Dawn at SemEval-2026 Task 8: Structured Control Decomposition for Faithful Multi-Turn Retrieval-Augmented Generation

Feiling Li[1], Xiaoya Qi[1], Xunyue Wang[1], Pusheng Chen[1], Zhiwen Tang[1], Han Yang[2]*,
[1]School of Information Science and Engineering, Yunnan University
[2]The First Affiliated Hospital of Kunming Medical University
{lifeiling.stu,qixiaoya.stu, wangxunyue.stu, chenpusheng.stu}@ynu.edu.cn
zhiwen.tang@ynu.edu.cn, yanghanantennae@163.com

Abstract

Multi-turn Retrieval-Augmented Generation faces structural challenges that go beyond single-turn retrieval and fusion. Context-dependent queries, cross-turn evidence accumulation, and uncertain answerability jointly affect retrieval quality and generation reliability. We propose a structured control framework that formulates multi-turn RAG as a regulated reasoning process rather than a loosely coupled pipeline. The system first performs evidence and context structuring, extracting atomic facts strictly grounded in reference passages while reconstructing a self-contained query from dialogue history. It then conducts decision-conditioned generation, where explicit control signals regarding question intent, dialogue dependency, and answerability govern response feasibility, scope, and organization. By separating structural decision making from surface realization, the framework enforces consistent information flow across stages and reduces hallucination.Experiments on SemEval-2026 Task 8 show that our approach achieves strong faithfulness and stable overall performance, ranking 17/26 on Task B (generation, H=0.6333).

1 Introduction

Knowledge-intensive natural language processing tasks, such as open-domain question answering and dialogue systems, require models not only to generate fluent text but also to ground their outputs in verifiable evidence. Despite the impressive parametric memory of large language models, purely generative approaches remain vulnerable to hallucination when relevant facts are absent or implicitly referenced. Retrieval-Augmented Generation (RAG) mitigates this limitation by coupling generation with external knowledge retrieval (Lewis et al., 2020). With the advent of dense retrieval methods such as DPR (Karpukhin et al., 2020) and improved evidence fusion strategies including Fusion-in-Decoder and generation-augmented retrieval (Izacard and Grave, 2021; Mao et al., 2021), single-turn open-domain QA has achieved substantial gains in factual accuracy.

However, the assumptions underlying single-turn RAG break down in multi-turn conversational settings. In dialogue, user queries are rarely self-contained; they frequently rely on prior turns through anaphora, ellipsis, or implicit topic continuation. When retrieval is performed solely on the surface form of the current query, the system effectively searches with incomplete semantic information, resulting in degraded recall and downstream answer quality (Mo et al., 2023; Wu et al., 2022). Query rewriting methods attempt to restore semantic completeness by reformulating context-dependent queries into standalone forms, using generative models or reinforcement learning objectives (Ma et al., 2023; Mo et al., 2023; Wu et al., 2022; Zhang et al., 2024). While these approaches improve retrieval robustness, they typically treat rewriting as an isolated preprocessing step, without jointly modeling its interaction with downstream retrieval and generation.

Beyond query ambiguity, multi-turn dialogue introduces a second structural challenge: evidence must be accumulated, updated, and selectively integrated across turns. As conversations evolve, relevant information may emerge incrementally, and naive retrieval at each turn can ignore previously retrieved evidence or introduce contradictions. Although multi-passage fusion improves single-turn evidence utilization (Izacard and Grave, 2021), there remains a lack of systematic mechanisms for cross-turn evidence management and controlled accumulation. Moreover, when retrieval fails to produce reliable support, unconstrained generation can propagate errors across turns, amplifying factual inconsistencies. The answerability modeling paradigm introduced in SQuAD 2.0 (Rajpurkar et al., 2018) underscores the importance

*Corresponding Author

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 2894–2902

of reliability-aware decision making, and recent analyses of multi-turn RAG systems (Wang et al., 2024; Cheng et al., 2024) and benchmarks such as MTRAG-UN (Rosenthal et al., 2026a) further highlight the need for coordinated retrieval, accumulation, and generation control.

These observations suggest that effective multi-turn RAG requires more than incremental improvements to individual components. Instead, it demands a unified architecture that explicitly models the interdependence among query rewriting, retrieval enhancement, and reliability-aware generation. SemEval-2026 Task 8 (Task B) (Rosenthal et al., 2026c) builds upon recent benchmarks such as MTRAG (Katsis et al., 2025), providing a rigorous evaluation setting for this problem, emphasizing the downstream impact of query rewriting and cross-turn reasoning in conversational retrieval-augmented systems.

In this work, we present a unified RAG framework tailored for multi-turn dialogue. The framework integrates three tightly coupled modules: (1) a generative query rewriting component that restores contextual completeness; (2) a multi-turn retrieval enhancement mechanism that accumulates and fuses cross-turn evidence in a controlled manner; and (3) a rule-guided generation strategy incorporating answerability assessment and posterior filtering to prevent unsupported responses. By explicitly coordinating these stages, our approach transforms multi-turn RAG from a loosely connected pipeline into a structured, reliability-aware reasoning process. Experimental results on the SemEval benchmark demonstrate consistent improvements over strong baselines, and ablation studies reveal clear synergistic effects between contextual rewriting and cross-turn evidence accumulation.

2 Related Work

Retrieval-Augmented Generation RAG has become a standard framework for knowledge-intensive NLP. By integrating neural retrieval with generative models (Lewis et al., 2020) and leveraging dense retrievers such as DPR (Karpukhin et al., 2020) together with Fusion-in-Decoder (Izacard and Grave, 2021), prior work significantly improves factual grounding in open-domain QA. Pre-training strategies such as REALM and Atlas (Guu et al., 2020; Izacard et al., 2022) and generation-augmented retrieval (Mao et al., 2021) further strengthen retrieval and generation interaction. However, these approaches are mainly evaluated in single-turn settings with self-contained queries.

In conversational scenarios, context-dependent queries containing anaphora or ellipsis degrade retrieval performance. Benchmarks such as TREC CAsT (Dalton et al., 2020) motivate query rewriting methods including CONQRR (Wu et al., 2022), ConvGQR (Mo et al., 2023), and LLM-based rewriting (Ma et al., 2023; Zhang et al., 2024). Although effective for restoring semantic completeness, rewriting is typically optimized independently of downstream retrieval dynamics and generation behavior.

Multi-turn dialogue also requires cross-turn evidence accumulation and reliability control. Existing RAG variants focus on passage-level fusion within a single turn (Lewis et al., 2020), while benchmarks such as CORAL (Cheng et al., 2024) reveal persistent performance gaps in conversational settings. At the same time, answerability modeling (Rajpurkar et al., 2018) and factuality studies (Maynez et al., 2020) highlight the necessity of reliability-aware generation when evidence is insufficient.

Overall, prior work improves rewriting, retrieval, or reliability modeling in isolation. A principled mechanism that regulates information flow across turns and coordinates context restoration, evidence accumulation, and controlled generation remains underexplored, motivating our approach for this task.

3 Methodology

3.1 Structured Control Formulation for Multi-Turn RAG

Multi-turn RAG differs from standard single-turn RAG in that the current query is often not a complete information request by itself. Let $H = \{u_1, \ldots, u_{t-1}\}$ denote the dialogue history, q_t the current user query, and $P = \{p_1, \ldots, p_k\}$ the provided reference passages. A conventional retrieve-then-generate formulation directly models the answer as $y = G(H, q_t, P)$. However, this formulation leaves several key decisions implicit inside the generator, including which facts in P are relevant, how q_t should be interpreted under H, whether the question is answerable, and what response structure should be adopted. In multi-turn settings, such implicit decision making can easily lead to incomplete context resolution, unsupported evidence use, and hallucinated answers.

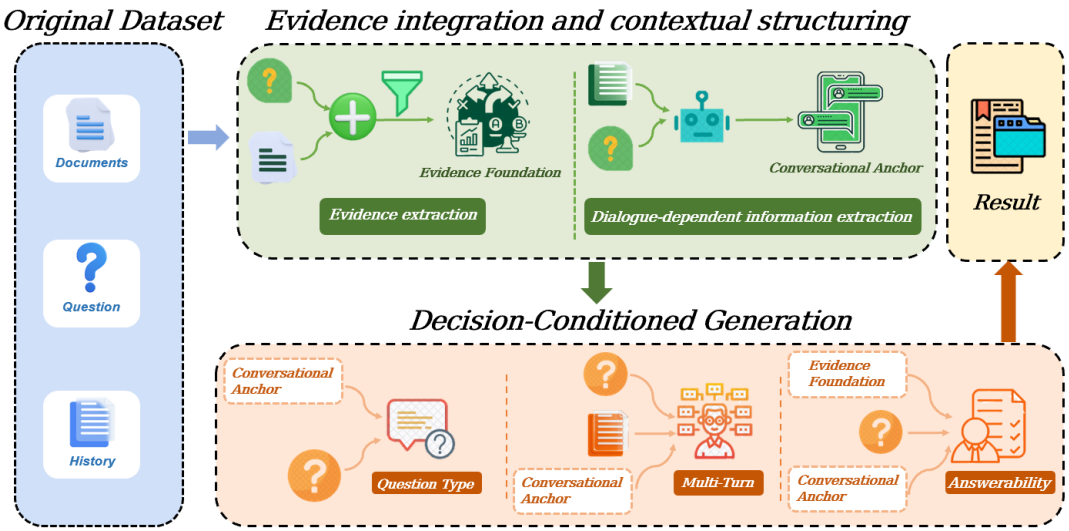


Figure 1: System Overview.

To address this issue, we formulate multi-turn RAG as a structured control process. The core idea is to decompose answer generation into a sequence of intermediate representations and explicit control variables, so that generation is performed under well-defined evidence, context, and decision constraints. Specifically, the system first constructs structured representations from the input:

$$(E, q_t') = S(H, q_t, P) \tag{1}$$

where $E = \{e_1, \ldots, e_n\}$ denotes a set of atomic evidence units extracted from the reference passages, and q_t' denotes a standalone query reconstructed from the dialogue history and the current query. The system then predicts a set of control variables:

$$z = D(E, q_t'), \tag{2}$$

where $z = (z_a, z_c, z_m)$ represents answerability, question category, and multi-turn dependency, respectively. Finally, the answer is generated as:

$$y = G(E, q_t', z). \tag{3}$$

Under this formulation, structured control refers to the explicit regulation of information flow across these stages. Evidence structuring determines what factual content may be used; context reconstruction determines how the current query should be interpreted; and decision variables determine whether and how the answer should be produced. This design does not rely on the generator to infer all structural properties implicitly. Instead, it exposes these properties as intermediate states, making the generation process more constrained, interpretable, and easier to diagnose.

The overall workflow is shown in Figure 1. Starting from the dialogue history, current query, and reference passages, the system first derives a grounded evidence representation and a context-resolved query. These outputs are then used to obtain control signals for answerability, question category, and dialogue dependency. The final response is produced only after these intermediate results have been made explicit, ensuring that surface realization is conditioned on both the available evidence and the structural requirements of the question.

3.2 Evidence and Context Structuring

The first step of structured control is to convert the original input into an explicit and grounded reasoning basis. In multi-turn RAG, the generator cannot safely operate on the raw tuple (H, q_t, P), because the current query q_t may contain unresolved references, while the passages P may include both relevant and irrelevant information. Directly feeding these inputs into the generator increases the risk of context omission, evidence misuse, and unsupported inference. Therefore, this stage aims to construct two intermediate representations: a structured evidence set E and a context-resolved query q_t'.

Formally, this stage implements the structuring function:

$$(E, q_t') = S(H, q_t, P) \tag{4}$$

which consists of two sub-functions:

$$E = \mathrm{Extract}(q_t, P), \qquad q'_t = \mathrm{Rewrite}(H, q_t). \tag{5}$$

The extraction function identifies question-relevant evidence from the reference passages. Each evidence unit $e_i \in E$ is required to satisfy three constraints: it must be explicitly supported by P, express verifiable facts, and avoid external knowledge or speculative inference. In this way, $E = \{e_1, \ldots, e_n\}$ serves as the admissible factual basis for subsequent decision making and generation, rather than a loose summary of the passages.

The rewriting function resolves the contextual incompleteness of the current query. Given the dialogue history H and the current query q_t, $\mathrm{Rewrite}(H, q_t)$ produces a standalone query q'_t by resolving referential expressions, recovering omitted constraints, and making implicit requirements explicit. For example, pronouns, ellipses, and follow-up expressions are replaced or completed using antecedents and constraints available in H. Importantly, this process is constrained to use only the dialogue history and is not allowed to introduce new factual assumptions.

These two operations are intentionally separated. Evidence extraction determines what information can be used, while query rewriting determines what information is being requested. Their outputs are then jointly used by the downstream decision module, so that answerability and generation are evaluated against both the available evidence E and the resolved user intent q'_t. In our implementation, both $\mathrm{Extract}(\cdot)$ and $\mathrm{Rewrite}(\cdot)$ are realized through constrained LLM prompts. The full prompts used for evidence extraction and context reconstruction are provided in Appendix A to support reproducibility.

3.3 Decision-Conditioned Generation

After obtaining the structured representations (E, q'_t), the remaining challenge is to determine how the answer should be produced under different question conditions. In multi-turn settings, even with correct evidence and a fully resolved query, the generation process still involves implicit decisions, such as whether the question is answerable, what type of response is expected, and how strongly the answer should depend on prior dialogue. If these decisions are left entirely to the generator, they are entangled with surface realization and become difficult to control, often leading to inconsistent response styles or unsupported content.

To address this issue, we explicitly separate decision making from generation by introducing a set of control variables. Formally, we define a decision function:

$$z = D(E, q'_t), \tag{6}$$

where $z = (z_a, z_c, z_m)$ corresponds to answerability, question category, and multi-turn dependency, respectively. Concretely, z_a indicates whether the question is answerable given E, e.g., ANSWERABLE, PARTIAL, UNANSWERABLE; z_c captures the structural intent of the question, e.g., factoid, explanation, or procedural; and z_m characterizes the relationship between the current query and the dialogue history, e.g., clarification, follow-up, or standalone. Each component of z is obtained through a zero-shot LLM classification prompt, and the exact prompt specifications are provided in Appendix A.

These control variables do not directly determine the answer content, but instead impose structured constraints on the generation process. Specifically, z_a defines a feasibility constraint that governs whether a direct answer can be produced and how uncertainty should be expressed; z_c defines a structural constraint that controls the organization and level of detail of the response; and z_m defines a contextual constraint that determines how the reconstructed query q'_t should incorporate dialogue history. In this way, the generation process is not a free-form mapping from (E, q'_t) to y, but a constrained mapping conditioned on z:

$$y = G(E, q'_t, z). \tag{7}$$

In our implementation, G is realized as a prompt-based generation module that takes (E, q'_t, z) as input and produces the final answer under strict grounding constraints, i.e., only information contained in E is allowed to appear in the output. Importantly, the control variables z function as soft constraints rather than hard routing decisions. Even when z is partially incorrect, the generator still has access to the full evidence set E and the resolved query q'_t, which prevents catastrophic failure. Instead, z primarily biases the answer format, scope, and contextual integration, making the overall system robust to imperfect decision signals.

By decoupling structural decision making from surface realization, this stage ensures that the final answer is consistently aligned with both the available evidence and the structural requirements of

the query. This explicit conditioning mechanism enables controllable and interpretable generation behavior, while preserving flexibility across diverse multi-turn scenarios.

4 Experiments

4.1 Experiment Setup

We conduct experiments on the SemEval-2026 Task 8 dataset, which consists of multi-turn dialogues ending with a user question and a set of reference passages. Following the official setting, we split the development set into 80% for training and 20% for validation, and report final results on the test set of 507 samples(Rosenthal et al., 2026b).

GPT-4o-mini is used as the generation backbone. Given the dialogue history and retrieved passages, the model produces grounded responses with the goal of minimizing hallucination and improving factual accuracy.

We evaluate performance using the official metrics: RBalg, the harmonic mean of Bert-Recall, RougeL, and Bert-K-Prec; RBllm, an LLM-based metric grounded in reference passages from RAD-Bench; and Rlf, the faithfulness metric from RAGAS. These metrics jointly assess answer quality and grounding fidelity.

4.2 Main Results

Table 1: Comparison of $\mathrm{RB}_{\mathrm{alg}}$ performance of different models on the development set

Model	$\mathrm{RB}_{\mathrm{alg}}$ **Mean**
qwen3:8b (zero-shot)	0.1148
qwen3:8b (post-trained)	0.3242
mistral:7b-v0.3 (zero-shot)	0.2691
mistral:7b-v0.3 (post-trained)	0.4270
llama3 (zero-shot)	0.2588
llama3 (post-trained)	0.3678
GPT-4o-mini (zero-shot)	0.4876

We first conduct model selection on the development set using a unified pipeline, varying only the generator. Using $\mathrm{RB}_{\mathrm{alg}}$ as the selection metric, GPT-4o-mini achieves the highest score of 0.4876 in zero-shot mode, outperforming all open-source instruction-tuned models, including fine-tuned variants. We therefore adopt GPT-4o-mini as the core generator in our system.

On the official test set of 507 tasks, our system achieves a harmonic mean of 0.6333, with RBagg 0.4876, RLF 0.7537, and RBllm 0.7357, ranking 17th out of 26 teams. Compared to the strong baseline `gpt-oss-120b` (OpenAI, 2025) at 0.639, the gap is only 0.0057. While not among the top-ranked systems, our model performs particularly well on RLF and $\mathrm{RB}_{\mathrm{llm}}$, indicating strong grounding and low hallucination.

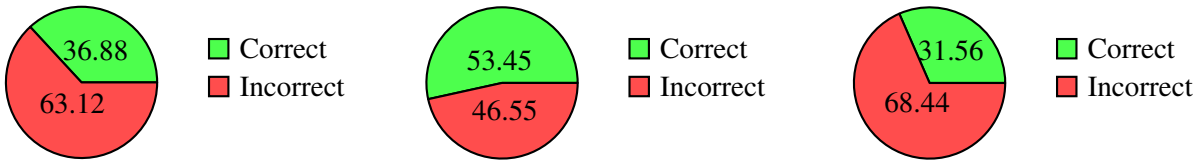
These results suggest two main findings. First, GPT-4o-mini provides a clear advantage in leveraging reference evidence, which translates into strong faithfulness performance in the full system. Second, the proposed architecture substantially improves overall performance beyond single-step generation, with explicit intermediate control signals contributing to the final harmonic mean of 0.6333. The remaining gap to the top system mainly lies in $\mathrm{RB}_{\mathrm{agg}}$, indicating that improving lexical alignment and evidence coverage remains a key direction for future work.

4.3 Error Analysis

To gain deeper insight into the limitations of our approach and to inform iterative improvements, we conducted a detailed error analysis on the test set. By comparing the generated answers with the reference passages, we identified three predominant error types: hallucination, missing evidence, and a uniform response strategy that fails to adapt to different question characteristics. Additionally, we examined the performance of the three explicit classifiers (question type, answerability, and multi-turn) to understand their contribution to the overall generation quality.

We first evaluated the accuracy of the system on the three classification tasks. As shown in Figure 2, the question type classifier achieves only 36.88% accuracy, indicating substantial difficulty in distinguishing nuanced query intents such as Factoid, Explanation, or How-To. The answerability classifier performs better (53.45%), but still misclassifies nearly half of the instances, often confusing PARTIAL with ANSWERABLE or UNANSWERABLE. The multi-turn classifier has the lowest accuracy (31.56%), reflecting the complexity of resolving coreferences and recovering omitted context in conversational dialogues. These classification errors directly propagate to the generation stage, contributing to hallucination and missing evidence.

We conduct error analysis on the test set and identify three main issues: hallucination, missing evidence, and a uniform response strategy that fails



(a) Question Type (b) Answerability (c) Multi-Turn

Figure 2: Accuracy of the three explicit classifiers on the test set.

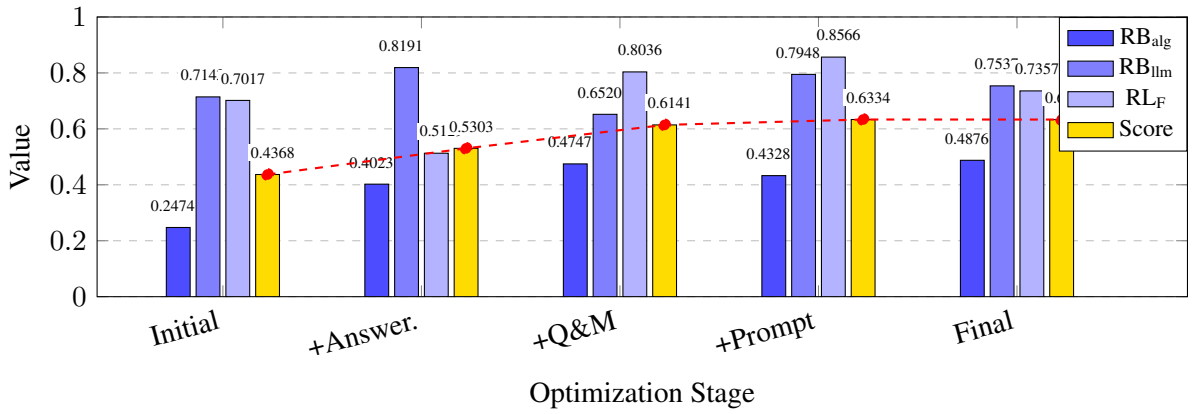


Figure 3: Progression of system performance across optimization stages. Blue bars (darker to lighter) represent RB_{alg}, RB_{llm}, and RL_F; yellow bars indicate the overall Score. The red dashed line (with markers) is aligned with the yellow bars to show the Score trend.

to adapt to question characteristics. We also examine the three explicit classifiers to understand their impact on generation quality.

The question type, answerability, and multi-turn classifiers achieve accuracies of 36.88%, 53.45%, and 31.56%, respectively. Errors are common, especially in distinguishing fine-grained intents and resolving conversational dependencies. These misclassifications propagate to generation, contributing to unsupported content and incomplete answers.

In the initial system, hallucination and evidence omission are frequent, and the system applies a uniform response pattern regardless of question type or answerability, resulting in a harmonic mean of 0.55 on the development set. To address these issues, we enforce stricter evidence extraction, require atomic fact coverage, and introduce explicit control signals from the three classifiers to guide final generation. These modifications progressively improve performance, raising the harmonic mean to 0.614 after prompt refinement and to 0.6333 after incorporating classification signals.

Overall, the analysis shows that hallucination and omission largely stem from weak control over evidence use and question characteristics. Strengthening evidence constraints and introducing structured decision signals significantly improves faithfulness and overall performance.

5 Conclusion

We present a structured control formulation for multi-turn RAG that emphasizes regulated information flow across evidence extraction, context reconstruction, and answer generation. Instead of independently optimizing rewriting, retrieval, or generation, our framework explicitly models question characteristics and answer feasibility to guide downstream behavior. Through strict grounding in reference passages and decision-conditioned generation, the system reduces hallucination and improves faithfulness in conversational settings. Experimental results confirm that structured decision signals substantially enhance performance beyond unconstrained generation, particularly in grounding-related metrics. Future work will focus on improving intent classification accuracy and refining evidence coverage to further narrow the gap with top-performing systems.

Acknowledgements

The work is supported by the Yunnan Provincial Philosophy and Social Sciences Planning Project (QN202564), Tsang Hin-chi Education Foundation, and the Undergraduate Innovation Training Program of Yunnan University (University-level, No. 20257145).

References

Yiruo Cheng, Kelong Mao, Ziliang Zhao, Guanghui Dong, Hao Qian, Yixin Wu, Tetsuya Sakai, Ji-Rong Wen, and Zhicheng Dou. 2024. CORAL: Benchmarking multi-turn conversational retrieval-augmentation generation. *arXiv preprint arXiv:2410.23090*.

Jeff Dalton, Chenyan Xiong, and Jamie Callan. 2020. Trec cast 2019: The conversational assistance track overview. In *Proceedings of the 29th Text REtrieval Conference (TREC 2019)*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning (ICML 2020)*, pages 3929–3938.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open-domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2021)*, pages 874–880.

Gautier Izacard, Edouard Grave, Teven Le Scao, Angela Fan, and Armand Joulin. 2022. Atlas: Few-shot learning with retrieval augmented language models. In *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, pages 6478–6492.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Yannis Katsis, Sara Rosenthal, Kshitij Fadnis, Chulaka Gunasekara, Young-Suk Lee, Lucian Popa, Vraj Shah, Huaiyu Zhu, Danish Contractor, and Marina Danilevsky. 2025. Mtrag: A multi-turn conversational benchmark for evaluating retrieval-augmented generation systems. *Transactions of the Association for Computational Linguistics*, 13:784–808.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Vladimir Karpukhin, Naman Goyal, Henrik Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and Sebastian Riedel. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, pages 9459–9474.

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

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. Generation-augmented retrieval for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL 2021)*, pages 4089–4100.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pages 1906–1919.

Fengran Mo, Kelong Mao, Yutao Zhu, Yixin Wu, Kaiyu Huang, and Jian-Yun Nie. 2023. ConvGQR: Generative query reformulation for conversational search. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL 2023)*, pages 4998–5012.

OpenAI. 2025. gpt-oss-120b benchmark model. Used in MTRAG benchmark.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SquAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pages 784–789.

Sara Rosenthal, Yannis Katsis, Vraj Shah, Lihong He, Lucian Popa, and Marina Danilevsky. 2026a. Mtrag-un: A benchmark for open challenges in multi-turn rag conversations. *Preprint*, arXiv:2602.23184.

Sara Rosenthal, Yannis Katsis, Vraj Shah, Lihong He, Lucian Popa, and Marina Danilevsky. 2026b. Mtrag-un: A benchmark for open challenges in multi-turn rag conversations. *Preprint*, arXiv:2602.23184.

Sara Rosenthal, Vraj Shah, Yannis Katsis, and Marina Danilevsky. 2026c. Semeval-2026 task 8: Mtrageval: Evaluating multi-turn rag conversations. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*, San Diego, California. Association for Computational Linguistics.

Xiaohua Wang, Zhenghua Wang, Xuan Gao, Fuzhen Zhang, Yixin Wu, Zhen Xu, and Tao Shi. 2024. Searching for best practices in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP 2024)*, pages 17716–17736.

Zeqiu Wu, Yi Luan, Hannah Rashkin, David Reitter, Hannaneh Hajishirzi, Mari Ostendorf, and Gaurav S Tomar. 2022. CONQRR: Conversational query rewriting for retrieval with reinforcement learning. In *Proceedings of the 2022 Conference on Empirical*

Methods in Natural Language Processing (EMNLP 2022), pages 10000–10014.

Tianhua Zhang, Kun Li, Hongyin Luo, Xiaodong Wu, James R Glass, and Helen Meng. 2024. Adaptive query rewriting: Aligning rewriters through marginal probability of conversational answers. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP 2024)*, pages 13444–13461.

A Prompt Templates

Prompt 1: Reference Evidence Extraction

You are a Reference Evidence Extractor. Given the user's latest question and reference passages, extract ALL useful supporting evidence.
Extraction Rules:
- Use ONLY information explicitly stated in the passages. Do not infer or speculate.
- Extract comprehensively: include all key facts, details, and supporting context.
- Evidence must be atomic: each point expresses one verifiable fact. Group related facts.

User's latest question: "{INPUT_TEXT}"
Reference passages: "{REFERENCE}"
Please extract and organize the evidence.
Output: "{EVIDENCE}"

Prompt 2: Conversation Context Extraction

You are a conversation context extractor. Extract ALL necessary contextual information from conversation history to interpret the final question, then generate a complete, explicit version of the user's last question.
Extraction Rules (based solely on history):
- Resolve pronouns and ambiguous references.
- Restore omitted context or constraints.
- Identify user preferences (format, language, style, scope, etc.).
Prohibited: introducing new facts, assumptions, or inferences beyond explicit statements.

User's latest question: "{INPUT_TEXT}"
History: "{HISTORY_TEXT}"
Please extract context and generate a complete, explicit version of the user's question.
Output: "{CONTEXT_QUERY}"

Prompt 3: Question Type Classification

You are a question type classifier. Output exactly one of:
- Factoid: specific fact (name, date, number).
- Explanation: cause/effect or background.
- Summarization: core points, ignore details.
- How-To: steps or methods.
- Non-Question: greeting, thanks, etc.
- Keyword: keyword expecting expansion.
- Composite: multiple sub-questions.
- Comparative: compare entities.
- Opinion: subjective evaluation.
- Troubleshooting: problem, causes, solutions.

Context query: "{CONTEXT_QUERY}"
User's latest question: "{INPUT_TEXT}"
Output only one word.
Output: "{QUESTION_TYPE}"

Prompt 4: Multi-Turn Classification

You are a multi-turn conversation classifier. Output exactly one of:
- Clarification: references previous context to resolve ambiguity within the same topic.
- Follow-up: extends or deepens the previous topic, introduces sub-topic or next step.
- N/A: unrelated to history, standalone.

History: "{HISTORY_TEXT}"
Context query: "{CONTEXT_QUERY}"
User's latest question: "{INPUT_TEXT}"
Output only one word.
Output: "{MULTI_TURN}"

Prompt 5: Answerability Classification

You are an answerability classifier. Output exactly one of:
- UNANSWERABLE: no relevant info in documents or history.
- CONVERSATIONAL: non-information-seeking input (greeting, acknowledgment).

- ANSWERABLE: explicit evidence in documents for a complete answer.
- PARTIAL: limited info; only some relevant details available.

Reference evidence: "{EVIDENCE}"
Context query: "{CONTEXT_QUERY}"
User's latest question: "{INPUT_TEXT}"
Output only one word.
Output: "{ANSWERABILITY}"

Prompt 6: Final Answer Generation

You are a faithful QA assistant. Answers must be fully grounded in provided reference evidence only. Adapt content and style according to answerability, question_type, and multi_turn.

- Use ONLY provided evidence. Do not guess or fabricate.
- Prefer directly citing evidence.

Step 1 – Follow answerability:
- ANSWERABLE: complete answer using all evidence. No "I don't know".
- UNANSWERABLE: clearly state lack of info.
- PARTIAL: provide supported parts, explicitly mention missing info.
- CONVERSATIONAL: polite natural reply, no factual grounding.

Step 2 – Adapt structure by question_type:
Factoid: short; Explanation: causal; Summarization: condensed; How-To: steps; Composite: address all sub-questions; Comparative: compare; Troubleshooting: causes/solutions; Opinion: evidence-supported; Keyword: expand; Non-Question: brief reply.

Step 3 – Adapt context by multi_turn:
Clarification: reference earlier context; Follow-up: continue naturally; N/A: standalone.

Reference evidence: "{EVIDENCE}"
Context query: "{CONTEXT_QUERY}"
User's latest question: "{INPUT_TEXT}"
Answerability: "{ANSWERABILITY}"
Question_type: "{QUESTION_TYPE}"
Multi_turn: "{MULTI_TURN}"
Please provide your answer following the above rules.
Output: "{ANSWER}"