

# MIRAGE: Scaling Test-Time Inference with Parallel Graph-Retrieval-Augmented Reasoning Chains

Kaiwen Wei<sup>\*1</sup>, Rui Shan<sup>\*1</sup>, Dongsheng Zou<sup>1†</sup>, Jianzhong Yang<sup>1</sup>,  
Bi Zhao<sup>1</sup>, Junnan Zhu<sup>2</sup>, Jiang Zhong<sup>1</sup>

<sup>1</sup>College of Computer Science, Chongqing University, Chongqing, China

<sup>2</sup>State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, CAS, Beijing, China  
{weikaiwen, dszhou, zjstud}@cqu.edu.cn, {ruishan, yangjianzhong, zhaobi}@stu.cqu.edu.cn, junnan.zhu@nlpr.ia.ac.cn

## Abstract

Large reasoning models (LRMs) have shown significant progress in test-time scaling through chain-of-thought prompting. Current approaches like search-o1 integrate retrieval augmented generation (RAG) into multi-step reasoning processes but rely on a single, linear reasoning chain while incorporating unstructured textual information in a flat, context-agnostic manner. As a result, these approaches can lead to error accumulation throughout the reasoning chain, which significantly limits its effectiveness in medical question-answering (QA) tasks where both accuracy and traceability are critical requirements. To address these challenges, we propose MIRAGE (Multi-chain Inference with Retrieval-Augmented Graph Exploration), a novel test-time scalable reasoning framework that performs dynamic multi-chain inference over structured medical knowledge graphs. Specifically, MIRAGE 1) decomposes complex queries into entity-grounded sub-questions, 2) executes parallel inference chains, 3) retrieves evidence adaptively via neighbor expansion and multi-hop traversal, and 4) integrates answers using cross-chain verification to resolve contradictions. Experiments on three medical QA benchmarks (GenMedGPT-5k, CMCQA, and ExplainCPE) show that MIRAGE consistently outperforms GPT-4o, Tree-of-Thought variants, and other retrieval-augmented baselines in both automatic and human evaluations. Additionally, MIRAGE improves interpretability by generating explicit reasoning chains that trace each factual claim to concrete chains within the knowledge graph, making it well-suited for complex medical reasoning scenarios.

**Code** — <https://github.com/Despacitobei/MIRAGE/>

## Introduction

The developments of large language models (LLMs) have led to the emergence of large reasoning models (LRMs), which exhibit strong multi-step reasoning abilities through chain-of-thought (CoT) prompting (Wei et al. 2022). Unlike conventional LLMs that often provide shallow or heuristic answers, LRMs adopt a slow-thinking paradigm that enables the progressive unfolding of reasoning steps. More importantly, recent advances in test-time scaling (Snell et al.

<sup>\*</sup>These authors contributed equally.

<sup>†</sup>Corresponding author.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

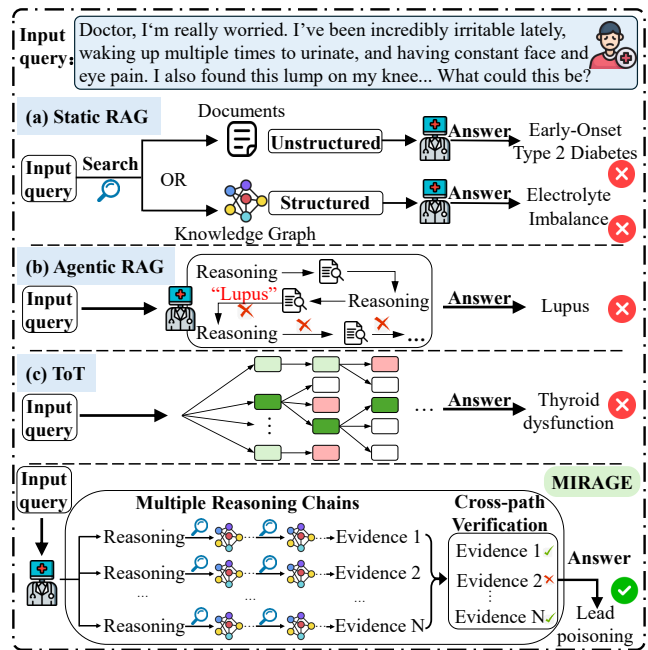


Figure 1: Comparison of medical QA strategies: (a) Static RAG retrieves documents or knowledge graph entries without explicit reasoning; (b) Agentic RAG methods like Search-o1 integrate retrieval with linear reasoning; (c) ToT explores multiple reasoning chains via sampling; (d) The proposed MIRAGE combines these approaches by performing graph-based retrieval across parallel reasoning chains.

2024) have demonstrated that reasoning capability can be significantly enhanced at inference time without retraining the underlying model, simply by allocating more computational resources during the reasoning process. This paradigm is exemplified by models such as OpenAI’s o1 (OpenAI et al. 2024b), Qwen-QwQ (Team 2024b), and DeepSeek-R1 (DeepSeek-AI et al. 2025), which achieve substantial performance improvements through extended reasoning chains and explicit step-by-step thought processes (Team 2024a). Building on these foundations, agentic frameworks like Search-o1 (Li et al. 2025a) further enhance test-

time scaling by integrating retrieval-augmented generation (RAG) into the reasoning loop, enabling models to dynamically access external knowledge sources during inference.

Despite the promise of test-time scaling, current approaches face a fundamental limitation in their scaling strategy: they primarily rely on linear expansion through sequential reasoning chains or iterative retrieval rounds. As illustrated in Figure 1 (b), this linear scaling approach is inherently inefficient and sensitive to error propagation. When early reasoning steps are incorrect or based on incomplete evidence, the entire extended reasoning chain becomes compromised. While techniques such as Tree-of-Thoughts (ToT) (Yao et al. 2023a) and its extensions like ARise (Zhang et al. 2025) attempt to explore multiple reasoning paths, they lack coherent mechanisms for coordinating parallel inference chains and enabling explicit cross-chain verification. This limitation becomes particularly pronounced in complex domains like medicine, where reasoning errors can have critical consequences and where the linear scaling paradigm fails to effectively leverage the additional computational budget.

Furthermore, current test-time scaling methods face a critical challenge in how they expand their knowledge coverage during inference. Existing retrieval-augmented approaches (Li et al. 2025a) typically acquire unstructured textual information and integrate it into the reasoning process in a flat, context-agnostic manner. This approach to knowledge scaling overlooks the inherent structural relationships and semantic hierarchies within domain knowledge, particularly in specialized fields like medicine, where understanding often depends on complex inter-entity relationships, causal chains, and hierarchical taxonomies (Fang et al. 2020; Wen, Wang, and Sun 2024; Wei et al. 2021). As a result, even when more computational resources are allocated to retrieve additional information, the flat integration of isolated text fragments limits the system’s ability to perform precise multi-hop reasoning and construct coherent knowledge-grounded arguments. This structural knowledge scaling challenge significantly constrains the effectiveness of test-time scaling in knowledge-intensive domains.

To address these fundamental limitations in test-time scaling, we propose Multi-chain Inference with Retrieval-Augmented Graph Exploration (MIRAGE), a novel reasoning framework that reimagines how computational resources are allocated during inference. Rather than scaling through linear chain extension, MIRAGE implements parallel scaling by decomposing complex queries into semantically coherent sub-questions and executing multiple reasoning trajectories simultaneously. This parallel inference paradigm enables more efficient utilization of computational resources while providing natural mechanisms for cross-chain verification and error correction. To address the knowledge scaling challenge, MIRAGE integrates an adaptive retrieval-reasoning loop that interacts with structured domain knowledge graphs, enabling dynamic exploration of semantic relationships through neighbor expansion and multi-hop traversal strategies. By organizing knowledge retrieval around graph structures rather than flat text fragments, MIRAGE can perform more sophisticated reasoning

that leverages the inherent hierarchical and relational properties of domain knowledge. Following the evaluation protocol established by MindMap (Wen, Wang, and Sun 2024), we conduct extensive experiments on three medical question answering (QA) benchmarks, demonstrating that MIRAGE consistently outperforms strong baselines including GPT-4o, Tree-of-Thought variants, and other retrieval-augmented approaches in both automatic metrics and human evaluations across all datasets. In summary, the contributions of this paper are as follows:

(1) We propose MIRAGE, a novel test-time scaling framework that shifts from linear chain extension to parallel multi-chain inference, enabling more efficient utilization of computational resources while providing mechanisms for cross-chain verification and error correction.

(2) We develop a structured knowledge scaling approach through adaptive graph-based retrieval, which dynamically explores semantic relationships and hierarchical structures during inference rather than relying on flat text integration.

(3) We conduct comprehensive evaluations on three medical QA benchmarks, demonstrating significant improvements over existing reasoning and retrieval baselines, with detailed analysis showing the effectiveness of both parallel reasoning scaling and structured knowledge scaling components in improving performance.

## Related Work

**LLMs for Reasoning** Prompt-based reasoning methods, including Chain-of-Thought (CoT) prompting (Wei et al. 2022), Self-Consistency (Wang et al. 2023), Tree-of-Thought (Yao et al. 2023a), Graph-of-Thought (Besta et al. 2024), and Chain-of-Specificity (Wei et al. 2025b), have substantially improved the reasoning capabilities of large language models (LLMs) without altering model parameters (Chu et al. 2024; Plaat et al. 2024). These methods rely on carefully designed prompts to elicit multi-step, interpretable rationales and have been extended to diverse tasks and modalities in various domains (Zhang et al. 2024).

Recent advances in large reasoning models (LRMs) have led to the development of large language models that are explicitly trained to internalize multi-step reasoning processes. Representative models include DeepSeek-R1 (DeepSeek-AI et al. 2025), OpenAI-o1 (OpenAI et al. 2024b), and Qwen-QwQ (Team 2024b). Through reinforcement learning and reasoning-oriented supervision, these models acquire explicit chain-of-thought abilities and demonstrate strong performance on complex benchmarks such as mathematics and coding (Qu et al. 2025). Nevertheless, both prompt- and model-based approaches are limited by knowledge gaps and error accumulation, often yielding unreliable answers (Chu et al. 2024; Plaat et al. 2024; Qu et al. 2025). Search-o1 (Li et al. 2025a) and its variants (Li et al. 2025b; Guan et al. 2025) integrate agentic search into the o1-like reasoning process of LLMs but follow a monolithic, sequential paradigm lacking modularity and cross-step coordination. ARise (Zhang et al. 2025) adds multi-path tree search with dynamic retrieval but lacks structured path alignment and global consistency, limiting complex multi-hop reasoning.

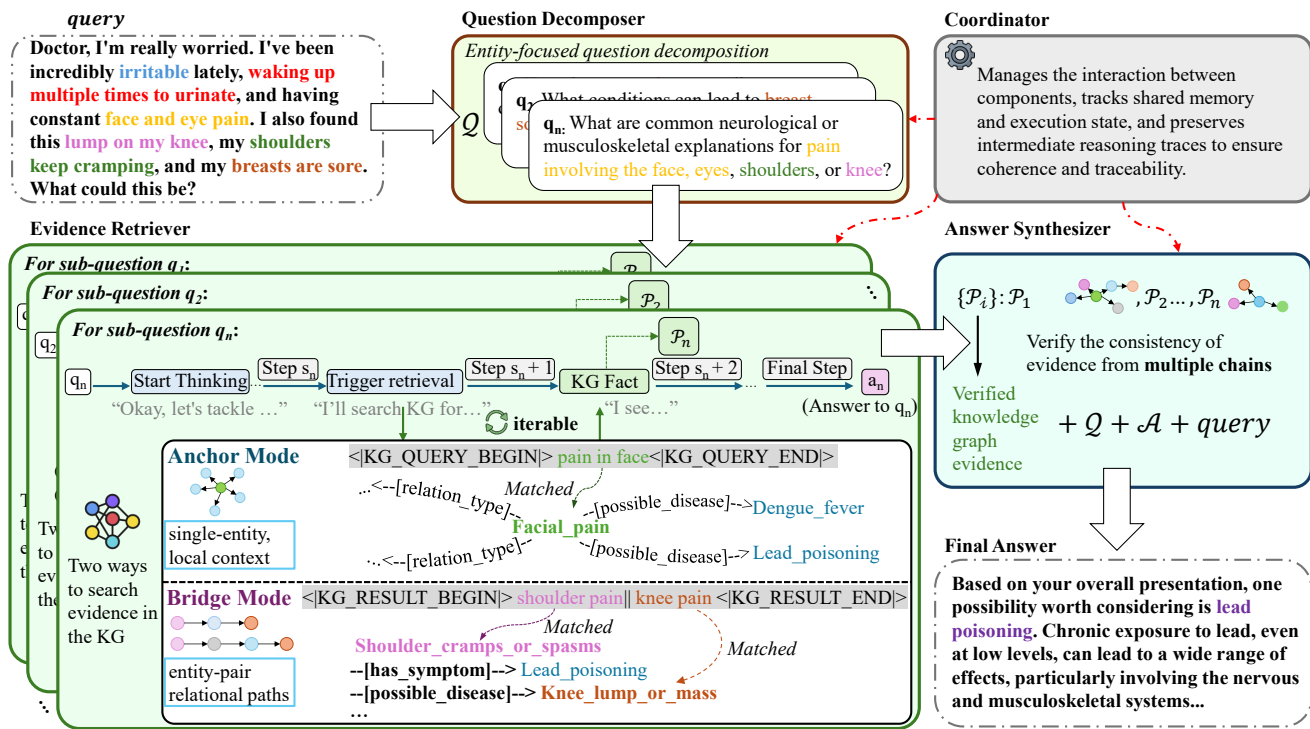


Figure 2: Overview of the proposed MIRAGE framework. Given a clinical query, the system decomposes it into sub-questions, each initiating a reasoning chain. For each chain, the system iteratively retrieves knowledge graph evidence using either Anchor mode or Bridge mode. Retrieved results are coordinated and aggregated to generate the final answer.

**Retrieval-Augmented Generation** Retrieval-Augmented Generation (RAG) integrates external knowledge retrieval into the generation process to enhance factual accuracy and reduce hallucinations (Lewis et al. 2020). Classic RAG methods typically employ static, single-step retrieval prior to generation and have been widely adopted in QA, dialogue, and summarization tasks (Guu et al. 2020; Zhu et al. 2025a; Wei et al. 2025a). To improve retrieval relevance, later studies explored query rewriting (Ma et al. 2023), reranking (Glass et al. 2022), document compression (Xu, Shi, and Choi 2024), and GraphRAG (Edge et al. 2024; Wen, Wang, and Sun 2024). Meanwhile, generation-side efforts included improved conditioning strategies (Izcard et al. 2023) and post-generation provenance verification and evaluation (Sankararaman et al. 2024; Zhu et al. 2025b).

Although existing RAG methods have demonstrated significant success, they remain limited in multi-step reasoning settings, where information needs to evolve dynamically and cannot be satisfied by a single retrieval step. Recent agentic RAG frameworks (Li et al. 2025a; Yao et al. 2023b; Nakano et al. 2022; Asai et al. 2024; Zhang et al. 2025) address this by interleaving retrieval and generation in a loop, enabling models to query external sources based on intermediate reasoning states. However, most systems still operate on unstructured text, lacking dynamic access to structured or semantically grounded representations (Wei et al. 2023). This limits their ability to support fine-grained, multi-hop reason-

ing, especially in the medical domain that requires entity-centric, relational, and hierarchical knowledge modeling.

## Method

### Task Definition

In this work, we focus on leveraging structured knowledge graphs to support multi-chain inference for medical question answering (QA). Specifically, let  $query$  be a natural-language medical question posed by a clinician or patient, and let  $\mathcal{G} = (\mathcal{E}, \mathcal{R})$  denote a medical knowledge graph, where  $\mathcal{E}$  is a set of medical entities (e.g., diseases, drugs, symptoms, foods, ...) and  $\mathcal{R}$  is a set of typed relations capturing associations between these entities. We use  $\mathcal{G}$  as the primary external knowledge source because its structured format, consisting of deduplicated entities and clearly defined relations, provides concise and consistent factual information. This helps mitigate the ambiguity and redundancy often found in free-form clinical text. The system returns a single natural-language answer  $a \in \mathbb{T}^+$ , where  $\mathbb{T}^+$  denotes the set of non-empty free-form text strings.

### Overall Framework

Figure 2 presents the structure of MIRAGE, which comprises four main components:

- **Question Decomposer:** counters the common weakness of earlier decomposers that emit entity-agnostic, free-

form sub-questions; instead, it produces concise splits that are explicitly grounded in concrete medical entities, keeping the context small and noise-free for downstream steps.

- **Evidence Retriever:** engages in a think-while-search cycle, alternating language-model reasoning with targeted knowledge-graph queries. Each round can refine or extend previous hypotheses, gradually assembling the evidence required for every sub-question.
- **Answer Synthesizer:** consolidates the partial answers of all sub-questions and verifies that they are mutually consistent, resolving any contradictions before emitting a single, coherent answer with traceable citations.
- **Coordinator:** manages the execution of the three components by facilitating communication through a shared in-memory workspace. It monitors the workspace and activates downstream modules as soon as their required inputs become available.

For example, given the query “*Why do I keep feeling fatigued even after sleeping well?*”, the question decomposer first extracts key entities like *fatigue* and *sleep quality* and generates sub-questions grounded in each, such as identifying causes of fatigue or conditions affecting sleep recovery. The evidence retriever then performs multi-hop reasoning over the knowledge graph, retrieving relevant entities like *anemia* or *thyroid disorders*. These intermediate results are merged by the answer synthesizer into a coherent explanation. Throughout the process, the coordinator monitors shared state and triggers each component as inputs become ready. Only output the final answer “*Based on your symptoms, chronic fatigue syndrome is the most likely cause...*”, keeping the response concise and readable.

**Question Decomposition** Question decomposition serves as the foundational step of the adaptive loop, transforming complex clinical queries into focused sub-questions that align with the granularity of  $\mathcal{G}$ . This step is designed to address the limitations of prior methods, which often generate entity-agnostic sub-questions prone to noise or irrelevant retrieval. Specifically, this decomposition strategy adheres to two domain-specific principles. First, it triggers exclusively for queries involving multiple distinct medical entities (e.g., concurrent symptoms or drug-condition interactions), thereby preserving the coherence of single-topic queries and preventing unnecessary fragmentation. Second, it replaces ambiguous references with explicit entities extracted from the original query, ensuring each sub-question remains self-contained and directly mapped onto entities in  $\mathcal{E}$ . This strategy is implemented via an LLM prompted with structured instructions to produce focused, entity-grounded sub-questions (see Appendix A.1 for details). This decomposition process initializes the loop by defining clear retrieval targets. To prevent excessive fragmentation, at most  $N_q$  sub-questions are generated per query. Each sub-question acts as a starting point for targeted knowledge graph exploration, guiding the loop to focus on relevant entities and relations rather than sprawling, unfocused searches.

**Graph-Augmented Evidence Retrieval** Following decomposition, the model enters an iterative retrieval-and-

reasoning loop, in which each stage is guided by structured prompts (see Appendix A.1). The loop consists of three steps: identifying retrieval targets, querying the knowledge graph, and verbalizing the retrieved evidence. When decoding a sub-question, the model may emit a special search block  $\vartheta$ , delimited by  $\langle |KG\_QUERY\_BEGIN| \rangle$  and  $\langle |KG\_QUERY\_END| \rangle$ , which defines the retrieval target. This block includes one or two entity mentions that are softly matched to entities in  $\mathcal{E}$  based on embedding similarity. The matched entity set  $\mathcal{E}^*$  is defined as follows:

$$\mathcal{E}^* = \{ \arg \max_{e \in \mathcal{E}} \text{sim}(\hat{e}, e) \mid \hat{e} \in \hat{\mathcal{E}}, \text{sim}(\hat{e}, e) \geq \tau \} \quad (1)$$

where  $\text{sim}(\cdot)$  is a normalized similarity function,  $\tau$  is a threshold, and  $\hat{\mathcal{E}}$  is the set of candidate entities from  $\vartheta$ . If no match exceeds the threshold, a special `no_entity_match` token is returned, prompting reformulation. Once the target is resolved, the model invokes the query function:

$$\mathcal{P}_i = \text{KGSEARCH}(\vartheta, \mathcal{G}) \quad (2)$$

The strategy depends on the number of entities in the search block. For a single entity  $e$ , the system retrieves its local neighborhood:

$$\mathcal{N}(e) = \{ (e, r, e') \mid (e, r, e') \in \mathcal{G}, r \in \mathcal{R}, |\mathcal{N}(e)| \leq k \} \quad (3)$$

where  $k$  is the maximum number of neighbors per relation. If two entities ( $e_1, e_2$ ) are specified, the system searches for typed relational chains of length up to  $h$ :

$$\mathcal{P}_h(e_1, e_2) = \{ p \mid p : e_1 \rightarrow e_2, |p| \leq h \} \quad (4)$$

where each  $p$  is a typed relational chain from  $e_1$  to  $e_2$ .

These retrieval behaviors can be categorized into two modes: (1) **Anchor Mode** handles single-entity queries by retrieving a fixed neighborhood around the matched entity, capturing local attributes such as symptoms or treatments. It refines entity semantics while keeping sub-questions focused and clinically precise. (2) **Bridge Mode** is triggered when two entities are identified, retrieving direct or multi-hop paths connecting them. This enables cross-entity reasoning (e.g., linking symptoms to comorbidities) via intermediate biomedical relations. All queries respect relation direction and are limited to a curated set of clinically validated types. Once graph paths are retrieved, they are verbalized as:

$$\text{Verbalize}(e_0, r, e_1) = “e_0 r e_1” \quad (5)$$

For example,  $(Diabetes, has\_symptom, Fatigue)$  becomes “Diabetes has symptom Fatigue.” These fragments are inserted back into the model’s context between  $\langle |KG\_RESULT\_BEGIN| \rangle$  and  $\langle |KG\_RESULT\_END| \rangle$ , and logged in a shared workspace along with metadata (e.g., retrieval counts and reasoning turns). The model then continues decoding with this updated context, potentially issuing more queries if the retrieval budget allows. Over multiple rounds, the verbalized fragments accumulate into  $\mathcal{P}_i$ , forming a compact, deduplicated, and directionally grounded evidence set for final answer synthesis.

This adaptive retrieval provides three benefits: (1) it keeps context focused by injecting only relevant facts; (2) it supports iterative refinement as new evidence is retrieved; and (3) it grounds all claims in specific, traceable graph paths, enabling clinical interpretability and auditability.

**Answer Synthesis and Source Attribution** Once the retrieval loop has streamed its final graph fragment, each sub-question  $q_i \in \mathcal{Q}$  is associated with an answer  $a_i \in \mathcal{A}$  and a corresponding chain set  $\mathcal{P}_i$ . At this stage, the responsibility shifts to the answer synthesizer, which revisits each  $(q_i, a_i)$  pair in the context of its supporting evidence  $\mathcal{P}_i$ . A dedicated prompt is used to implement answer synthesis and verification (see Appendix A.1 for details). Medical terms are normalized to their canonical synonyms, dosage units are standardized, and domain-specific rules derived from the knowledge graph are applied. For example, if a drug is found to both *treat* and *cause* the same symptom, the system flags the combination as biologically inconsistent.

Next, the synthesizer performs pairwise comparisons across all answers, identifying mutually exclusive diagnoses or conflicting therapeutic claims. When such conflicts arise, the system retains the answer whose supporting chain set spans a broader neighborhood of corroborating relations or aligns more closely with the original query. This majority-based verification strategy prefers answers supported by multiple independently retrieved evidence chains, while suppressing less consistent alternatives. This ensures that the final output is logically coherent and clinically sound before natural-language generation.

The reconciled set is then supplied to an LLM under a bounded prompt that forbids contradictions with verified answers but allows fallback to medical priors when the knowledge graph is silent. The model condenses the material into one or two paragraphs of patient-oriented prose explaining the likely condition, suggesting pertinent investigations, and outlining first-line management, while staying within clinically coherent bounds. Formally, using the sub-question set  $\mathcal{Q}$ , the corresponding answer set  $\mathcal{A}$ , and supporting evidence chain sets  $\{\mathcal{P}_i\}$ , the final reply is synthesized as:

$$a = \text{SYNTH}(query, \mathcal{Q}, \mathcal{A}, \{\mathcal{P}_i\}_{q_i \in \mathcal{Q}}) \quad (6)$$

where  $a$  is the reply shown to the patient.

After generation, the system serializes the final response  $a$ , the original *query*, the validated sub-answers  $(q_i, a_i)$ , their supporting evidence chains  $\mathcal{P}_i$ , and lightweight runtime metadata into a unified, machine-readable audit record. This record consolidates every stage of the pipeline, including decomposition, retrieval, and synthesis, and serves as a provenance-preserving trace that enhances transparency, mitigates the opacity of large language models, and supports downstream tasks such as auditing, dataset refinement, and reasoning inspection. By detecting contradictions before generation and suppressing unsupported statements, the synthesizer reduces hallucinations and reinforces clinical accuracy. The system’s modular architecture also supports seamless integration of alternative language models or rule-based engines without changes to upstream retrieval components.

The overall workflow of MIRAGE is illustrated in Algorithm 1 of Appendix B.

## Experiment

**Dataset** We evaluate MIRAGE on three public medical QA benchmarks with corresponding knowledge graphs:

GenMedGPT-5k with EMCKG, and CMCQA and ExplainCPE with CMCKG (Wen, Wang, and Sun 2024). These datasets cover diverse clinical QA settings, including open-ended questions, multi-turn dialogues, and multiple-choice exams, across both English and Chinese. The paired knowledge graphs contain structured medical facts (e.g., diseases, symptoms, treatments) to support multi-hop reasoning. Dataset statistics are provided in Appendix C.

**Evaluation** We use both automatic and human-aligned metrics for comprehensive evaluation. For semantic similarity, we report *BERTScore* (Zhang et al. 2020). To assess factual and clinical quality, we adopt *GPT-4o Ranking* (OpenAI et al. 2024a), prompting GPT-4o to perform pairwise and listwise evaluations based on correctness, reasoning, and completeness (specific prompt in Appendix A.2). To reduce bias, output order is randomized across trials. For pairwise comparisons, we report win/tie/loss rates. On the ExplainCPE dataset, we additionally report *answer accuracy* based on ground-truth labels.

**Baselines** We compare MIRAGE with a range of baselines covering different reasoning and retrieval strategies. GPT-4o (OpenAI et al. 2024a) serves as a strong general model, and GPT-4o+ToT (Yao et al. 2023a) augments it with Tree-of-Thought prompting for multi-step reasoning. QWQ-32B (Team 2025) represents a large reasoning model trained for end-to-end inference without prompting. We also include retrieval-augmented QWQ-32B variants: BM25 Retriever (Robertson and Zaragoza 2009) (sparse matching), Embedding Retriever (Lewis et al. 2020) (dense similarity), and MindMap (Wen, Wang, and Sun 2024), which uses knowledge graph-based multi-hop retrieval. Search-o1 (Li et al. 2025a) performs dynamic document retrieval with agent-driven iterative refinement.

**Implementation Details** We use the open-source Qwen-QWQ-32B (Team 2025) model as the backbone LLM for all core components of the proposed framework, including question decomposition, evidence retrieval, and answer synthesis. For the evidence retrieval and final answer generation, we set the maximum input length to 32,768 tokens. Please refer to Appendix D for more details.

**Main Results** Table 1 reports the performance of all methods across three medical QA benchmarks. We summarize the following key observations: (1) MIRAGE consistently achieves the best overall performance in both GPT-4o ranking and answer accuracy. For instance, on GenMedGPT-5k, it reaches a rank of 1.8, and on ExplainCPE, it achieves 84.8% accuracy, outperforming both large-scale models and retrieval-augmented baselines. (2) QWQ-32B outperforms GPT-4o+ToT, indicating that reasoning abilities learned during pretraining may generalize better than prompt-based strategies, even with more powerful LLMs. (3) Static retrieval methods like BM25 and dense retrievers show unstable performance, sometimes helpful but often introducing irrelevant or noisy content. In contrast, structured retrieval via knowledge graphs (e.g., MindMap) provides more consistent improvements, reflecting the need for structure-aware

Method	GenMedGPT-5k				CMCQA				ExplainCPE				
	BERT-Score			Rank	BERT-Score			Rank	BERT-Score			Rank	Acc. (%)
	Prec.	Rec.	F1		Prec.	Rec.	F1		Prec.	Rec.	F1		
GPT-4o	0.810	0.840	0.825	7.4	0.839	0.859	0.849	7.2	0.873	0.875	0.873	6.6	77.8%
GPT-4o+ToT	0.825	0.858	0.841	5.9	0.843	0.858	0.850	6.7	0.891	0.872	0.881	5.8	80.2%
QWQ-32B	0.816	0.857	0.836	4.4	0.840	0.859	0.849	4.6	0.892	0.882	0.887	4.6	82.8%
BM25 Retriever	0.805	0.862	0.832	4.0	0.841	<b>0.861</b>	0.851	4.5	0.890	0.878	0.884	3.3	83.5%
Embedding Retriever	0.799	0.860	0.828	5.4	0.841	0.860	0.850	4.5	0.890	0.881	0.885	5.0	82.4%
MindMap	0.820	0.861	0.841	3.8	0.838	0.858	0.847	3.1	0.891	<b>0.887</b>	<b>0.889</b>	<b>3.1</b>	84.6%
Search-01	0.832	<b>0.868</b>	0.849	3.3	<b>0.847</b>	0.857	0.852	3.0	0.883	0.886	0.885	4.6	80.7%
MIRAGE (ours)	<b>0.841</b>	0.864	<b>0.852</b>	<b>1.8</b>	0.845	<b>0.861</b>	<b>0.853</b>	<b>2.8</b>	<b>0.893</b>	0.884	0.888	<b>3.1</b>	<b>84.8%</b>

Table 1: Performance comparison across three medical QA datasets. Rank indicates the average GPT-4o Ranking (lower is better). Acc. (%) is answer accuracy on ExplainCPE, which consists of multi-answer questions.

Method	ExplainCPE				
	Prec.	Rec.	F1	Rank	Acc.(%)
GPT-4o	0.873	0.875	0.873	7.0	77.8%
GPT-4o+ToT	<b>0.891</b>	0.872	0.881	4.9	80.2%
DeepSeek-R1-32B	0.889	0.875	0.882	4.5	82.2%
BM25 Retriever (DS)	0.888	0.881	0.884	3.5	83.4%
Embedding Retriever (DS)	0.890	0.880	<b>0.885</b>	4.3	83.3%
MindMap (DS)	0.887	0.881	0.884	3.1	84.1%
Search-01 (DS)	0.878	0.873	0.875	5.4	81.1%
MIRAGE (DS)	0.887	<b>0.883</b>	<b>0.885</b>	<b>2.9</b>	<b>84.4%</b>

Table 2: Comparison of MIRAGE and baseline methods using the DeepSeek-R1-32B backbone (denoted as DS).

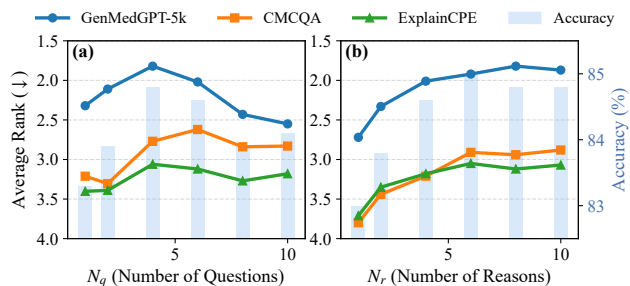


Figure 3: Effect of the decomposition threshold  $N_q$  (a) and retrieval threshold  $N_r$  (b) on GPT-4o Ranking and accuracy.

reasoning in medical QA. (4) Search-01 performs competitively on GenMedGPT-5k and CMCQA but drops notably on ExplainCPE, likely due to noisy web content. MIRAGE remains robust by relying on curated, structured medical knowledge, which enhances both accuracy and reliability.

To validate the reliability of GPT-4o-based automatic evaluation, we conduct a human study on 100 randomly sampled examples from GenMedGPT-5k. We recruit 2 medical graduate students with strong English proficiency (IELTS score  $\geq 7.0$ ) and biomedical backgrounds, and perform pairwise comparisons between MIRAGE and all baselines, assessing factual accuracy, reasoning clarity, and clinical fluency. The specific metric is shown in Appendix E. As illustrated in Figure 4, MIRAGE receives the highest overall

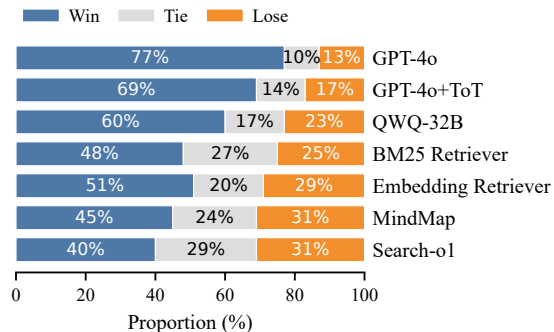


Figure 4: Human evaluation results on GenMedGPT-5k.

preference rate, with substantial win margins and few ties or losses. These results confirm the superiority of MIRAGE from a human perspective and demonstrate strong alignment between GPT-4o rankings and human judgments.

To evaluate the generalizability of MIRAGE, we conduct an experiment on the ExplainCPE dataset using DeepSeek-R1-32B (DeepSeek-AI et al. 2025) as the backbone. All baselines use this backbone accordingly for fair comparison. As shown in Table 2, MIRAGE still outperforms other DeepSeek-based variants, including DeepSeek+ToT, achieving the highest GPT-4o rank and strong answer accuracy. These results confirm that MIRAGE remains effective across different backbone models, highlighting its adaptability and robustness.

**Ablation Study** We perform GPT-4o-based pairwise comparisons between MIRAGE and its ablated variants, averaging over swapped output orders to mitigate positional bias. As shown in Table 3, MIRAGE consistently demonstrates a strong win-loss advantage over all ablated versions, with win rates around 40–48% and loss rates below 15%. While ties are common due to the conservative nature of the evaluator, the clear win margins indicate that both the Question Decomposer and Answer Synthesizer play essential roles in MIRAGE’s effectiveness.

**Further Analysis** We explore the impact of the sub-question threshold  $N_q$  and retrieval threshold  $N_r$ . Specifi-

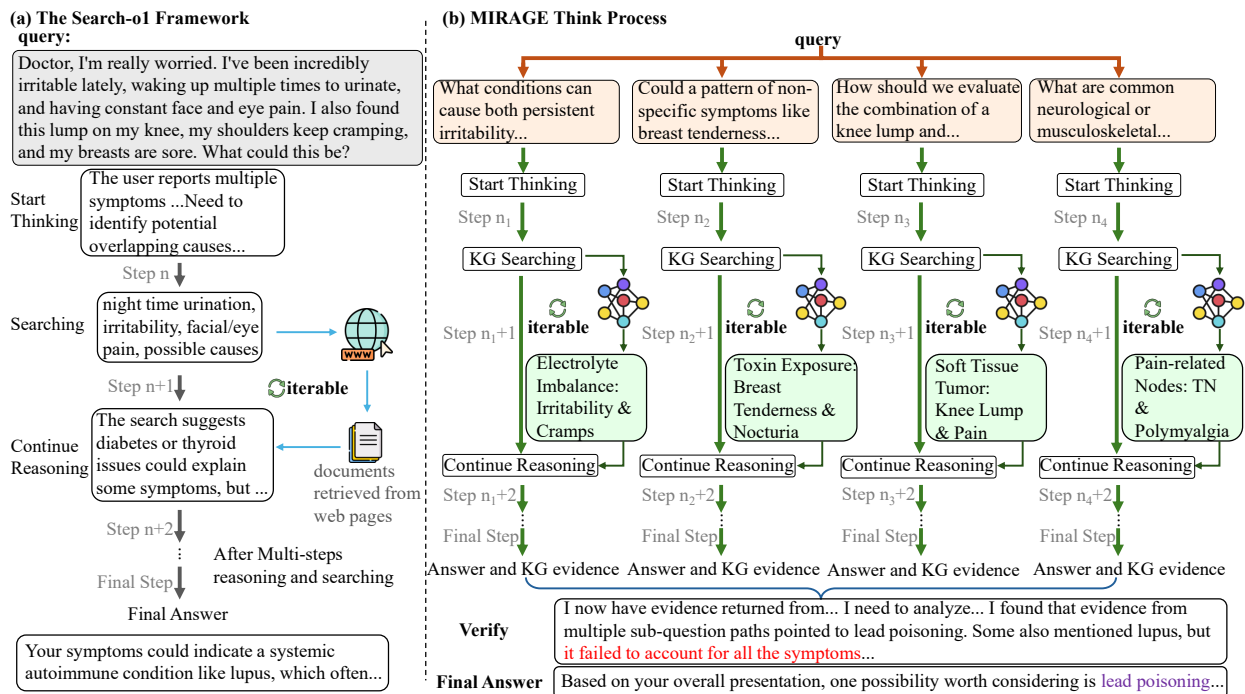


Figure 5: Case Study comparison between single-chain method and the proposed multi-chain Graph RAG reasoning method.

MIRAGE vs Ablated Versions	w/o Decomposer			w/o Synthesizer			w/o Decomposer + Synthesizer		
	Win	Tie	Lose	Win	Tie	Lose	Win	Tie	Lose
Illness identification	49.53	31.13	19.34	47.64	36.32	16.04	52.36	28.77	18.87
Treatment suggestion	34.43	54.25	11.32	41.04	50.47	8.49	47.17	47.64	5.19
Overall correctness	38.21	49.53	12.26	43.40	42.92	13.68	45.28	39.62	15.09
Average	40.72	44.97	14.31	44.03	43.23	12.73	48.27	38.68	13.05

Table 3: Pairwise comparison between ablated versions.

cally, we vary  $N_q$  while fixing  $N_r$  at its default of 5, and vary  $N_r$  while fixing  $N_q$  at its default of 4. This analysis examines how query decomposition and evidence retrieval jointly affect multi-hop reasoning and answer quality. We report the average GPT-4o rank across all datasets and the accuracy on ExplainCPE. As shown in Figure 3 (a), performance improves with increasing  $N_q$  up to a point, then declines as over-decomposition introduces longer or noisier reasoning chains. Although  $N_q$  only sets an upper bound, higher values often lead to overly aggressive splitting. On dialogue-style queries (e.g., CMCQA), performance may even drop from  $N_q = 1$  to  $N_q = 2$ , likely due to disrupted contextual flow. In contrast, Figure 3 (b) shows that increasing  $N_r$  yields diminishing but generally positive returns, without the sharp decline seen with  $N_q$ . This suggests that the retrieval process is demand-driven: the system often performs fewer retrieval steps than the upper limit  $N_r$ , stopping early when sufficient information is obtained, which helps avoid introducing unnecessary noise or redundancy.

**Case Study** To illustrate the contrast between single-chain and multi-chain graph RAG reasoning, we present a case

study comparing Search-o1 and MIRAGE (Figure 5). The baseline Search-o1 performs monolithic reasoning over the entire symptom set using unstructured web retrieval, often leading to information overload and vague explanations. In contrast, MIRAGE identifies toxin exposure and neurological symptoms via separate chains and integrates them to suggest lead poisoning, effectively disentangling complex cases to produce coherent, clinically useful conclusions.

## Conclusion

In this paper, we propose MIRAGE, a test-time scalable framework for dynamic multi-chain reasoning over structured medical knowledge graphs. By decomposing complex queries, adaptively retrieving structured evidence, and verifying answers across parallel inference chains, MIRAGE overcomes key limitations of existing linear and unstructured methods. Experiments on three medical QA benchmarks show that MIRAGE consistently improves both performance and interpretability, making it well-suited for high-stakes domains such as medicine, where accurate and traceable reasoning is essential.

## Acknowledgments

This research was supported by the China National Key R&D Program (Grant No. 2024YFC2707800, 2024YFC2707804), the National Science and Technology Major Project (2025ZD0551300, 2025ZD0551301, 2025ZD0551302), the National Natural Science Foundation of China (62176029, 62506050), the China Postdoctoral Science Foundation Funded Project (2024M763867), and the Chongqing Higher Education Teaching Reform Research Project (No. 242009).

## References

- Asai, A.; Wu, Z.; Wang, Y.; Sil, A.; and Hajishirzi, H. 2024. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Besta, M.; Blach, N.; Kubicek, A.; Gerstenberger, R.; Podstawski, M.; Gianinazzi, L.; Gajda, J.; Lehmann, T.; Niewiadomski, H.; Nyczyk, P.; and Hoefler, T. 2024. Graph of Thoughts: Solving Elaborate Problems with Large Language Models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16): 17682–17690.
- Chu, Z.; Chen, J.; Chen, Q.; Yu, W.; He, T.; Wang, H.; Peng, W.; Liu, M.; Qin, B.; and Liu, T. 2024. Navigate through Enigmatic Labyrinth A Survey of Chain of Thought Reasoning: Advances, Frontiers and Future. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 1173–1203. Bangkok, Thailand: Association for Computational Linguistics.
- DeepSeek-AI; Guo, D.; Yang, D.; Zhang, H.; and et al. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *CoRR*, abs/2501.12948.
- Edge, D.; Trinh, H.; Cheng, N.; Bradley, J.; Chao, A.; Mody, A.; Truitt, S.; and Larson, J. 2024. From Local to Global: A Graph RAG Approach to Query-Focused Summarization. *CoRR*, abs/2404.16130.
- Fang, Y.; Sun, S.; Gan, Z.; Pillai, R.; Wang, S.; and Liu, J. 2020. Hierarchical Graph Network for Multi-hop Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 8823–8838. Online: Association for Computational Linguistics.
- Glass, M.; Rossiello, G.; Chowdhury, M. F. M.; Naik, A.; Cai, P.; and Gliozzo, A. 2022. Re2G: Retrieve, Rerank, Generate. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (ACL)*, 2701–2715. Seattle, United States: Association for Computational Linguistics.
- Guan, X.; Zeng, J.; Meng, F.; Xin, C.; Lu, Y.; Lin, H.; Han, X.; Sun, L.; and Zhou, J. 2025. DeepRAG: Thinking to Retrieve Step by Step for Large Language Models. arXiv:2502.01142.
- Guu, K.; Lee, K.; Tung, Z.; Pasupat, P.; and Chang, M.-W. 2020. REALM: retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, ICML'20. JMLR.org.
- Izacard, G.; Lewis, P.; Lomeli, M.; Hosseini, L.; Petroni, F.; Schick, T.; Dwivedi-Yu, J.; Joulin, A.; Riedel, S.; and Grave, E. 2023. Atlas: few-shot learning with retrieval augmented language models. *J. Mach. Learn. Res.*, 24(1).
- Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; Riedel, S.; and Kiela, D. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS)*, NIPS '20. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713829546.
- Li, X.; Dong, G.; Jin, J.; Zhang, Y.; Zhou, Y.; Zhu, Y.; Zhang, P.; and Dou, Z. 2025a. Search-o1: Agentic Search-Enhanced Large Reasoning Models. arXiv:2501.05366.
- Li, X.; Jin, J.; Dong, G.; Qian, H.; Zhu, Y.; Wu, Y.; Wen, J.; and Dou, Z. 2025b. WebThinker: Empowering Large Reasoning Models with Deep Research Capability. *CoRR*, abs/2504.21776.
- Ma, X.; Gong, Y.; He, P.; Zhao, H.; and Duan, N. 2023. Query Rewriting in Retrieval-Augmented Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5303–5315. Singapore: Association for Computational Linguistics.
- Nakano, R.; Hilton, J.; Balaji, S.; Wu, J.; Ouyang, L.; Kim, C.; Hesse, C.; Jain, S.; Kosaraju, V.; Saunders, W.; Jiang, X.; Cobbe, K.; Eloundou, T.; Krueger, G.; Button, K.; Knight, M.; Chess, B.; and Schulman, J. 2022. WebGPT: Browser-assisted question-answering with human feedback. arXiv:2112.09332.
- OpenAI; ; Hurst, A.; Lerer, A.; Goucher, A. P.; and et al. 2024a. GPT-4o System Card. arXiv:2410.21276.
- OpenAI; ; Jaech, A.; Kalai, A.; Lerer, A.; and et al. 2024b. OpenAI o1 System Card. arXiv:2412.16720.
- Plaat, A.; Wong, A.; Verberne, S.; Broekens, J.; van Stein, N.; and Bäck, T. 2024. Reasoning with Large Language Models, a Survey. *CoRR*, abs/2407.11511.
- Qu, X.; Li, Y.; Su, Z.; Sun, W.; Yan, J.; Liu, D.; Cui, G.; Liu, D.; Liang, S.; He, J.; Li, P.; Wei, W.; Shao, J.; Lu, C.; Zhang, Y.; Hua, X.-S.; Zhou, B.; and Cheng, Y. 2025. A Survey of Efficient Reasoning for Large Reasoning Models: Language, Multimodality, and Beyond. arXiv:2503.21614.
- Robertson, S.; and Zaragoza, H. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.*, 3(4): 333–389.
- Sankararaman, H.; Yasin, M. N.; Sorensen, T.; Bari, A. D.; and Stolcke, A. 2024. Provenance: A Light-weight Fact-checker for Retrieval Augmented LLM Generation Output. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track (EMNLP)*, 1305–1313. Miami, Florida, US: Association for Computational Linguistics.
- Snell, C.; Lee, J.; Xu, K.; and Kumar, A. 2024. Scaling LLM Test-Time Compute Optimally can be More Effective than Scaling Model Parameters. *CoRR*, abs/2408.03314.
- Team, D. 2024a. Deepseek-r1-lite-preview is now live: unleashing supercharged reasoning power.

- Team, Q. 2024b. QwQ: Reflect Deeply on the Boundaries of the Unknown.
- Team, Q. 2025. QwQ-32B: Embracing the Power of Reinforcement Learning.
- Wang, X.; Wei, J.; Schuurmans, D.; Le, Q. V.; Chi, E. H.; Narang, S.; Chowdhery, A.; and Zhou, D. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations (ICLR)*.
- Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E. H.; Le, Q. V.; and Zhou, D. 2022. Chain-of-thought prompting elicits reasoning in large language models. NIPS '22. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713871088.
- Wei, J.; Xu, N.; Zhu, J.; Haoyanni; Wu, G.; Chen, Q.; Yu, B.; and Wang, L. 2025a. ChartMind: A Comprehensive Benchmark for Complex Real-world Multimodal Chart Question Answering. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4555–4569.
- Wei, K.; Sun, X.; Zhang, Z.; Zhang, J.; Zhi, G.; and Jin, L. 2021. Trigger is Not Sufficient: Exploiting Frame-aware Knowledge for Implicit Event Argument Extraction. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, 4672–4682. Online: Association for Computational Linguistics.
- Wei, K.; Yang, Y.; Jin, L.; Sun, X.; Zhang, Z.; Zhang, J.; Li, X.; Zhang, L.; Liu, J.; and Zhi, G. 2023. Guide the Many-to-One Assignment: Open Information Extraction via IoU-aware Optimal Transport. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 4971–4984. Toronto, Canada: Association for Computational Linguistics.
- Wei, K.; Zhong, J.; Zhang, H.; Zhang, F.; Zhang, D.; Jin, L.; Yu, Y.; and Zhang, J. 2025b. Chain-of-Specificity: Enhancing Task-Specific Constraint Adherence in Large Language Models. In Rambow, O.; Wanner, L.; Apidianaki, M.; Al-Khalifa, H.; Eugenio, B. D.; and Schockaert, S., eds., *Proceedings of the 31st International Conference on Computational Linguistics (ACL)*, 2401–2416. Abu Dhabi, UAE: Association for Computational Linguistics.
- Wen, Y.; Wang, Z.; and Sun, J. 2024. MindMap: Knowledge Graph Prompting Sparks Graph of Thoughts in Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 10370–10388. Bangkok, Thailand: Association for Computational Linguistics.
- Xu, F.; Shi, W.; and Choi, E. 2024. RECOMP: Improving Retrieval-Augmented LMs with Context Compression and Selective Augmentation. In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Yao, S.; Yu, D.; Zhao, J.; Shafran, I.; Griffiths, T. L.; Cao, Y.; and Narasimhan, K. 2023a. Tree of thoughts: deliberate problem solving with large language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NIPS)*, NIPS '23. Red Hook, NY, USA: Curran Associates Inc.
- Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2023b. ReAct: Synergizing Reasoning and Acting in Language Models. In *International Conference on Learning Representations (ICLR)*.
- Zhang, T.; Kishore\*, V.; Wu\*, F.; Weinberger, K. Q.; and Artzi, Y. 2020. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations (ICLR)*.
- Zhang, Y.; Wang, T.; Chen, S.; Wang, K.; Zeng, X.; Lin, H.; Han, X.; Sun, L.; and Lu, C. 2025. ARise: Towards Knowledge-Augmented Reasoning via Risk-Adaptive Search. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 10978–10995. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-251-0.
- Zhang, Z.; Zhang, A.; Li, M.; Zhao, H.; Karypis, G.; and Smola, A. 2024. Multimodal Chain-of-Thought Reasoning in Language Models. arXiv:2302.00923.
- Zhu, J.; Wang, J.; Yu, B.; Wu, X.; Li, J.; Wang, L.; and Xu, N. 2025a. TableEval: A Real-World Benchmark for Complex, Multilingual, and Multi-Structured Table Question Answering. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 7137–7157.
- Zhu, J.; Xiao, M.; Wang, Y.; Zhai, F.; Zhou, Y.; and Zong, C. 2025b. TROVE: A Challenge for Fine-Grained Text Provenance via Source Sentence Tracing and Relationship Classification. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 11755–11771. Association for Computational Linguistics.