MEETINGQA2: A Generative Speaker-Centric Benchmark for Multi-Party Dialogue Question Answering

Yuhan Chen[1]*, Yanling Li[1]*, Bowei Zou[2], Yifan Fan[1], Yu Hong[1]†
[1]School of Computer Science and Technology, Soochow University, Suzhou, China
[2]Institute of Advanced Intelligence and Computing (IAIC), A*STAR, Singapore
{cyhhh1121, li4861988, yifanfannlp, tianxianer}@gmail.com
zou_bowei@a-star.edu.sg

Abstract

Multi-party Dialogue Question Answering (MDQA) requires the machine to answer questions based on a conversation with multiple interlocutors. Existing benchmarks are largely extractive, leaving positional shortcuts and providing limited evidence about whether models can synthesize distributed evidence into speaker-grounded answers. We introduce MEETINGQA2, the first generative MDQA benchmark, derived from MEETINGQA (Prasad et al., 2023) with reformulated questions and speaker-centric summary answers. It contains 1,033 QA pairs, including 108 unanswerable questions. Beyond standard overlap metrics, we introduce attribution accuracy and factual correctness. Experiments show that strong LLMs perform consistently worse on MEETINGQA2 than on MEETINGQA, with especially large drops in attribution accuracy. The results indicate persistent weaknesses in speaker attribution, factual grounding, and long-context dialogue understanding. [1]

1 Introduction

Recent advancements in conversational AI have spurred a growing interest in multi-party dialogue research, particularly in QA systems (She et al., 2024; Li et al., 2023). However, existing datasets such as Molweni (Li et al., 2020) and FriendsQA (Yang and Choi, 2019) remain largely extractive, focus on short answers, and assume that a question has a single localized answer. These assumptions overlook the inherent complexity of real-world discussions, in which multiple speakers often express divergent perspectives. In this case, meaningful answers often span multiple utterances and involve contributions from more than one speaker.

MEETINGQA (Prasad et al., 2023) represents an important step toward this setting by allowing multi-span and multi-speaker answers, but two limitations remain. First, its questions are taken directly from dialogue utterances, leaving positional cues that models can exploit instead of demonstrating genuine comprehension. Second, its answers are still extractive, limiting evaluation of abstractive reasoning over long conversational contexts (Pang et al., 2022; Soleimani et al., 2021). In addition, conventional automatic metrics may have limited sensitivity to speaker attribution errors. This motivates the introduction of supplementary diagnostic metrics that more directly characterize the challenges posed by MEETINGQA2.

To bridge this research gap, we present MEETINGQA2, a generative MDQA benchmark derived from the MEETINGQA test set. Questions are reformulated with GPT-4o[2], and answers are rewritten as speaker-centric summaries that separate and condense each speaker's contribution. The dataset contains 1,033 QA pairs, including 108 unanswerable questions, and the longest dialogue spans 19,225 words. MEETINGQA2 preserves the multi-speaker and multi-span structure of MEETINGQA while removing positional shortcuts and requiring abstractive, speaker-aware synthesis over long contexts.

The contributions of our work include: 1) MEETINGQA2 is, to the best of our knowledge, the first generative, speaker-centric QA benchmark derived from multi-party meeting transcripts; 2) We introduce attribution accuracy and LLM-as-judge factual correctness metrics to compare MEETINGQA2 against MEETINGQA across different slices; and 3) We show that current LLMs exhibit substantial limitations in speaker grounding and factual understanding on this setting.

*Equal contribution.
†Corresponding author
[1]The source code is publicly available at https://github.com/cyh0208/MeetingQA2
[2]https://openai.com/index/hello-gpt-4o/

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 36957–36967
October 24-29, 2026 ©2026 Association for Computational Linguistics

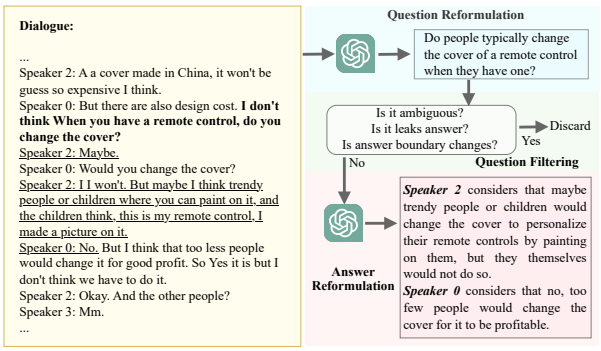


Figure 1: Data construction process for answerable cases in MEETINGQA2. The bold and underlined sentences in the yellow box represent the questions and corresponding answers from MEETINGQA, respectively.

2 Our Dataset: MEETINGQA2

2.1 Data Construction

As shown in Figure 1, our dataset creation framework employs a hybrid approach combining LLM-powered rewriting with multi-stage quality assurance. It is built through question reformulation, question filtering, and answer reformulation.

To keep generation grounded, we use the short-context version of MEETINGQA during construction. At evaluation time, however, models are tasked with answering these questions using the complete and full-length dialogue transcript, thereby directly assessing their capabilities in long-context comprehension.

2.1.1 Question Reformulation

Building upon the extractive QA framework of MEETINGQA, we first convert extracted questions Q into their non-extracted counterparts Q'. Specifically, to enhance GPT's ability to resolve referential and elliptical expressions, we first prepend 20 preceding dialogue turns to each sample. To avoid contamination from post-question context, we place the source utterance of Q as the final turn in the input sequence. The resulting input consists of three components: (1) the expanded dialogue context, (2) the original question Q, and (3) the associated answer span(s) A. This enriched context is subsequently formatted into a structured prompt for few-shot question generation using GPT-4o. The model is explicitly instructed to incorporate referential and elliptical cues when reformulating questions. Complete prompt details appear in Table 7.

2.1.2 Question Filtering

To ensure high quality of questions, we develop a multi-stage filtering system to address the issues of ambiguity resolution, answer leakage, and answer boundary changes.

Ambiguity Resolution Detection Even after question reformulation, some questions remain semantically vague (e.g., the original question "*Or are you going to prime them in any way?*" is transformed into "*Are you planning to influence the participants with the language choice or content of the voice-over in any way?*", retaining ambiguity "*you*"). To address this, we exclude any reformulated question Q' containing ungrounded pronouns (e.g., "*he*", "*you*") without explicit speaker attribution. This step retains 76.6% of the samples.

Answer Leakage Detection In the question reformulation process, answer spans A are input into GPT-4o to generate Q', which may inadvertently reveal the answers. To mitigate this risk, we discard samples where the ROUGE-L recall between Q' and A is ≥ 0.4, preserving 80.9% of the dataset. The effects of alternative leakage detection strategies are detailed in Appendix B.

Answer Boundary Detection In MEETINGQA, questions are constrained to the ongoing discussion topic, which limits answers to a specific topical scope. In contrast, our reformulated questions may broaden the answer scope beyond the original spans by referencing the conversation as a whole. Detailed explanations are shown in the Appendix C.

To address this issue, we implement a two-stage GPT-assisted method filter. First, we perform backward context extension by 50 consecutive utterances. Second, we instruct GPT-4o to localize answer boundaries by identifying specific utterance IDs within the extended context, so as to identify samples with a larger range of answers. The prompt is shown in Table 8. This automated screening identifies 189 potentially over-extended samples among 1,061 candidates. Expert review then confirms 136 of them as genuine overextension cases and removes them from the final dataset.

2.1.3 Answer Reformulation

In the final stage, GPT-4o generates speaker-centric summary-style answers A' through structured prompt engineering. Specifically, the input includes (1) the expanded dialogue comprising 20 preceding conversational turns, (2) the reformulated question Q' and (3) the original answer span A. They are formatted into a highly structured prompt that requires speaker-ordered summaries, as detailed in Table 9.

2.1.4 Unanswerable Question Construction

Our approach to constructing unanswerable questions differs slightly from that for answerable ones. Specifically, it comprises three primary steps. First, we filter out ambiguous questions that often lead to inconsistent or fabricated outputs during GPT generation. In particular, questions with fewer than two meaningful words after stopword removal are discarded. Second, we reformulate the remaining questions using GPT-4o as described in Section 2.1.1. Third, we use GPT-4o to exclude questions that become answerable after reformulation. The prompt is shown in Table 10. Finally, we obtain 108 rigorously unanswerable question samples.

2.2 Quality Verification

To evaluate the quality of the GPT-generated QA pairs, we manually inspect 300 randomly sampled instances (approximately 30% of the dataset). Three expert annotators focus on evaluating the factual correctness and consistency of each QA pair. Disagreements are resolved through discussion, and a sample is considered correct only when at least two annotators agree that the rewritten question and answer are factually grounded and speaker-consistent. This verification process achieves a 95% accuracy rate, indicating that the majority of the GPT-generated QA pairs are highly accurate and reliable for practical use.

We further conduct a qualitative error analysis of the remaining 5% incorrect cases and categorize them into the following types: a) factual hallucination (1%), where the answer introduces unsupported claims not grounded in the source; b) minor content omission (1%), where a relevant speaker-related detail is partially missing, leading to a slightly incomplete response; and c) stylistic awkwardness (3%), where the QA content is factually plausible but exhibits unnatural or awkward phrasing (e.g., overly rigid formulations), reducing fluency and readability. These findings suggest that most errors are attributable to surface-level expression rather than substantive factual inconsistency, while hallucination and omission errors occur relatively rarely in our dataset.

2.3 Statistics and Analysis

Table 1 presents a comparison between MEETINGQA and MEETINGQA2. The two datasets have nearly identical dialogue length, but MEETINGQA2 has longer average questions and answers because reformulation resolves ellipsis and coreference and converts spans into summaries. Their multi-span and multi-speaker ratios remain comparable, indicating that reformulation changes answer form rather than discourse structure. The smaller size of MEETINGQA2 reflects our quality-oriented filtering pipeline, whose reliability is supported by the manual evaluation described in Section 2.2.

	MEETINGQA	MEETINGQA2
Formulation	Span	Free-form
Question count	2,467	1,033
Avg/Max dialog length	8,755.7/19,225	8,741.2/19,225
Avg/Max question length	11.6/155	17.6/41
Avg/Max answer length	40.5/228	46.6/144
Multi-span answer ratio	38.7%	38.6%
Multi-speaker answer ratio	49.1%	48.1%

Table 1: Comparison between test sets of MEETINGQA and MEETINGQA2.

3 Experimentation

3.1 Settings

We evaluate Meta's LLaMA3 series (Dubey et al., 2024), GPT-3.5 Turbo[3], and GPT-4o[2] under three context settings. In **Zero-Shot Inference (ZSI)**, the model directly receives the dialogue and question. For models with limited context windows, we apply dynamic RoPE scaling or input truncation. In **Retrieval-Augmented Generation (RAG)**, each dialogue is divided into 512-token chunks with five-utterance overlaps, and BGE-large retrieves the top three chunks for answer generation, achieving 77.8% retrieval recall. In **Standard Context Answer Generation (SCG)**, models receive the localized context used during answer reformulation, providing an oracle-context condition that isolates answer-generation performance from retrieval errors. Full implementation details are provided in Appendix D.

For the cross-benchmark comparison, we additionally evaluate Claude-4.6[4], Gemini-2.5 (Comanici et al., 2025), and GPT-4o on both MEETINGQA and MEETINGQA2. To analyze structural difficulty, we define four slices: SS (single-speaker, single-span), SM (single-speaker, multi-span), MS (multi-speaker, single-span), and MM (multi-speaker, multi-span). This partition characterizes samples according to speaker cardinality and the dispersion of supporting evidence.

[3] https://openai.com/index/gpt-3-5-turbo-fine-tuning-and-api-updates/

[4] https://www.anthropic.com/

Model	B4	RL	BS	MS
		ZSI		
LLaMA3.2-3B	8.83	26.08	26.05	20.63
LLaMA3-8B	12.24	29.51	29.98	22.03
LLaMA3.1-8B	**15.96**	**35.10**	**37.26**	**26.30**
GPT-3.5 Turbo	7.81	21.01	21.13	16.94
GPT-4o	11.32	27.99	30.86	24.03
		RAG		
LLaMA3.2-3B	12.25	30.01	31.19	23.51
LLaMA3-8B	**17.35**	**35.16**	**37.23**	**27.44**
LLaMA3.1-8B	16.84	33.68	35.58	26.05
GPT-3.5 Turbo	9.48	23.91	26.07	20.66
GPT-4o	12.85	28.94	32.23	24.84
		SCG		
LLaMA3.2-3B	15.57	33.14	34.54	25.81
LLaMA3-8B	18.59	**36.59**	38.67	**28.85**
LLaMA3.1-8B	**19.02**	36.43	**38.73**	28.51
GPT-3.5 Turbo	10.99	25.17	28.41	22.19
GPT-4o	15.79	32.73	36.60	28.28

Table 2: Performance on MEETINGQA2 under different experimental configurations (B4: BLEU-4, RL: ROUGE-L, BS: BERTScore, MS: MoverScore).

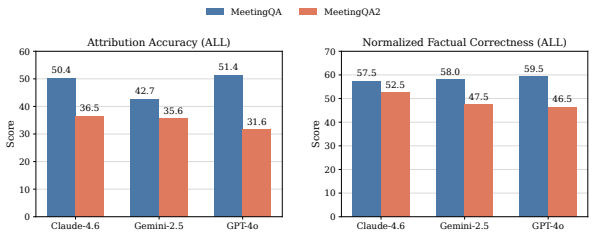


Figure 2: Overall comparison between MEETINGQA and MEETINGQA2 using strong proprietary LLMs.

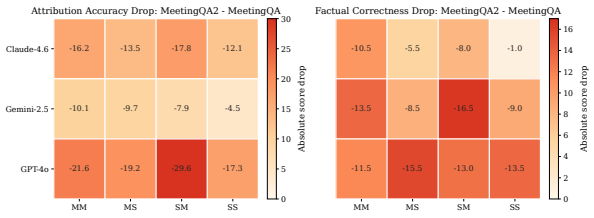


Figure 3: Slice-wise performance drop between MEETINGQA and MEETINGQA2.

We report BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), BERTScore (Zhang et al., 2019), and MoverScore (Zhao et al., 2019). For the cross-benchmark analysis, we further use two targeted metrics. *Attribution accuracy* counts a prediction as correct only when a predicted fact matches a gold fact in content and is assigned to the correct speaker, thereby measuring speaker-to-fact binding. *Normalized factual correctness* uses GPT-5.4 as an LLM judge to assign a score in $\{0, 1, 2\}$ according to semantic agreement with the gold facts. We report the dataset-level mean after linearly rescaling. These diagnostics complement standard automatic metrics by separately measuring attribution fidelity and semantic faithfulness. The judge is run with temperature 0 and structured JSON output. Additional implementation details appear in Appendix D.1.

3.2 Main Results

Table 2 reports standard generative metrics on MEETINGQA2. Even with oracle SCG context, performance remains modest, with the best ROUGE-L and BERTScore reaching 36.59 and 38.73. These metrics are informative but incomplete because MEETINGQA2 requires abstractive, speaker-centric answers, overlap-based scores do not fully reflect speaker grounding and may diverge from human judgment. We therefore treat Table 2 as a coarse comparison and rely on the diagnostic evaluation in Section 3.3 for a more faithful analysis.

Three patterns are consistent. First, RAG helps models without strong long-context pretraining: BERTScore rises by 5.14 points for LLaMA3.2-3B and 7.25 points for LLaMA3-8B, but only marginally for LLaMA3.1-8B. Second, SCG consistently outperforms RAG, showing that retrieval remains a bottleneck in multi-party settings. Third, under these standard automatic metrics, GPT models lag behind the strongest open-source long-context models, likely because they generate longer and more elaborated answers (Figure 4), which further motivates the diagnostic comparison below.

3.3 Diagnostic Comparison with MEETINGQA

To isolate the effect of reformulation, we compare MEETINGQA2 and MEETINGQA using Attribution Accuracy and Factual Correctness. Full numerical results are reported in Appendix Table 6, and Figures 2 and 3 summarize the main trends.

Figure 2 shows a consistent performance gap between MEETINGQA and MEETINGQA2 across all three proprietary models and both diagnostics. Relative to MEETINGQA, MEETINGQA2 reduces attribution accuracy by 13.88, 7.11, and 19.79 points for Claude-4.6, Gemini-2.5, and GPT-4o, respectively. The corresponding drops in factual correctness are 5.00, 10.50, and 13.00 points. Since the underlying meeting transcripts are identical, these degradations suggest that question reformulation and speaker-centric summary targets substantially increase the difficulty of the task. The two diagnostics reveal complementary error patterns. The average drop is larger for attribution accuracy than

for factual correctness, at 13.59 versus 9.50 points. This indicates that models often preserve part of the coarse semantic content but fail to bind the recovered facts to the correct speakers. In other words, the major challenge introduced by MEETINGQA2 is not only factual understanding, but also speaker-to-fact grounding in abstractive answers.

Figure 3 further breaks down the performance changes by structural slices. For attribution accuracy, the largest average degradation appears on SM, where answers involve multiple spans from a single speaker. This suggests that the difficulty of MEETINGQA2 cannot be attributed solely to speaker disambiguation, as models must also aggregate dispersed evidence from the same speaker into a coherent summary. For factual correctness, the largest average degradation appears on SM and MM, indicating that both cross-span aggregation and speaker-aware reasoning contribute to factual errors. Overall, the results demonstrate that MEETINGQA2 imposes additional demands on evidence aggregation, factual faithfulness, and speaker attribution. The larger degradation in attribution accuracy further suggests that current LLMs are more capable of recovering answer content than reliably assigning that content to the correct interlocutors.

3.4 Error Analysis

These diagnostic patterns are consistent with manual inspection of 100 randomly sampled answers generated by LLaMA3.1-8B, where a response may exhibit multiple error types. Only 16% of responses are fully correct, mostly on relatively simple yes/no questions with single-speaker or single-span evidence. Speaker-related errors are the most frequent, occurring in 40% of responses; within this subset, 90% omit at least one relevant speaker and 50% contain speaker misattribution. Factual inaccuracies appear in 24% of responses. In addition, 10% of responses are incomplete, 6% are irrelevant, and 4% incorrectly answer unanswerable questions. Overall, the dominant failures involve speaker grounding, factual consistency, and answerability recognition.

4 Conclusion

We introduce MEETINGQA2, a generative MDQA benchmark with speaker-centric summary answers. Diagnostic comparisons with MEETINGQA show that this reformulation increases task difficulty, especially in speaker grounding. MEETINGQA2 therefore offers a focused benchmark for multi-party dialogue understanding.

Limitations

While MEETINGQA2 establishes a novel benchmark in MDQA systems, it exhibits several notable constraints that warrant attention. Primarily, the dataset's limited scale restricts its capacity to support extended training periods essential for developing robust neural models. This scale is partly a consequence of our design choice to prioritize benchmark reliability over sheer quantity: each QA pair is produced via a multi-stage pipeline that requires multiple LLM invocations and stringent filtering (e.g., reformulation, ambiguity filtering, answer leakage detection, boundary detection, and generative answer summarization), which is computationally and financially costly. As a result, we focus on constructing a smaller set of dense, complex, and carefully verified QA pairs. Moreover, our dialogue sources solely focus on meeting transcripts, overlooking other types of conversational transcripts like forum discussions, TV series transcripts, etc., potentially constraining the utility of MEETINGQA2. Addressing these limitations requires future research in the field of MDQA.

Acknowledgements

This research was supported by National Natural Science Foundation of China under No.62376182.

References

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. 2005. The ami meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction*, pages 28–39.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the SIGKDD 2024*, pages 6491–6501.

Sarah E. Finch and Jinho D. Choi. 2024. ConvoSense: Overcoming monotonous commonsense inferences for conversational AI. *Transactions of the Association for Computational Linguistics*, pages 467–483.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Deepanway Ghosal, Siqi Shen, Navonil Majumder, Rada Mihalcea, and Soujanya Poria. 2022. CICERO: A dataset for contextualized commonsense inference in dialogues. In *Proceedings of the ACL 2022*, pages 5010–5028.

Hengli Li, Song-Chun Zhu, and Zilong Zheng. 2023. Diplomat: a dialogue dataset for situated pragmatic reasoning. volume 36, pages 46856–46884.

Jiaqi Li, Ming Liu, Min-Yen Kan, Zihao Zheng, Zekun Wang, Wenqiang Lei, Ting Liu, and Bing Qin. 2020. Molweni: A challenge multiparty dialogues-based machine reading comprehension dataset with discourse structure. In *Proceedings of the COLING 2020*, pages 2642–2652.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proceedings of the ACL 2004*, pages 74–81.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, and Samuel Bowman. 2022. QuALITY: Question answering with long input texts, yes! In *Proceedings of the NAACL 2022*, pages 5336–5358.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the ACL 2002*, pages 311–318.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2024. Yarn: Efficient context window extension of large language models. In *Proceedings of the ICLR 2024*, pages 1–20.

Archiki Prasad, Trung Bui, Seunghyun Yoon, Hanieh Deilamsalehy, Franck Dernoncourt, and Mohit Bansal. 2023. MeetingQA: Extractive question-answering on meeting transcripts. In *Proceedings of the ACL 2023*, pages 15000–15025.

Jin Qian, Bowei Zou, Mengxing Dong, Xiao Li, AiTi Aw, and Yu Hong. 2022. Capturing conversational interaction for question answering via global history reasoning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2071–2078.

Shuaijie She, Shujian Huang, Xingyun Wang, Yanke Zhou, and Jiajun Chen. 2024. Exploring the factual consistency in dialogue comprehension of large language models. In *Proceedings of the NAACL 2024*, pages 6087–6100.

Siqi Shen, Deepanway Ghosal, Navonil Majumder, Henry Lim, Rada Mihalcea, and Soujanya Poria. 2022. Multiview contextual commonsense inference: A new dataset and task. *arXiv preprint arXiv:2210.02890*.

Amir Soleimani, Christof Monz, and Marcel Worring. 2021. NLQuAD: A non-factoid long question answering data set. In *Proceedings of the EACL 2021*, pages 1245–1255.

Kai Sun, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi, and Claire Cardie. 2019. DREAM: A challenge data set and models for dialogue-based reading comprehension. *Transactions of the Association for Computational Linguistics*, pages 217–231.

Xindi Wang, Mahsa Salmani, Parsa Omidi, Xiangyu Ren, Mehdi Rezagholizadeh, and Armaghan Eshaghi. 2024. Beyond the limits: A survey of techniques to extend the context length in large language models. *arXiv preprint arXiv:2402.02244*.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the SIGIR 2024*, pages 641–649.

Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Retrieval meets long context large language models. In *Proceedings of the ICLR 2024*, pages 1–16.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Zhengzhe Yang and Jinho D Choi. 2019. FriendsQA: Open-domain question answering on tv show transcripts. In *Proceedings of the SIGDIAL 2019*, pages 188–197.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. *arXiv preprint arXiv:1909.02622*.

A Related Work

A.1 Multi-party Dialogue Question Answering

Recent years have witnessed the emergence of MDQA datasets, primarily sourced from two domains: chit-chat exchanges in TV shows (Sun et al., 2019; Yang and Choi, 2019) and dialogues in the Ubuntu Chat Corpus (Li et al., 2020). However, these works suffer from notable limitations: (1) Human-generated questions often lead to shallow queries with brief, single-speaker answers. (2) Dialogue contexts are typically short, averaging under 400 words. Other dialogue QA datasets construct generative QA pairs using manual annotations (Ghosal et al., 2022; Shen et al., 2022) or GPT (Finch and Choi, 2024), but their dialogues are limited to two-person conversations.

MEETINGQA (Prasad et al., 2023) attempts to address these limitations by constructing QA pairs from meeting transcripts (Carletta et al., 2005), allowing for multi-speaker and multi-span answers. However, it still has constraints: (1) Questions are directly derived from dialogue utterances, creating positional biases that allow models to rely on sequential patterns instead of showcasing true understanding. (2) Answers are strictly extractive, limiting the assessment of models' ability to synthesize information.

Our work builds upon the MEETINGQA foundation but extends it to a generative QA setting, enabling more comprehensive evaluation of LLMs' conversational understanding capabilities.

A.2 Question Answering on Long Context

Contemporary studies reveal that QA models exhibit limitations in accurately responding to questions requiring comprehension of extended contextual information (Pang et al., 2022; Soleimani et al., 2021; Qian et al., 2022). To address this challenge, researchers have developed two primary mitigation strategies. The first approach employs RAG frameworks (Gao et al., 2023; Xu et al., 2024; Fan et al., 2024), which implement a two-stage process involving relevant paragraph retrieval followed by context-aware answer generation. The second solution leverages parameter-efficient length extrapolation techniques (Peng et al., 2024; Wang et al., 2024), with empirical evidence demonstrating significant improvements in model performance on long-text comprehension tasks. Moreover, contemporary LLMs are increasingly integrating progressive context window expansion during their pre-training phases to bolster their capabilities in long-context reasoning. This strategy is used by prominent models including LLaMA (Dubey et al., 2024) and Qwen (Yang et al., 2024), which systematically increase their context window sizes through scaled training methodologies.

In our experiments, we utilize both RAG frameworks and length extrapolation techniques, specifically dynamic RoPE scaling, to evaluate their impact on performance. Additionally, we employ LLMs that are adept at handling varying lengths of text to ensure comprehensive analysis.

	TBF	LAV
Recall	100%	80%
Precision	29.41%	12.12%

Table 3: Comparative performance evaluation of answer leakage detection methods.

B Impact of Different Answer Leakage Detection Methods

To ensure the quality of question formulation, we investigate two distinct methods for identifying and eliminating questions with answer leakage. The first approach (**TBF: Threshold-based Filtering**) establishes a rigorous criterion by discarding samples exhibiting ROUGE-L recall ≥ 0.4 between the reformulation question Q' and original answer spans A. The second strategy (**LAV: LLM-assisted Validation**) implements a filtering mechanism through a task-specific few-shot prompting strategy with GPT-4. The complete prompt structure is detailed as Table 11.

We conduct a comparative evaluation using 100 randomly sampled instances to assess detection effectiveness. Table 3 presents the performance metrics, demonstrating our ultimate implementation of the threshold-based approach.

C Supplementary Explanations on Answer Boundary Detection

The motivation for performing answer boundary detection stems from the latent assumption of extractive QA pairs in MEETINGQA. In MEETINGQA, questions inherently relate to the ongoing discussion topic, naturally confining answers within that scope. However, reformulated questions may broaden the answer scope beyond the original spans by targeting the entirety of a lengthy conversation.

For instance, the original question "*Now have we decided anything, are we doing anything?*" is a sentence within the dialogue and carries positional context, it pertains specifically to a certain topic. In contrast, the reformulated question "*Have Speaker 3 and Speaker 4 made any decisions or decided on any actions to take?*" is a free-form question without dialogue location information. It is for all discussion topics. Consequently, the scope of possible answers shifts accordingly.

D Additional Experimental Settings

Zero-Shot Inference (ZSI) In this configuration, models generate answers directly from dialogue and question. Detailed prompt is shown in Table 12. For long context constrained models like LLaMA3-8B, we use dynamic RoPE scaling[5] for sequence length extrapolation. For GPT-3.5 Turbo suited for tokens up to 16k, we post-truncate tokens exceeding 16k. Remaining models employ standard zero-shot inference.

Retrieval-Augmented Generation (RAG) Given the extensive length of meeting transcripts, a suitable solution is RAG (Fan et al., 2024; Gao et al., 2023). The method involves three steps. First, each dialogue is divided into 512-token segments using sliding windows with 5-utterance overlaps, preserving conversational coherence while averting data fragmentation. Second, we utilize the BGE-large retriever (Xiao et al., 2024) to compute the dot product between query embeddings and pre-indexed chunks, returning top-3 relevant chunks. This step achieves 77.8% recall on MEETINGQA2. Finally, the retrieved chunks are combined with the question to facilitate answer generation by models.

Standard Context Answer Generation (SCG) In this setting, the context from the answer reformulation stage is provided to eliminate retrieval bias, simulating a 100% recall scenario to isolate model performance from retrieval limitations.

Cross-benchmark Comparison To compare MEETINGQA and MEETINGQA2 directly, we evaluate Claude-4.6, Gemini-2.5, and GPT-4o with the same full-context prompting configuration on both datasets. We keep the dialogue transcript and question as input and compute diagnostic metrics on the generated answers. To analyze which structures are most affected by the reformulation, we reuse the original MEETINGQA annotations to partition examples into SS, SM, MS, and MM, where S/M denote single/multi speaker and single/multi span, respectively. This setup isolates the effect of question and answer reformulation from changes in slice definitions.

D.1 Metrics

The evaluation criteria we employ can be categorized as follows: (1) N-gram Metrics: BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) measure lexical overlap between generated and reference texts. (2) Semantic Metrics: BERTScore (Zhang et al., 2019) and MoverScore (Zhao et al., 2019). BERTScore leverages BERT's (Devlin et al., 2019) contextual embeddings to capture semantic similarity, and we rescale it to enhance interpretability[6]. MoverScore extends BERTScore through optimal transport theory, enabling soft many-to-one word alignments by minimizing transformation costs between text pairs.

For the cross-benchmark comparison, we further introduce two diagnostic metrics. **Attribution Accuracy** averages speaker-level scores over the union of gold-relevant and predicted speakers, assigning 1, 0.5, or 0 for fully correct, partially correct, or incorrect attribution, respectively, with omitted and hallucinated speakers receiving 0. **Factual Correctness** uses GPT-5.4 as an LLM judge to score how well the predicted answer matches the gold facts in meaning. Specifically, the original factual correctness score assigned by the LLM judge lies in $\{0, 1, 2\}$ and is linearly normalized. This normalization makes factual correctness directly comparable with attribution accuracy in terms of score points. In our implementation, we set the temperature to 0 and enable JSON-format output to improve evaluation consistency. For robustness, each judge call is retried up to three times when parsing or errors occur, with a one-second interval between retries. We report the average judge score over the test set in Table 6, where a higher value indicates better semantic faithfulness to the reference facts. The prompts we use for LLM-as-judge are shown in Table 4 and Table 5.

[5] https://www.reddit.com/r/LocalLLaMA/comments/14mrgpr/dynamically_scaled_rope_further_increases/

[6] https://github.com/Tiiiger/bert_score/blob/master/journal/rescale_baseline.md

You are an expert evaluator for multi-party dialogue QA. Evaluate whether the model answer correctly summarizes speaker-specific opinions.
Steps:
1. Identify all relevant speakers needed to answer the question.
2. Extract speakers mentioned in the model prediction.
3. Speaker Attribution Accuracy (SAA):
For each relevant speaker:
    1 = fully faithful
    0.5 = partially correct
    0 = incorrect or hallucinated
Then compute the average.
Return STRICT JSON with this schema:
"relevant speakers": [xxx],
"predicted speakers": [xxx],
"speaker attribution scores": {
    "Speaker X": 1
},
"average attribution score": xxx,
"hallucination": "No",
"brief explanation": "..."

Table 4: Prompt for evaluating Attribution Accuracy.

You are evaluating a model answer for multi-party dialogue QA. Score the prediction using the rubric below. Return strict JSON only.
Rubric (0-2):
    factual correctness (How well the predicted answer matches the gold facts in meaning.)
Scoring guide:
    0 = poor
    1 = partial
    2 = good
Return schema:{
    "factual correctness": 0,
    "rationale": "..."
}
Question/query: {question}
Gold speakers: {gold speakers}
Gold facts: {gold facts}
Predicted output: {prediction}

Table 5: Prompt for evaluating Factual Correctness.

E Prompts

The prompts we use in our method are shown in Table 7 8 9 10 11 12.

Given the dialogue a question and answers, you need to rewrite the question to generate a complete and clear question to the answer. There should be no ambiguity or omission in the rewritten question. If pronouns appear in the question, they need to refer back to the correct entities. Let's think step by step.
Output the generated content as follows:
Thinking Process: xxx
Rewrite Question: xxx

Table 7: Question reformulation prompt.

Given a dialogue and a question, identify the utterances that provide sufficient evidence to answer the question. Only output the utterance IDs of utterances that can answer the question.

Table 8: Answer boundary detection prompt.

Given the dialogue, a question and answers of corresponding speakers, you need to generate summaries of the answers. There should be no ambiguity or omission in the summaries. If a summary can answer the question directly, it needs to answer the question first, then make a supplement if have supplementary information. The summary content must come from the specific speaker who answers. Let's think step by step.
Output the generated content in JSON format as follows.
{"Answer-related speaker": List the speakers who give the answers,
"Speaker 1": Speaker 1 considers that ... (the summary of speaker 1's answer),
"Speaker 2": Speaker 2 considers that ... (the summary of speaker 2's answer)}

Table 9: Answer reformulation prompt.

You need to determine whether the dialogue can answer the given question. Let's think step by step.
Output the generated content as follows.
Thinking Process: The thinking process of determining the answerability of the question.
Answerability: If the question is answerable, output 'Yes', otherwise output 'No'.

Table 10: Answerability detection prompt.

Give the dialogue, a question and answers, you need to judge if the question leaks the answer.
Output the generated content as follows:
Thinking Process: The thinking process of the leakibility of the question.
Leak or not: Yes or No.

Table 11: Answer leakage detection prompt.

Given the dialogue and a question, you need to generate the answer.
The answer may involve multiple utterances or speakers.
The answer should not simply copy dialogue spans; it should summarize the relevant speakers' views.
You need to summarize the views of the speakers who answered the question, respectively.
You should generate content as follows:
Speaker x considers that ...

Table 12: Zero-shot prompt.

Metric	Model	Dataset	ALL	MM	MS	SM	SS
Attribution Accuracy	Claude-4.6	MEETINGQA	50.38	67.22	57.05	56.25	37.97
	Claude-4.6	MEETINGQA2	36.50	50.99	43.57	38.45	25.91
	Gemini-2.5	MEETINGQA	42.71	53.36	51.81	44.60	33.48
	Gemini-2.5	MEETINGQA2	35.60	43.28	42.10	36.74	29.01
	GPT-4o	MEETINGQA	51.43	63.34	63.38	56.25	40.02
	GPT-4o	MEETINGQA2	31.64	41.70	44.19	26.70	22.72
Normalized Factual Correctness	Claude-4.6	MEETINGQA	57.50	63.00	60.00	72.00	50.50
	Claude-4.6	MEETINGQA2	52.50	52.50	54.50	64.00	49.50
	Gemini-2.5	MEETINGQA	58.00	55.50	57.00	70.00	58.00
	Gemini-2.5	MEETINGQA2	47.50	42.00	48.50	53.50	49.00
	GPT-4o	MEETINGQA	59.50	64.50	67.00	70.00	53.00
	GPT-4o	MEETINGQA2	46.50	53.00	51.50	57.00	39.50

Table 6: Cross-benchmark comparison between MEETINGQA and MEETINGQA2 using the proposed diagnostic metrics. Factual correctness is linearly normalized from the original 0-2 judge scale to 0-100 for comparability with attribution accuracy. Higher is better for both metrics.

Dialogue:

Speaker 1: NITE object model, I think.
Speaker 2: NITE.
Speaker 3: Okay, that makes sense.
Speaker 0: What do you say?
Speaker 1: NITE object model, I think that's With the removing the disfluencies, have you done anything with that yet?
Speaker 0: Okay.
Speaker 3: Mm.
Speaker 1: Anybody.
Speaker 3: No.
Speaker 0: No, we haven't.
Speaker 1: Cause I was wondering how we're gonna write it back to XML, if we did the processing.
Speaker 3: I thought we were just going to have that as words.
Speaker 1: Writing it back to XML Seems There's a blop of text.

Question: Have Speaker 3 and Speaker 0 done anything with removing disfluencies using the NITE object model yet?

Gold: Speaker 3 considers that they have not done anything with removing disfluencies using the NITE object model yet. Speaker 0 confirms that they have not done anything with removing disfluencies using the NITE object model yet.

LLaMA3.1-8B (BS=0.51): Speaker 3 considers that they haven't done anything with removing disfluencies using the NITE object model yet, as they say "No" to the question "With the removing the disfluencies, have you done anything with that yet?".

GPT-3.5 Turbo (BS=0.28): Speaker 3 and Speaker 0 have not done anything with removing disfluencies using the NITE object model yet. They discussed the possibility of removing disfluencies and displaying the text without disfluencies in a text or XML format. They mentioned the option of using XSLT to transform the text into XML. However, they also considered the option of selectively printing the text without disfluencies on the fly. They acknowledged that displaying the text without disfluencies would limit the ability to search and listen to the audio. Overall, they concluded that they would focus on displaying the text without disfluencies as a summary rather than a fully searchable and listenable transcript.

GPT-4o (BS=0.61): Speaker 3 and Speaker 0 confirm that they have not done anything with removing disfluencies using the NITE object model.

Dialogue:

Speaker 4: Yeah. But I'm still confused cause I thought that's just what Jonathan said that we can't load a massive document of that size.
Speaker 3: It doesn't matter, cause all the calculation's done off-line.
Speaker 4: On the other hand it shouldn't be should be fifty mega-byte in RAM or something, it shouldn't be massive, should it? Actually fifty hundred megabyte is quite big in RAM.
Speaker 0: Hmm.
Speaker 4: Just thinking, what's the so We do get an error message with the project if we load everything into the project with all the data they load. So we know that doesn't work. So our hope is essentially that we load less into it.
Speaker 1: Yes.
Speaker 0: Yeah.
Speaker 4: What's this lazy loading thing, somebody explain lazy loading to me.

Question: Should loading fifty megabytes of audio data into RAM be considered too massive?

Gold: Speaker 4 considers that fifty to a hundred megabytes in RAM is quite large. He highlights that an error occurs when trying to load all data into the project, implying that this amount of data may be too massive for their purposes and they hope to load less into RAM to avoid issues.

LLaMA3.1-8B (BS=0.52): Speaker 4 considers that loading fifty megabytes of audio data into RAM might be too massive.

GPT-3.5 Turbo (BS=0.35): Speaker 4 considers that loading fifty megabytes of audio data into RAM is not too massive. They discuss the possibility of chunking in pre-annotated audio and using a baseline for the parts that are not pre-annotated. They also mention the importance of prioritizing packages and prototype development.

GPT-4o (BS=0.41): Speaker 4 considers that loading fifty megabytes of audio data into RAM might be quite big and potentially massive, given concerns about the total memory complexity. They suggest checking the capabilities of the audio player and how files can be streamed or buffered to handle the data efficiently.

Figure 4: QA examples from MeetingQA2. BS represents BERTScore.