A Multi-Agent LLM Framework with Hierarchical Citation Graph for Survey Generation

Minh-Anh Nguyen [*1,2] **Minh Duc Nguyen** [*2] **N.T. Ha Lan** [*2] **Hai Dang Kieu** [2] **Dong Tien Nguyen** [1,2]
Dung D. Le [2]

[*]Equal contribution [1]*CMC-OpenAI* [2]*VinUniversity*. Correspondence to: Dung D. Le dung.ld@vinuni.edu.vn.

1. Introduction

Large language models (LLMs) are increasingly adopted for automating survey paper generation [1, 2, 3, 4, 5]. Existing approaches typically extract content from a large collection of related papers and prompt LLMs to summarize them directly. However, such methods often overlook the structural relationships among papers, resulting in generated surveys that lack a coherent taxonomy and a deeper contextual understanding of research progress [6, 7, 8]. To address these shortcomings, we propose **SurveyG**, an LLM-based agent framework that integrates *hierarchical citation graph*, where nodes denote research papers and edges capture both citation dependencies and semantic relatedness between their contents, thereby embedding structural and contextual knowledge into the survey generation process. The graph is organized into three layers: **Foundation**, **Development**, and **Frontier**, to capture the evolution of research from seminal works to incremental advances and emerging directions. By combining horizontal search within layers and vertical depth traversal across layers, the agent produces multi-level summaries, which are consolidated into a structured survey outline. A multi-agent [9, 10] validation stage then ensures consistency, coverage, and factual accuracy in generating the final survey. Experiments, including evaluations by human experts and LLM-as-a-judge, demonstrate that SurveyG outperforms state-of-the-art frameworks, producing surveys that are more comprehensive and better structured to the underlying knowledge taxonomy of a field.

2. Methodology

We propose **SurveyG**, a two-stage automated survey generation framework comprising a *Preparation Phase* for graph-based literature structuring and a *Generation Phase* for survey synthesis.

2.1 Preparation Phase: Literature Structuring

Given a user query, SurveyG first expands it into a diverse set of keywords using an LLM and retrieves relevant papers. The retrieved literature is organized as a hierarchical citation graph $G = (V, E)$, where each node $v \in V$ represents a paper and edges $e \in E$ encode citation relationships augmented with semantic similarity computed from abstract embeddings. Each node v is associated with an attribute set $A(v)$, including bibliographic metadata and an LLM-generated paper-specific summary.

To capture the temporal and conceptual evolution of a research field, papers are automatically assigned to three layers:

- **Foundation layer** (V_1): seminal and highly cited works,
- **Development layer** (V_2): intermediate studies extending foundational ideas,
- **Frontier layer** (V_3): recent works reflecting emerging trends.

Horizontal Traversal. Within each layer V_l, SurveyG performs community detection using the Leiden algorithm, partitioning the layer into

$$C_l = \{C_{l,1}, \ldots, C_{l,m_l}\}, \quad \bigcup_{j=1}^{m_l} C_{l,j} = V_l.$$

Each community $C_{l,j}$ corresponds to a coherent research direction. For every community, an LLM generates a synthesized summary by aggregating the attributes of its member papers: $T_{l,j} = \text{LLM}(\{A(v_i) \mid v_i \in C_{l,j}\})$. These summaries capture dominant methodologies and thematic scopes within each layer, providing a global, layer-wise view of the research landscape.

Vertical Traversal. To model cross-layer dependencies and research evolution, SurveyG performs a weighted breadth-first search (WBFS) starting from each foundation paper. The traversal prioritizes semantically relevant edges and generates citation paths that span from foundation through development to frontier layers. For each path, node attributes $A(v)$ encountered during WBFS are aggregated and summarized hierarchically across layers, enabling incremental knowledge integration and mitigating long-context limitations. This vertical traversal yields path-specific summaries that explicitly reflect methodological progression and research trajectories.

Through horizontal and vertical traversals, SurveyG generates N layer-wise summaries $T_{l,j}$ and K path-wise summaries, capturing both field breadth and depth.

2.2 Generation Phase: Multi-Agent Survey Synthesis

In the generation phase, SurveyG employs a multi-agent LLM framework consisting of a *Writing Agent* and an *Evaluation Agent*. The Writing Agent constructs a structured survey outline grounded in the pre-computed horizontal and vertical summaries, while the Evaluation Agent iteratively critiques coherence, balance, and topical coverage. Through iterative refinement, the agents expand each section and optionally retrieve additional references when necessary. By combining hierarchical graph-based summarization with agent-based generation, SurveyG produces coherent and comprehensive survey papers

Evaluation Type	Model	Score Win Rate	Comparative Win Rate	Human Eval
Full Paper	SurveyForge	38.85%	27.75%	36.00%
	SurveyG (ours)	**61.15%**	**72.25%**	**64.00%**
Outline	SurveyForge	45.00%	42.00%	45.00%
	SurveyG (ours)	**55.00%**	**58.00%**	**55.00%**

Table 1: Win-rate comparison of SurveyForge and SurveyG on full-paper and outline evaluations. *Score Win Rate* denotes higher absolute scores, and *Comparative Win Rate* denotes pairwise LLM preferences.

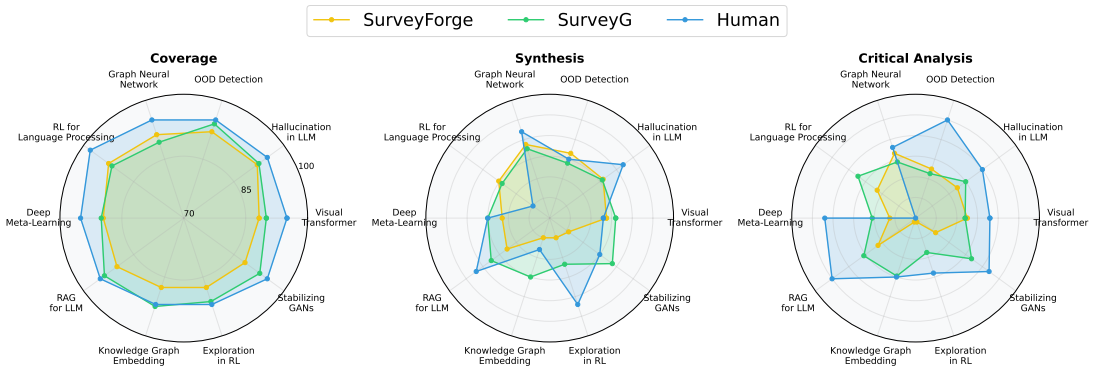

Fig. 1: LLM-as-a-judge evaluation of human-written ground-truth surveys, SurveyForge, and SurveyG across ten topics using GPT-4o as the evaluator.

that more faithfully reflect the logical structure and evolutionary dynamics of a research field compared to flat retrieval-based approaches.

3. Experimental Results

Baselines. We compare **SurveyG** with **SurveyForge** [3], a state-of-the-art survey generation system that leverages human-written surveys from related domains for heuristic outline construction and reference retrieval.

Dataset and Setup. We evaluate on ten diverse computer science topics from the **SurGE** benchmark [4], which contains 205 ground-truth surveys and over one million papers. For each topic, domain experts curated reference papers and selected a representative ground-truth survey. The same experts served as human evaluators. Following [3], we retrieve 1,500 candidate papers for outline generation and 60 papers per chapter during writing. All experiments use *GPT-4o-mini-2024-07-18* as the backbone model. We generate ten surveys per topic (100 total) and report averaged results.

Evaluation Metrics. We evaluate generated surveys along three dimensions: **(1) Outline Quality**, using the same prompt and protocol as SurveyForge; **(2) Content Quality**, assessed by three standard metrics: *Coverage*, *Synthesis*, and *Critical Analysis*, rated on a 0–100 scale by both LLM and human judges.

3.1 Evaluation Against Ground Truth

Figure 1 shows that **SurveyG** achieves more stable and balanced performance across topics than SurveyForge. While SurveyForge occasionally reaches higher peak scores, SurveyG exhibits lower variance, particularly in *Coverage* and *Critical Analysis*, and achieves synthesis quality closer to human-written surveys. These results indicate stronger robustness and generalization across diverse and specialized domains.

3.2 Human Evaluation

We conduct a blinded, win-rate–based comparison between **SurveyG** and **SurveyForge** across all ten topics. SurveyG outperforms SurveyForge with a *Score Win Rate* of 61.15%, a *Comparative Win Rate* of 72.25%, and a *Human Preference Rate* of 64.00%, demonstrating strong alignment between automated evaluation and expert judgment. A separate evaluation focused on outline quality further confirms SurveyG's advantage, achieving higher win rates across all metrics, validating the effectiveness of hierarchical citation graph–based retrieval for outline generation.

3.2.1 Inter-Rater Reliability

To assess evaluation reliability, two domain experts independently evaluated anonymized outputs for each topic. Cohen's κ indicates substantial agreement between LLM and human evaluators ($\kappa = 0.70$ for outlines, 0.61 for content) and stronger agreement between human raters ($\kappa = 0.75$ and 0.71, respectively), confirming the reliability of LLM-as-a-judge.

4. Conclusion

We present **SurveyG**, a hierarchical, multi-agent framework for automated survey generation that captures research structure and evolution via graph-based traversal. Experiments on the SurGE benchmark show that SurveyG consistently outperforms existing methods in both automated and human evaluations.

References

[1] Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Min Zhang, Qingsong Wen, et al. Autosurvey: Large language models can automatically write surveys, 2024. *URL https://arxiv.org/abs/2406.10252*.

[2] Xun Liang, Jiawei Yang, Yezhaohui Wang, Chen Tang, Zifan Zheng, Shichao Song, Zehao Lin, Yebin Yang, Simin Niu, Hanyu Wang, et al. Surveyx: Academic survey automation via large language models. *arXiv preprint arXiv:2502.14776*, 2025.

[3] Xiangchao Yan, Shiyang Feng, Jiakang Yuan, Renqiu Xia, Bin Wang, Bo Zhang, and Lei Bai. Surveyforge: On the outline heuristics, memory-driven generation, and multi-dimensional evaluation for automated survey writing. *arXiv preprint arXiv:2503.04629*, 2025.

[4] Weihang Su, Anzhe Xie, Qingyao Ai, Jianming Long, Jiaxin Mao, Ziyi Ye, and Yiqun Liu. Benchmarking computer science survey generation. *arXiv preprint arXiv:2508.15658*, 2025.

[5] Zhiyuan Wen, Jiannong Cao, Zian Wang, Beichen Guo, Ruosong Yang, and Shuaiqi Liu. Interactivesurvey: An llm-based personalized and interactive survey paper generation system. *arXiv preprint arXiv:2504.08762*, 2025.

[6] Shai Erera, Michal Shmueli-Scheuer, Guy Feigenblat, Ora Peled Nakash, Odellia Boni, Haggai Roitman, Doron Cohen, Bar Weiner, Yosi Mass, Or Rivlin, et al. A summarization system for scientific documents. *arXiv preprint arXiv:1908.11152*, 2019.

[7] Yutong Li, Lu Chen, Aiwei Liu, Kai Yu, and Lijie Wen. Chatcite: Llm agent with human workflow guidance for comparative literature summary. *arXiv preprint arXiv:2403.02574*, 2024.

[8] Teo Susnjak, Peter Hwang, Napoleon Reyes, Andre LC Barczak, Timothy McIntosh, and Surangika Ranathunga. Automating research synthesis with domain-specific large language model fine-tuning. *ACM Transactions on Knowledge Discovery from Data*, 19(3):1–39, 2025.

[9] Yusen Zhang, Ruoxi Sun, Yanfei Chen, Tomas Pfister, Rui Zhang, and Sercan Arik. Chain of agents: Large language models collaborating on long-context tasks. *Advances in Neural Information Processing Systems*, 37:132208–132237, 2024.

[10] Yushi Bai, Xin Lv, Jiajie Zhang, Yuze He, Ji Qi, Lei Hou, Jie Tang, Yuxiao Dong, and Juanzi Li. Longalign: A recipe for long context alignment of large language models, 2024. *URL https://arxiv.org/abs/2401.18058*.

Appendix A. Survey Topics

We compiled a collection of ten representative survey papers covering diverse research areas, as summarized in Table A1. Each topic reflects an active line of inquiry within machine learning and natural language processing, providing a strong foundation for evaluating literature review generation.

Appendix B. Prompt Templates

This section presents the prompt templates designed to guide each stage of automated literature review generation and evaluation. Each template specifies goals, inputs, and evaluation criteria to ensure consistency and quality across generated outputs.

2.1 Prompt to generate structured outline

We provide a short version of the prompt template (Figure A2) that instructs the model to construct a coherent, hierarchical outline that captures the logical flow of a literature review topic before detailed writing begins.prior to

2.2 Prompt to evaluate structured outline

The prompt in Figure A3 guides the model to write complete, citation-based literature review subsections grounded in the provided focus, summaries, and development directions. The following evaluation prompt extends this process to assess individual sections for depth, synthesis, and analytical quality.

2.3 Prompt to generate subsections

This prompt guides the model to write complete, citation-based literature review subsections grounded in the provided focus, summaries, and development directions (Figure A4).

2.4 Improve Section Quality

As shown in Figure A5, this prompt systematically assesses literature review sections across multiple dimensions—such as content coverage, synthesis, and critical analysis—while offering actionable feedback and retrieval suggestions for refinement.

Appendix C. Case studies

We provided a subsection generated by **SurveyG** (Figure A6) to illustrate its ability to synthesize complex research trends in modular and agentic RAG. Overall, this subsection highlights a clear progression in the RAG landscape from simple retrieval pipelines toward multi-stage, agentic, and modular architectures. The discussed works collectively show how LLMs are evolving from passive generators to proactive reasoning agents capable of planning, coordina-

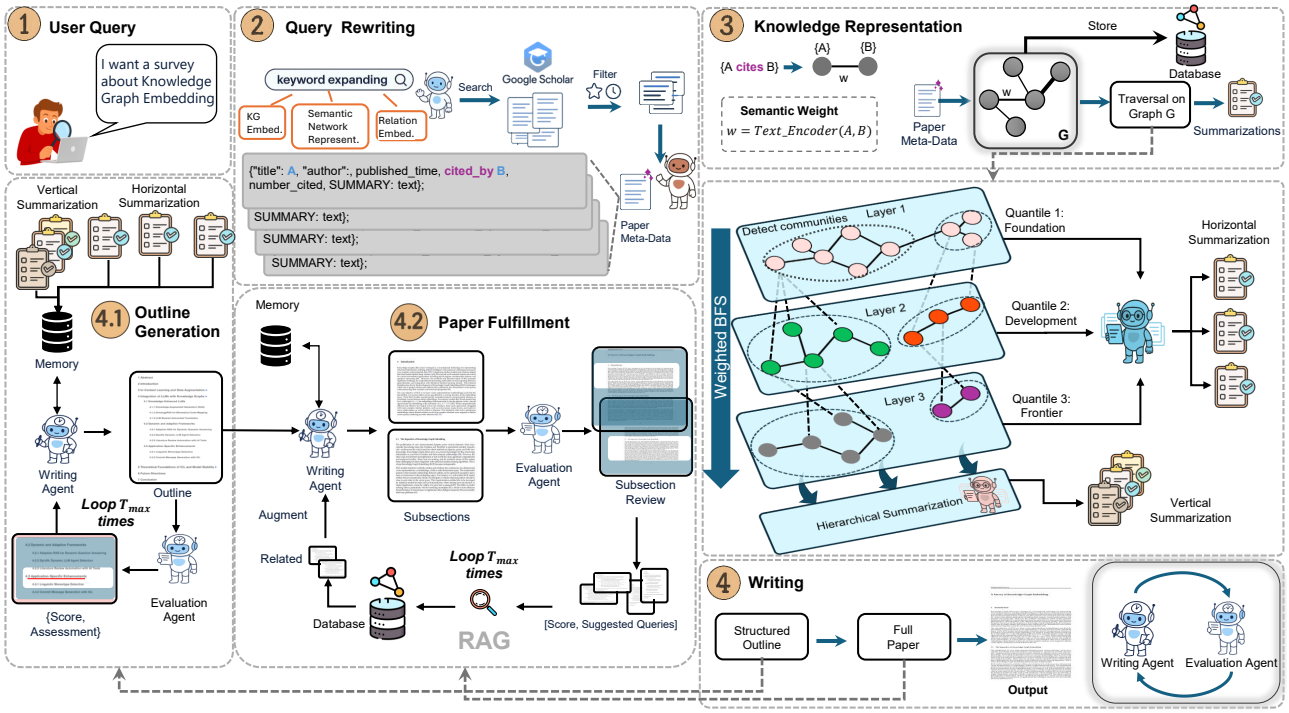

Fig. A1: Starting from a user's query, SurveyG retrieves and filters relevant papers (step 1-2), builds a hierarchical citation graph, and applies horizontal and vertical traversals to produce multi-aspect summaries (step 3). A multi-agent framework then leverages these pre-built summaries to produce a structured outline and a complete survey paper (step 4).

tion, and self-optimization. The emergence of meta-frameworks such as AutoRAG and FlashRAG further reflects a shift toward automated orchestration of RAG components, underscoring a broader trend toward unified, adaptive systems that integrate retrieval and reasoning for scalable knowledge synthesis.

Topic	Ground Truth Survey	Citation
Visual Transformer	A Survey of Visual Transformers	405
Hallucination in Large Language Models	Siren's Song in the AI Ocean: A Survey on Hallucination in LLMs	808
Graph Neural Networks	Graph Neural Networks: Taxonomy, Advances, and Trends	129
Deep Meta-Learning	A Survey of Deep Meta-Learning	459
Knowledge Graph Embedding	Knowledge graph embedding: A survey from the perspective of representation spaces	130
Generalized Out-of-Distribution Detection	Generalized Out-of-Distribution Detection: A Survey	1406
Reinforcement Learning for Language Processing	Survey on reinforcement learning for language processing	206
Exploration Methods in Reinforcement Learning	Exploration in Deep Reinforcement Learning: From Single-Agent to Multi-Agent Domain	194
Stabilizing Generative Adversarial Networks	Stabilizing Generative Adversarial Networks: A Survey	149
Retrieval-Augmented Generation for LLMs	Retrieval-Augmented Generation for Large Language Models: A Survey	953

Table A1: Survey Papers Overview

Goal: Generate a structured Literature Review Outline for: "`[QUERY]`"

INPUT SYNTHESIS DATA

- **Communities**: `[PAPER_COMMUNITIES]`
- **Directions**: `[DEVELOPMENT_DIRECTIONS]`

REQUIREMENTS & CONSTRAINTS

1. Structure:

- **Progression**: Follow Foundations → Core → Advanced → Applications → Future.
- **Mandatory Sections**: Must include Introduction, Foundational Concepts, and Conclusion.
- **Hierarchy**: Use exactly **TWO levels** (e.g., 2.1, 2.2). No deeper nesting.

2. Content & Quality:

- Create a **coherent narrative** (evolutionary story, not a list).
- Group material by **methodological families** and thematic depth.
- Include dedicated sections for Applications and Future Trends/Challenges.

3. Evidence & Output:

- **Proof IDs**: Each subsection **MUST** be grounded with 1-3 `proof_ids` (from `layer`, `community_X`, or `seed` IDs).
- **Focus Synthesis**: Provide `section_focus` (broad theme) and `subsection_focus` (specific details).
- **Format**: Return only a **JSON ARRAY** of main sections and their hierarchical subsections.

Fig. A2: Generate Outline Prompt.

Evaluate the quality and structure of the following literature review outline. Assess whether the outline demonstrates meaningful organization of works rather than a simple concatenation of summaries. **Your feedback should include:**

- Strengths of the outline.
- Weaknesses or issues (if any).
- Specific suggestions for improvement (only if issues are found).
- Final score (1-5, with 5 being the maximum) evaluate overall organization, coherence, and coverage.

Outline to evaluate: {outline_text}

Fig. A3: Prompt to evaluate structured outline

Task: Write a comprehensive literature review subsection titled `[SUBSECTION_TITLE]` in LaTeX.
Inputs:

- **Focus:** [SUBSECTION_FOCUS]
- **Community summaries:** [COMMUNITY_SUMMARY]
- **Development directions:** [DEVELOPMENT_DIRECTION]
- **Papers (chronological):** [PAPER_INFO]

Guidelines:

- Use LaTeX format with citations (`\cite{citation_key}`).
- Minimum 400 words, no numbered subsection titles.
- Focus strictly on the subsection topic.
- Each paper: 2–3 sentences describing technical contributions.
- Connect papers by showing how later work addresses earlier limitations.
- Conclude with remaining challenges or future directions.

Avoid: sequential listing, vague critiques, unsupported claims, isolated descriptions, or ignoring contradictions.

Fig. A4: Generate Outline Prompt

Evaluate the quality of the following literature review section within the context of the overall survey outline. Your evaluation should address the following aspects, each rated from 1-5 (5 = excellent): (1) content coverage, (2) citation density, (3) academic rigor, (4) synthesis across works, (5) critical analysis, (6) coherence, (7) depth of discussion, and (8) specificity of scope.
In addition to numeric ratings, provide:
- A brief natural language summary of the section's strengths and weaknesses,
- An overall score (1-5) with justification,
- Suggestions for improvement, focusing on areas that fall short,
- A list of search queries that could retrieve additional relevant literature to strengthen the section.

Fig. A5: Section Quality and Retrieval Prompt

Multi-stage and Modular RAG Frameworks

The foundational paradigm of Retrieval-Augmented Generation (RAG) typically operates on a straightforward "retrieve-then-generate" sequence [lewis2020pwr]. However, as Large Language Models (LLMs) are increasingly tasked with complex, multi-faceted queries and dynamic information needs, this simple pipeline proves insufficient [huang2024a59, zhao2024931]. This has spurred the evolution of RAG into more sophisticated, multi-stage, and modular architectures, where the LLM transcends a passive role to become an intelligent agent capable of proactive planning, dynamic decision-making, and the orchestration of various sub-tasks [gao20238ea]. This section focuses on frameworks that empower LLMs to actively manage the information-seeking process through iterative planning, query decomposition, and the dynamic assembly of specialized modules. It is crucial to distinguish these proactive, agentic approaches from reactive or corrective mechanisms (e.g., self-correction, re-ranking) that primarily refine retrieval quality, which are discussed in detail in Section 3.

A significant advancement in modular RAG involves empowering LLMs to act as sophisticated planning agents, iteratively refining their information-seeking process and orchestrating multi-round interactions. [lee2024hif] introduced PlanRAG, which extends the popular ReAct framework by incorporating explicit "Plan" and "Re-plan" steps. This allows LLMs to dynamically generate and iteratively refine analytical approaches based on intermediate retrieval results, effectively acting as decision-makers for complex data analysis tasks. Similarly, [yang20243nb] presented IM-RAG, a multi-round RAG system that leverages learned inner monologues and a multi-agent reinforcement learning approach. In IM-RAG, an LLM-based "Reasoner" dynamically switches between a "Questioner" role (crafting queries) and an "Answerer" role, guided by mid-step rewards from a "Progress Tracker," leading to flexible and interpretable multi-round information gathering. Building on the concept of autonomous interaction, [yu2024c32]'s Auto-RAG enables LLMs to engage in multi-turn dialogues with the retriever, systematically planning retrievals and refining queries until sufficient external information is gathered. This framework highlights the LLM's powerful decision-making capabilities, autonomously adjusting iterations based on query difficulty and knowledge utility. Another approach, [wang2024zt3]'s M-RAG, proposes a multi-partition paradigm for external memories, employing a multi-agent reinforcement learning framework with an "Agent-S" for dynamic partition selection and an "Agent-R" for memory refinement. This enables more fine-grained and focused retrieval by orchestrating memory access across different knowledge partitions. To further optimize the interaction between these modular components, [li20243nz]'s RAG-DDR (Differentiable Data Rewards) offers an end-to-end training method that aligns data preferences between different RAG modules (agents). By collecting rewards and evaluating the impact of perturbations on the entire system, RAG-DDR optimizes agents to produce outputs that enhance overall RAG performance, particularly for smaller LLMs. These agentic frameworks collectively transform RAG into a dynamic, adaptive system capable of tackling complex, multi-hop queries that require sophisticated reasoning and iterative information synthesis.

...

In conclusion, the evolution towards multi-stage and modular RAG frameworks marks a significant advancement, transforming RAG from a simple pipeline into an intelligent, adaptive system. By enabling LLMs to engage in iterative refinement, agentic planning, and dynamic orchestration of sub-tasks, these architectures enhance robustness, reduce hallucinations, and improve the depth and faithfulness of generated responses, particularly for complex, multi-hop queries [tang2024i5r]. However, this sophistication often introduces challenges related to increased computational overhead, the complexity of orchestrating multiple modules, and the need for robust evaluation methodologies that can accurately assess the contributions of each stage and the overall system performance. Benchmarks like [friel20241ct]'s RAGBench, [krishna2024qsh]'s FRAMES, and [tang2024i5r]'s MultiHop-RAG highlight these challenges, emphasizing the need for explainable metrics and unified frameworks to evaluate the intricate interplay of retrieval, reasoning, and generation in these advanced systems. Future research will likely focus on optimizing the efficiency of these multi-stage processes, developing more autonomous and self-correcting agents, and creating more generalized frameworks that can seamlessly integrate diverse knowledge sources and reasoning paradigms while addressing the inherent trade-offs between complexity and efficiency.

Fig. A6: Case studies about the result of generated subsection.