial information, such as demographic information and chief complaint when available, is exposed as the initial patient description, while the remaining facts are hidden from the doctor agent and can only be obtained through follow-up questions.

A.7 Descriptions of Baselines

We compare GraphMed-LT with baselines that support follow-up questioning and maintain information acquired across turns.

MEDIQ. MEDIQ (Li et al., 2024) formulates medical QA as an interactive doctor–patient reasoning process. The doctor agent asks follow-up questions to a patient simulator and reasons over accumulated conversation history before producing a final answer.

ProMed. ProMed (Ding et al., 2025) encourages proactive information seeking using reinforcement learning with Shapley Information Gain rewards. We adapt it to the same MCQ-based multi-turn protocol by allowing the doctor agent to use acquired patient responses before selecting the final answer.

DDO. DDO (Jia et al., 2025) formulates medical conversation as a dual-decision optimisation problem, jointly modelling inquiry and diagnosis. Since DDO was originally designed for symptom-centric disease diagnosis, we adapt its inquiry-and-diagnosis process to MCQ answer selection under the same interaction protocol.

UoT. UoT (Hu et al., 2024) uses uncertainty-aware planning to guide information seeking. In our setting, it selects follow-up questions based on uncertainty in the current reasoning state and then predicts the final MCQ answer after sufficient evidence has been collected or the turn budget is reached.

CoD. CoD (Chen et al., 2025) maintains an interpretable Chain-of-Diagnosis process over acquired clinical evidence. We adapt it to update the diagnosis chain across turns and use the final chain to select one answer from the MCQ candidates.

B Patient Agent Prompts

Atomic fact decomposition Prompt

You are a truthful medical assistant that understands the patient's information.
Break the following patient information into a list of independent atomic facts, with one piece of information in each statement. Each fact should only include the smallest unit of information, but should be self-contained.
"She says that, despite going to bed early at night, she is unable to fall asleep. She denies feeling anxious or having disturbing thoughts while in bed. Even when she manages to fall asleep, she wakes up early in the morning and is unable to fall back asleep. She says she has grown increasingly irritable and feels increasingly hopeless, and her concentration and interest

Atomic fact decomposition Prompt (cont'd)

at work have diminished. The patient denies thoughts of suicide or death. Because of her diminished appetite, she has lost 4 kg (8.8 lb) in the last few weeks and has started drinking a glass of wine every night instead of eating dinner. She has no significant past medical history and is not on any medications."
Respond with the list of atomic facts and nothing else, prepend each fact by an index starting from 1. No sub-list allowed.

Example LLM Output

1.patient goes to bed early at night but is unable to fall asleep.
2.patient denies feeling anxious or having disturbing thoughts while in bed.
3.patient wakes up early in the morning and is unable to fall back asleep.
4.patient has grown increasingly irritable and feels increasingly hopeless.
5.patient's concentration and interest at work have diminished.
6.patient denies thoughts of suicide or death.
7.patient has lost 4 kg (8.8 lb) in the last few weeks.
8.patient started drinking a glass of wine every night instead of eating dinner.
9.patient has no significant past medical history.
10.patient is not on any medications."

patient Response Generation Prompt

You are a truthful assistant that understands the patient's information, and you are trying to answer questions from a medical doctor about the patient.
Below is a list of factual statements about the patient:
1.patient goes to bed early at night but is unable to fall asleep.
2.patient denies feeling anxious or having disturbing thoughts while in bed.
3.patient wakes up early in the morning and is unable to fall back asleep.
4.patient has grown increasingly irritable and feels increasingly hopeless.
5.patient's concentration and interest at work have diminished.
6.patient denies thoughts of suicide or death.
7.patient has lost 4 kg (8.8 lb) in the last few weeks.
8.patient started drinking a glass of wine every night instead of eating dinner.
9.patient has no significant past medical history.
10.patient is not on any medications.
Question from the doctor: "What time do you usually go to bed at night?"
Which of the above atomic factual statements answer the question? Select at most two statements. If no statement answers the question, simply say "The patient cannot answer this question, please do not ask this question again." Answer only what the question asks for. Do not provide any analysis, inference, or implications. Respond by selecting all statements that answer the question from above ONLY and NOTHING ELSE.

Example LLM Output

patient goes to bed early at night but is unable to fall asleep.

C Doctor Agent Prompts

GraphMed-LT-doctor — Structured Assessment Prompt

You are a medical doctor answering real-world medical entrance exam questions. Based on your understanding of basic and clinical science, medical knowledge, and mechanisms underlying health, disease, patient care, and modes of therapy, answer the following multiple choice question. Select one correct answer from A to D. Base your answer on current standards and evidence-based guidelines.
Task: You will be asked to reason through the current patient's information using both the patient initial information and structured clinical facts (UMLS-style triplets).
patient initial information: "A 40-year-old woman presents with difficulty falling asleep, diminished appetite, and tiredness for the past 6 weeks."
Knowledge triplets:
(patient, HAS_SYMPTOM, insomnia)
(patient, HAS_SYMPTOM, reduced appetite)
(patient, HAS_SYMPTOM, fatigue)
(insomnia, DURATION, six weeks)

GraphMed-LT-doctor — Structured Assessment Prompt (cont'd)

Given the information above, your task is to choose one of four options that best answers the inquiry.
INQUIRY: "Which of the following is the best course of treatment in this patient?"
OPTIONS: A: Diazepam, B: Paroxetine, C: Zolpidem, D: Trazodone
Medical conditions are complex, so you should consider the relevant specialty, enumerate the key features required for a correct judgement, and assess whether the structured evidence provided by the triplets is sufficient. Think step by step about the patient initial information, the triplets, the inquiry, and the options. **Do not provide the answer choice**; keep your response to one paragraph.

Example LLM Output – Case 1: Follow-up Question

Have you also been experiencing feelings of sadness or loss of interest in daily activities?

Example LLM Output – Case 2: Prediction

B

D Triplet Agent Prompts

GraphMed-LT Triplet Extraction – Initial Information Prompt

You are a medical terminology normaliser and knowledge graph constructor. Your task is to extract structured, clinically relevant facts from the patient's initial presentation, representing each fact as a UMLS-style triplet in the format (subject, relation, object). Only include directly observed and verifiable information without inferring diagnoses, causes, or risks not explicitly stated. Use only medically meaningful relations such as HAS_SYMPTOM, HAS_SIGN, HAS_MEDICATION, HAS_ALLERGY, HAS_PAST_MEDICAL_HISTORY, FAMILY_HISTORY_OF, SOCIAL_HISTORY_OF, FINDING_OF, TEMPORAL_PATTERN, SEVERITY, LOCATION, LATERALITY, DURATION, COURSE, and FUNCTIONAL_IMPACT. When appropriate, express information in a two-step form by stating a base fact and then attaching qualifiers to that concept, avoiding compounded objects. Extract at most three triplets per sentence, outputting NO_TRIPLET if no relevant fact is present, and maintain consistent clinical terminology across extractions.

patient initial information: "A 40-year-old woman presents with difficulty falling asleep, diminished appetite, and tiredness for the past 6 weeks."

Example LLM Output

(patient, HAS_SYMPTOM, insomnia)
(insomnia, DURATION, six weeks)
(patient, HAS_SYMPTOM, reduced appetite)
(patient, HAS_SYMPTOM, fatigue)

GraphMed-LT Triplet Extraction – Incremental Update Prompt

You are a medical terminology normaliser and knowledge graph constructor. Given the patient's existing knowledge triplets and new conversational information, extract up to three additional UMLS-style triplets in the format (subject, relation, object). Only include directly observed and verifiable information from the new input without inferring diagnoses, causes, or risks not explicitly stated. Avoid duplicating facts already present in the existing triplets. Use only medically meaningful relations such as HAS_SYMPTOM, HAS_SIGN, HAS_MEDICATION, HAS_ALLERGY, HAS_PAST_MEDICAL_HISTORY, FAMILY_HISTORY_OF, SOCIAL_HISTORY_OF, FINDING_OF, TEMPORAL_PATTERN, SEVERITY, LOCATION, LATERALITY, DURATION, COURSE, and FUNCTIONAL_IMPACT. When appropriate, express information in a two-step form by stating a base fact and then attaching qualifiers to that concept, avoiding compounded objects. Extract at most three triplets per sentence, outputting NO_TRIPLET if no relevant fact is present, and ensure consistent terminology with the existing knowledge graph.

Existing triplets: (patient, HAS_SYMPTOM, insomnia)
(patient, HAS_SYMPTOM, reduced appetite)
(patient, HAS_SYMPTOM, fatigue)

New patient information: "Her sleep problem is worse in the early mornings and she often needs daytime naps. She drinks two cups of coffee after dinner."

Example LLM Output

(insomnia, TEMPORAL_PATTERN, worse in early mornings)
(patient, FUNCTIONAL_IMPACT, daytime naps)
(patient, SOCIAL_HISTORY_OF, evening caffeine use)