MEBench: Benchmarking Large Language Models for Cross-Document Multi-Entity Question Answering

Anonymous ACL submission

Abstract

Cross-Document Multi-entity question answering (MEQA) demands the integration of scattered information across documents to resolve complex queries involving entities, relationships, and contextual dependencies. Although large language models (LLMs) and retrieval-augmented generation (RAG) systems show promise, their performance on cross-document MEQA remains underexplored due to the absence of tailored benchmarks. To address this gap, we introduce MEBench, a scalable multi-document, multi-entity benchmark designed to systematically evaluate LLMs' capacity to retrieve, consolidate, and reason over scattered and dense information. Our benchmark comprises 4,780 questions which are systematically categorized into three primary categories: *Comparative Reasoning, Statistical Reasoning* and *Relational Reasoning*, further divided into eight distinct types, ensuring broad coverage of real-world multi-entity reasoning scenarios. Our experiments on state-of-the-art LLMs reveal critical limitations: even advanced models achieve only 59% accuracy on MEBench. Our benchmark emphasizes the importance of completeness and factual precision of information extraction in MEQA tasks, using Entity-Attributed F1 (EA-F1) metric for granular evaluation of entity-level correctness and attribution validity. MEBench not only highlights systemic weaknesses in current LLM frameworks but also provides a foundation for advancing robust, entity-aware QA architectures. The source code and data have been made available at https://github.com/tl2309/SRAG.

1 Introduction

The emergence of large language models (LLMs) has significantly advanced natural language processing capabilities, demonstrating exceptional performance in diverse tasks spanning text generation to complex logical reasoning Achiam et al. (2023) Meta Llama3 (2024). Nevertheless, long-context LLMs exhibit notable limitations in processing entity-dense analytical reasoning, particularly when contextual dependencies are distributed across multiple documents, and we analytically argue that context window limitations, over-reliance on parametric knowledge, and poor cross-document attention as the key bottlenecks. On the other hand, current implementations of retrieval-augmented generation (RAG) architectures Wu et al. (2025) Fan et al. (2024) Tang et al. (2024) frameworks' effectiveness in addressing cross-document multi-entity question answering (MEQA) remains insufficiently investigated. Furthermore, the field lacks comprehensive benchmarking frameworks specifically designed to evaluate the performance of LLMs and RAG systems for cross-document entity-intensive tasks. As shown in Figure 1, existing evaluation metrics frequently inadequately represent the complexities inherent in real-world MEQA applications Song et al. (2024), where queries such as "What is the number distribution of all Turing Award winners by fields of study by 2023?" necessitate not only high-precision information retrieval but also reasoning over fragmented, entity-specific information across heterogeneous document sources.

To address this methodological gap, we present MEbench, a novel benchmarking framework specifically designed to assess the performance of large language models and RAG systems in cross-document multi-entity question answering scenarios. The benchmark simulates real-world information integration challenges where correct answers require synthesizing entity-centric evidence distributed across multiple documents, with a single instance of document omission or entity misinterpretation can propagate errors through the reasoning chain. As shown in Table 2, MEBench features a mean entity density of 409 entities per query, with systematically varied entity cardinality across three operational tiers: low (0-10 enti-

ties), medium (10-100 entities), and high complexity (>100 entities). This stratified design enables granular performance evaluation across different entity scales and task difficulty levels. The framework comprises 4,780 validated question-answer pairs systematically categorized into three primary categories and eight distinct types, MEBench spans diverse real-world scenarios, from academic field distributions to geopolitical event analysis. Our experiments with state-of-the-art models, including GPT-4 and Llama-3, reveal significant shortcomings: even the most advanced LLMs achieve only 59% accuracy on MEBench. This underscores systemic weaknesses in current frameworks, for example, models frequently fail to locate all entity and their attributes or infer implicit relationships, highlighting the need for architectures that prioritize entity-aware retrieval and contextual consolidation.

Our main contributions are summarized as follows:

- **Development of MEBench.** A scalable benchmark designed to evaluate LLMs and RAG systems in cross-document aggregation and reasoning. It includes 4,780 validated question-answer pairs spanning three categories and eight types, simulating real-world scenarios that demand integration of scattered, entity-specific information.
- **Entity-centric Task Categories and Evaluation.** Utilization of Entity-Attributed F1 (EA-F1), a granular metric for assessing entity-level correctness and attribution validity, alongside a stratified entity density design (low: 0–10, medium: 11–100, high: >100 entities per query). Our framework emphasizes completeness and factual precision in information extraction, addressing gaps in existing metrics for entity-dense MEQA tasks.
- **Scalable Benchmark Construction.** A scalable, automated pipeline: Knowledge graph extraction from structured Wikipedia for cross-document relationship discovery; Relational table generation to preserve entity-property relationships; Template-based QA generation ensuring reproducibility and reducing cost and labor.

2 Related Work

Recent advancements in question answering (QA) have been driven by breakthroughs in LLMs and RAG systems. While these technologies excel in single or a few document settings, demonstrating proficiency in tasks like fact extraction, summarization, and reasoning within a single source, their performance in cross-document, multi-entity scenarios remains underexplored. This section contextualizes our work within three key research areas: single-document QA, cross-document aggregation, and entity-centric evaluation.

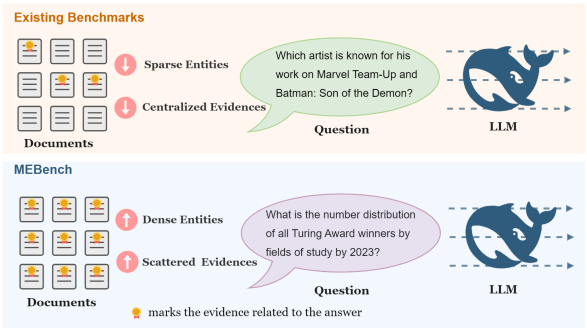

Figure 1: Existing benchmarks vs. MEBench. Unlike existing benchmarks which feature centralized evidence distributions and sparse entity mentions, MEBench presents entity-dense scene where critical evidences are dispersed across multiple documents, necessitating that when seeking an answer, no document or entity can be ignored.

2.1 Single-Document QA and LLM Progress.

Many QA benchmarks, such as SQuAD Rajpurkar et al. (2016), Natural Questions Kwiatkowski et al. (2019), L-eval An et al. (2024) and needle-in-a-haystack Kamradt (2023), focus on extracting answers from individual document. Modern LLMs like GPT-4 Achiam et al. (2023), Llama-3 Meta Llama3 (2024), and PaLM Chowdhery et al. (2023) have achieved near-human performance on these tasks, leveraging their ability to parse and reason within localized contexts. However, these benchmarks do not address the complexities of integrating information across multiple documents, a critical limitation for real-world applications.

2.2 Cross-Document Aggregation Challenges.

Efforts to extend QA to multi-document settings include datasets like HotpotQA Yang et al. (2018), MuSiQue Trivedi et al. (2021), LooGLE Li et al. (2024), LM-Infinit Han et al. (2024), ∞ Bench Zhang et al. (2024), CLongEval Qiu et al. (2024), BAMBOO Dong et al. (2024) and Loong Wang et al. (2024), which emphasize multi-hop reasoning and cross-source synthesis. While

these benchmarks highlight the need for systems to connect disparate information, they often prioritize breadth over depth in entity-centric reasoning. For instance, questions in these datasets rarely demand the consolidation of attributes for dozens of or more entities (*e.g.,* aggregating ACM Fellows' expertise across fields), a gap that limits their utility in evaluating entity-dense scenarios. Recent RAG frameworks Fan et al. (2024) aim to enhance retrieval-augmented QA but struggle with ensuring completeness and attribution validity when handling multi-entity queries.

2.3 Entity-Centric Evaluation Metrics.

Existing evaluation metrics for QA, such as F1 score and exact match (EM), focus on answer surface-form correctness but overlook granular entity-level attribution Rostampour et al. (2010). Metrics in FEVER Thorne et al. (2018) Attributed QA Bohnet et al. (2023) and emphasize source verification, yet they lack the specificity to assess multi-entity integration. For example, they do not systematically measure whether all relevant entities are retrieved, their attributes are correctly extracted, or their sources are accurately used, which is a shortcoming that becomes critical in MEQA tasks.

2.4 The Gap in Multi-Entity QA Benchmarks.

Prior work has yet to establish a benchmark that systematically evaluates LLMs and RAG systems on entity-dense, cross-document reasoning. Current datasets either lack the scale and diversity of real-world multi-entity questions or fail to provide fine-grained metrics for assessing entity-level completeness and attribution Song et al. (2024) Wang et al. (2024) Bai et al. (2025). MEBench addresses these limitations by introducing a comprehensive evaluation framework that challenges models to retrieve, consolidate, and reason over scattered entity-centric data across heterogeneous sources. By incorporating the Entity-Attributed F1 (EA-F1) metric, our benchmark advances the field toward more precise, entity-aware QA systems.

3 MEBench

3.1 Task overview

MEBench is a structured evaluation framework designed to systematically assess the capabilities of LLMs and RAG systems in performing cross-document multi-entity question answering. This framework targets three core reasoning modalities: comparative analysis, statistical inference, and relational reasoning, and each decomposed into specialized subtasks that rigorously test distinct facets of LLM performance. (Examples of tasks are provided in Table 1), ensuring broad coverage of real-world multi-entity reasoning scenarios. Each of three primary task categories addresses distinct reasoning challenges:

Comparative Reasoning Comparative reasoning tasks evaluate LLM's ability to juxtapose entities across heterogeneous documents, demanding both attribute alignment and contextual synthesis.

Statistical Reasoning Statistical tasks Zhu et al. (2024) assess LLM's proficiency in **quantitative synthesis**, including aggregation, distributional analysis, correlation analysis, and variance analysis across multi-document.

Relational Reasoning Relational tasks probe model's capacity to model explicit interactions and counterfactual dependencies among entities.

3.2 Benchmark Construction

MEBench was constructed through a systematic pipeline:

3.2.1 Data Collection

Concept Topic Identification. In the initial phase of data collection for MEbench, a meticulous process is employed to determine the concept topics that are applicable to multi-entity scenarios. These topics are carefully selected based on their significance, prevalence, and the potential for generating complex multi-entity questions, and examples can be seen in Appedix Table 5.

Entity and Property Identification. Once the concept topics are determined, We input descriptions related to the concept topic into the LLM (we use GPT-4), which then processes the text to identify concept entity and property, as illustrated in Figure 2-a1. After the LLM identifies the entity and Property via iterative semantic refinement, we map them to entity IDs and Property IDs in the Wiki graph. This mapping is crucial as it allows for seamless integration with the vast amount of structured data available in Wikipedia. The detailed method is in Appendix A.1. Using the Entity ID and property ID, we synthesise SPARQL. We then utilize the API provided by Wikipedia to retrieve the wiki web pages of all entities related to the

Table 1: Examples of multi-entities queries.

Categories	Types	Examples
Comparison	Intercomparison	Which has more ACM fellow, UK or USA?
	Superlative	Which city has the highest population?
Statistics	Aggregation	How many ACM fellow are from MIT?
	Distribution Compliance	Does the nationality of ACM fellows follow a normal distribution?
	Correlation Analysis	Is there a linear relationship between number of events and records broken in Olympic Games?
	Variance Analysis	Do the variances in the number of participating countries and total events in the Summer Olympics differ significantly?
Relationship	Descriptive Relationship	Is there a relationship between the year of ACM fellowship induction and the fellows' areas of expertise?
	Hypothetical Scenarios	If China wins one more gold medal, will it overtake the US in the gold medal tally at the 2024 Olympics?

Table 2: Statistics of MEBench benchmark.

Categories	MEBench-train	MEBench-test	MEBench-total
#-Queries	3406	1374	4780
#-Topics	165	76	241
Ave. #-entities /Q	460	391	409
Hops			
#-one-hop Q	1406	606	2012
#-multi-hop Q	1322	768	2090
Categories			
#-Comparison	1107	438	1545
#-Statistics	1440	585	2025
#-Relationship	859	351	1210
Entity Density			
#-low (0–10)	487	196	683
#-medium(11–100)	973	393	1366
#-high (>100)	1946	785	2731

topic. For example, if our concept topic is "ACM Fellows", we would obtain the Wikipedia pages of all ACM Fellows, which contain their detailed information. We use GPT-4 to generate a set of interesting entity attributes. These attributes are carefully chosen based on the general interest and relevance in the domain. For ACM Fellows, as example, nationality, research field, institution, and academic contribution maybe the attributes that people commonly pay attention to.

Structured Information Processing. Once the document set is collected, we proceed to the structured information processing stage. The documents we have gathered from Wikipedia have well-defined and accurate structural relations. Due to the structured nature of the documents, we do not need to rely on the long context ability of large lan-

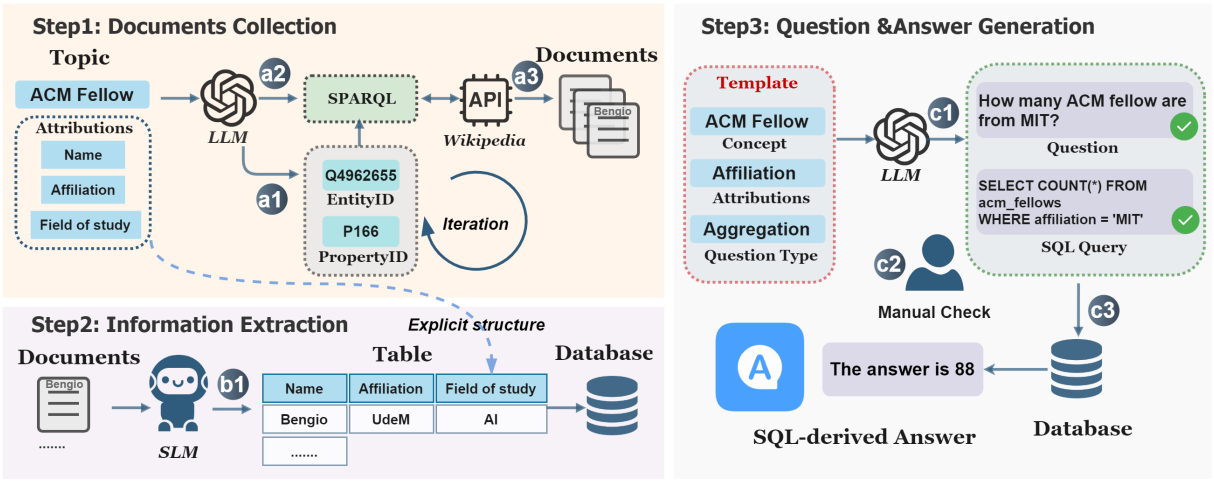

Figure 2: The systematic pipeline of Benchmark Construction. It comprising three phases: documents collection, information extraction and question-answer generation. In the documents collection phase, concept topics relevant to multi-entity scenarios are selected, followed by GPT-4 processing descriptions to extract entities and properties mapped to Wikipedia IDs for integration with structured Wiki data. Structured information from Wikipedia documents is processed using small language models (SLMs) due to the structured nature of the documents, culminating in table creation with entity attributes as columns. For QA generation, questions are generated following a "template-driven, entity-attribute coupling" paradigm using GPT-4 with predefined templates, and undergo syntactic, semantic, and ambiguity checks, while answers are programmatically derived via SQL queries against the table and standardized into canonical forms. The final dataset ensures traceability (SQL-linked answers), scalability (template-driven approach), and rigor (execution-based answering reduces hallucination risks).

guage models. Instead, we can use small language models (SLMs) for information extraction. They are well-suited for tasks where the information is already structured and the focus is on extracting specific details Fan et al. (2025).

Table Generation. The final step in the data collection process is to generate a table, as shown in Figure 2-b1. We use the the entity attributes as the column headers of the table. Each row in the table represents an individual entity. For example, in the case of ACM Fellows, each row would correspond to an individual ACM Fellow.

3.2.2 Question and answer Generation

The question and answer generation framework for MEBench is a structured, multi-phase process that leverages LLM and tabular data to produce both semantically coherent questions and computationally verifiable answers.

Question Generation. The foundational input for the QA generation pipeline is the table generated in last step. The generation of questions follows a "template-driven, entity-attribute coupling" paradigm, implemented through LLM (GPT-4), as illustrated in Figure 2-c1. Predefined syntactic and semantic templates govern the grammatical structure and intent of questions. These templates are shown in Appendix Table 6. The LLM instantiates templates with entity-attribute pairs, ensuring syntactic diversity while adhering to logical constraints. Generated questions undergo validation via: Syntactic Checks, ensuring grammatical correctness; Semantic Grounding, verifying that the question is answerable using the table's data; Ambiguity Reduction, pruning underspecified questions (*e.g.,* "Describe the economy" revised to "Describe the GDP growth rate of Brazil in 2023").

Answer Generation. Answers are derived programmatically through automated SQL query execution, ensuring reproducibility and alignment with the table's ground-truth data. The synthesized SQL is executed against the table, yielding direct answers or sub-tables (Intermediate results requiring post-processing), as illustrated in Figure 2-c3. Answers are standardized to ensure consistency: Numeric results are rounded to significant figures; Categorical answers are converted to canonical forms (*e.g.,* "USA" to "United States").

3.3 Data Statistics

The benchmark comprises 4,780 methodically structured questions partitioned into two subsets: a training set (3,406 questions) for model fine-tuning

or train, and a test set (1,374 questions) for rigorous evaluation. Based on entity count, the data is divided into three groups: “low” (0-10), “Medium” (11-100), and “high” (>100), containing 683, 1366, and 2731 entries, respectively. Table 2 details comprehensive statistics of the benchmark. We also analyze the proportion of questions rejected during manual review and about 21% of the questions are failure to meet quality standards.

4 Experiment

4.1 Experiment Setup

4.1.1 Models

For open-source LLMs, we conduct experiments using the representative Meta-Llama-3-8B-Instruct Meta Llama3 (2024) and apply QLoRA Dettmers et al. (2023) to fine-tune it with the training set of MEBench. For proprietary LLMs, we select the widely recognized GPT models, including GPT-3.5-turbo Ouyang et al. (2022) and GPT-4 Achiam et al. (2023).

4.1.2 RAG

We implement a hierarchical retrieval framework that explicitly incorporates document organizational structures into the RAG pipeline to explore whether RAG can enhance the model’s performance on MEBench. For the Embedding choice, we employ the OpenAI Embedding model OpenAI, and the chunk size is 1024. For each document, we retrieve the top-5 most related chunks and concatenate them in their original order to form the context input for the model.

4.1.3 Evaluation Metrics

We adopt Accuracy (Acc) as the primary metric to assess the performance of LLMs on MEBench tasks. For the subcategories of Variance Analysis, Correlation Analysis, and Distribution Compliance within the Statistics tasks, which are shown in Table 1, we focus solely on prompting LLMs to identify relevant columns and applicable methods, evaluating the accuracy of their selections instead of the computational results, as LLMs’ abilities in precise calculations are not the central focus of this study. In addition, we evaluate performance of information extraction using Entity-Attributed F1 (EA-F1). This is an F1 score applied to the predicted vs. gold sets of the (entity, atrribution, value) . All three elements in the tuple must exactly match the tuple in the ground truth to be marked correct.

4.2 Results and Analysis

Various models exhibit notable variations in performance on MEBench. Table 3 presents experimental results alongside overall accuracy on MEBench, and Figure 3 shows accuracy on eight further-divided tasks.

4.2.1 Main result

GPT-4 + RAG achieved superior accuracy (59.3%), outperforming the second-ranked model (FT Llama-3-Instruct: 55.6%) by a statistically significant margin. Notably, GPT-4 + RAG excelled in relational (68.7%) and comparative (76.3%) queries, likely due to its superior contextual understanding. However, all models exhibited markedly lower accuracy in statistical queries (GPT-4 + RAG: 41.0%), suggesting inherent challenges in numerical reasoning. In our evaluation, we focused on analyzing the capability of LLMs to extract question-related data. This assessment aimed to understand how well these sophisticated models can organize and present data for the question. The result is shown in Table 4. These results underscore the critical role of information extraction architectures in mitigating hallucinations and grounding outputs in factual data. Introducing RAG significantly improves overall performance, particularly in comparison tasks, while fine-tuning LLaMA-3-Instruct alone does not yield substantial gains without RAG. On MEBench, open-source models like LLaMA-3-Instruct, even with RAG, can’t match proprietary models like GPT-4, which achieves a 59.3% accuracy compared to LLaMA-3-Instruct’s 32.5%.

4.2.2 Fine-grained Performance on Sub-tasks.

Figure 3 shows that vanilla LLMs perform well in correlation analysis and descriptive relationship tasks, while RAG significantly improves intercomparison and superlative tasks. However, neither fine-tuning nor RAG overcomes challenges in variance analysis and aggregation tasks, while GPT-4 + RAG achieves accuracy of 15.3% and 32.1%.

4.2.3 Entity density Analysis.

As we can see from Table 3, our experiments underscore the impact of entity density on model performance in MEQA tasks. This phenomenon arises because higher entity densities amplify two critical challenges inherent to MEQA systems: (1) Semantic ambiguity due to overlapping relational predicates among entities (e.g., distinguishing "Paris [person]" vs. "Paris [location]" within narrow con-

Table 3: Experimental results for MEBench.

Models	Accuracy			
	Comparison	Statistics	Relationship	Overall
All sets				
GPT-3.5-turbo	0.105	0.198	0.476	0.239
GPT-3.5-turbo + RAG	0.605	0.260	0.476	0.425
GPT-4	0.199	0.289	0.507	0.316
GPT-4 + RAG	0.763	0.410	0.687	0.593
Llama-3-Instruct	0.046	0.118	0.256	0.130
Llama-3-Instruct + RAG	0.447	0.181	0.410	0.325
FT Llama-3-Instruct	0.046	0.253	0.259	0.189
FT Llama-3-Instruct + RAG	0.687	0.448	0.573	0.556
Set1 (0-10)				
GPT-3.5-turbo	0.435	0.583	0.560	0.530
GPT-3.5-turbo + RAG	0.548	0.654	0.620	0.612
GPT-4	0.451	0.595	0.540	0.535
GPT-4 + RAG	0.870	0.619	0.740	0.729
Llama-3-Instruct	0.322	0.500	0.400	0.418
Llama-3-Instruct + RAG	0.419	0.571	0.480	0.500
FT Llama-3-Instruct	0.322	0.511	0.380	0.418
FT Llama-3-Instruct + RAG	0.580	0.677	0.690	0.676
Set2 (11-100)				
GPT-3.5-turbo	0.364	0.495	0.544	0.466
GPT-3.5-turbo + RAG	0.613	0.581	0.640	0.607
GPT-4	0.348	0.476	0.521	0.447
GPT-4 + RAG	0.791	0.511	0.661	0.638
Llama-3-Instruct	0.240	0.385	0.357	0.332
Llama-3-Instruct + RAG	0.428	0.454	0.459	0.447
FT Llama-3-Instruct	0.240	0.434	0.344	0.349
FT Llama-3-Instruct + RAG	0.612	0.608	0.655	0.640
Set3 (>100)				
GPT-3.5-turbo	0.09	0.158	0.291	0.173
GPT-3.5-turbo + RAG	0.389	0.191	0.311	0.285
GPT-4	0.142	0.202	0.309	0.210
GPT-4 + RAG	0.436	0.270	0.405	0.357
Llama-3-Instruct	0.055	0.108	0.168	0.106
Llama-3-Instruct + RAG	0.265	0.147	0.253	0.212
FT Llama-3-Instruct	0.055	0.177	0.167	0.136
FT Llama-3-Instruct + RAG	0.401	0.291	0.355	0.345

texts), and (2) computational overhead in attention-based architectures attempting parallel reasoning over entangled entity-attribution pairs (e.g. transformer self-attention weights saturate under dense cross-entity dependencies).

- Low Entity Density: Models generally performed well in low-density scenarios. The simplicity of context allowed for accurate entity recognition and minimal ambiguity.
- Medium Entity Density: Performance began to decrise among models in medium-density scenarios by 6% average acc. This variance suggests differences in how models handle increased entity complexity and overlapping contexts.
- High Entity Density: High-density questions posed significant challenges, with an average

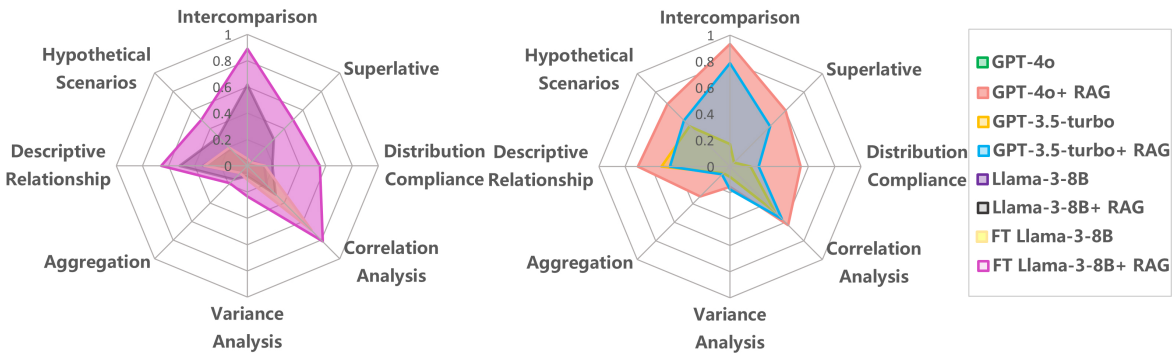

Figure 3: The Experimental results for eight subtasks of each model.

Table 4: Quality of Large Language Models (LLMs) in EA-F1.

Models	$EA-F1$
GPT-3.5-turbo	0.25
GPT-3.5-turbo + RAG	0.43
GPT-4	0.36
GPT-4 + RAG	0.71
Llama-3-Instruct	0.21
Llama-3-Instruct + RAG	0.39
FT Llama-3-Instruct	0.21
FT Llama-3-Instruct + RAG	0.59

acc drop to 22.8% across models. The result highlighting limitations in current architectures' ability to handle complex multi-entity questions.

5 Limitations

While MEBench provides a comprehensive framework for evaluating cross-document multi-entity reasoning, our work has several limitations that warrant further investigation. Although MEBench covers eight distinct reasoning types across three broad categories, real-world MEQA scenarios may involve even more intricate combinations of logical, temporal, or causal dependencies. The current benchmark does not explicitly model dynamic or time-sensitive entity interactions, which could limit its applicability to domains like financial forecasting or event-driven narratives. The benchmark relies on a curated collection of documents to ensure controlled evaluation. While this design choice minimizes noise, it may not fully replicate the challenges of real-world environments where documents vary widely in quality, redundancy, and structure. Future iterations could incorporate noisy or incomplete data sources to better simulate practical scenarios. While the Entity-Attributed F1 (EA-F1) metric rigorously assesses entity-level correctness and attribution validity, it prioritizes factual precision over semantic coherence. This may undervalue partially correct answers that demonstrate valid reasoning chains but contain minor factual inaccuracies. A hybrid evaluation framework combining EA-F1 with human judgment could provide a more holistic assessment.

6 Conclusion

In this study, we have comprehensively addressed the significant challenges that multi-entity question answering (MEQA) poses to LLMs and RAG systems. The limitations of existing methods in handling cross-document aggregation, especially when dealing with entity-dense questions, have been clearly identified and analyzed. We introduced MEBench, a groundbreaking multi-document, multi-entity benchmark. Our experiments on state-of-the-art LLMs such as GPT-4 and Llama-3, along with RAG pipelines, have shed light on the critical limitations of these advanced models. The fact that even these leading models achieve only 59% accuracy on MEBench underscores the magnitude of the challenges in MEQA. MEBench has effectively highlighted the systemic weaknesses in current LLM frameworks. These weaknesses serve as valuable insights for future research directions. For instance, the need for improved algorithms to retrieve and consolidate fragmented information from heterogeneous sources is evident. Additionally, there is a pressing need to develop more robust entity-aware QA architectures that can better handle the complexities of MEQA.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. 2024. L-eval: Instituting standardized evaluation for long context language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14388–14411, Bangkok, Thailand. Association for Computational Linguistics.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2025. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks.

Bernd Bohnet, Vinh Q. Tran, Pat Verga, Roee Aharoni, Daniel Andor, Livio Baldini Soares, Massimiliano Ciaramita, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, Kai Hui, Tom Kwiatkowski, Ji Ma, Jianmo Ni, Lierni Sestorain Saralegui, Tal Schuster, William W. Cohen, Michael Collins, Dipanjan Das, Donald Metzler, Slav Petrov, and Kellie Webster. 2023. Attributed question answering: Evaluation and modeling for attributed large language models.

Aakanksha Chowdhery, Sharan Narang, and Jacob Devlin. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*.

Zican Dong, Tianyi Tang, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2024. BAMBOO: A comprehensive benchmark for evaluating long text modeling capacities of large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2086–2099, Torino, Italia. ELRA and ICCL.

Tianyu Fan, Jingyuan Wang, Xubin Ren, and Chao Huang. 2025. Minirag: Towards extremely simple retrieval-augmented generation.

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.

Chi Han, Qifan Wang, Hao Peng, Wenhan Xiong, Yu Chen, Heng Ji, and Sinong Wang. 2024. LM-infinite: Zero-shot extreme length generalization for large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3991–4008, Mexico City, Mexico. Association for Computational Linguistics.

Greg Kamradt. 2023. Needle in a haystack- pressure testing llms. https://github.com/gkamradt/LLMTest_NeedleInAHaystack.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7(15):453–466.

Jiaqi Li, Mengmeng Wang, Zilong Zheng, and Muhan Zhang. 2024. Loogle: Can long-context language models understand long contexts?

Chunwei Liu, Matthew Russo, Michael Cafarella, Lei Cao, Peter Baille Chen, Zui Chen, Michael Franklin, Tim Kraska, Samuel Madden, and Gerardo Vitagliano. 2024. A declarative system for optimizing ai workloads. *arXiv preprint arXiv:2405.14696*.

Meta Llama3. 2024. Meta llama3. https://llama.meta.com/llama3/. Accessed: 2024-04-10.

OpenAI. Openai embedding model. https://huggingface.co/Xenova/text-embedding-ada-002. Accessed [Date of access].

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Pratyush Patel, Esha Choukse, Chaojie Zhang, Íñigo Goiri, Aashaka Shah, Saeed Maleki, and Ricardo Bianchini. 2023. Splitwise: Efficient generative llm inference using phase splitting. *arXiv preprint arXiv:2311.18677*.

Zexuan Qiu, Jingjing Li, Shijue Huang, Xiaoqi Jiao, Wanjun Zhong, and Irwin King. 2024. Clongeval: A chinese benchmark for evaluating long-context large language models.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

A. Rostampour, A. Kazemi, F. Shams, A. Zamiri, and P. Jamshidi. 2010. A metric for measuring the degree of entity-centric service cohesion. In *2010 IEEE International Conference on Service-Oriented Computing and Applications (SOCA)*, pages 1–5.

Mingyang Song, Mao Zheng, and Xuan Luo. 2024. Counting-stars: A multi-evidence, position-aware, and scalable benchmark for evaluating long-context large language models.

Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. 2024. Graphgpt: Graph instruction tuning for large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 491–500, New York, NY, USA. Association for Computing Machinery.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2021. Musique: Multi-hop questions via single-hop question composition.

Minzheng Wang, Longze Chen, Cheng Fu, Shengyi Liao, Xinghua Zhang, Bingli Wu, Haiyang Yu, Nan Xu, Lei Zhang, and Run Luo. 2024. Leave no document behind: Benchmarking long-context llms with extended multi-doc qa.

Kevin Wu, Eric Wu, and James Zou. 2025. Clasheval: Quantifying the tug-of-war between an llm's internal prior and external evidence.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Khai Hao, Xu Han, Zhen Leng Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. 2024. ∞bench: Extending long context evaluation beyond 100k tokens.

Yizhang Zhu, Shiyin Du, Boyan Li, Yuyu Luo, and Nan Tang. 2024. Are large language models good statisticians?

A Appendix

A.1 Methodology for composite SPARQL Generation via Iterative Semantic Refinement

A.1.1 Initial Query Parsing Using GPT-4

We employ a transformer-based large language model (LLM), specifically GPT-4, to perform preliminary natural language question decomposition. This stage generates a proto-SPARQL query containing candidate triple patterns with hypothesized entity-property relationships. While this initial output captures broad syntactic structures (e.g., basic graph pattern groupings), it frequently exhibits two critical inaccuracies:

Entity Misalignment: Incorrect Wikidata Q-ID assignments due to lexical ambiguity (e.g., "Java" as programming language vs. Indonesian island)

Property Mismatch: Invalid P-ID selections arising from underspecified predicate semantics (e.g., using P19 [place of birth] instead of P20 [place of death])

A.1.2 Semantic Validation Layer

To address these limitations, we implement a multi-stage correction framework:

(a) Structured Knowledge Anchoring

The system interfaces with the Wikipedia API through programmatic endpoints that map surface forms to canonical entities via:

```
def getwikidataid(term):
    response = requests.get(
    f"https:en.wikipedia.orgwapi.php?
                                    ac-
tion=query&format=json&prop=pageprops&titles=term"
    )
                                return re-
sponse.json()["query"]["pages"].get("pageprops",
    ).get("wikibaseitem")
```

(b) Neural-Semantic Disambiguation Module

GPT-4 serves as our semantic analysis engine, performing three key operations:

a. Contextual disambiguation using entity linking algorithms enhanced by Wikifier-style mention detection

b. Property type validation against Wikidata's ontology constraints (rdf:type, owl:ObjectProperty)

c. Temporal scope verification for time-sensitive queries requiring qualifiers like P585 [point in time]

A.1.3 Iterative Refinement Protocol

The system implements closed-loop feedback through successive cycles of:

a. Executing candidate SPARQL on the Wikidata Query Service endpoint;

b. Analyzing result cardinality and type consistency;

c. Applying constraint satisfaction heuristics:

```
FILTER (?population > 1e6 && ?country
IN wd:Q30) # Example numerical/entity con-
straints
```

Each iteration tightens precision metrics until meeting termination criteria defined by either:
$\frac{|ValidResults_t|}{|TotalResults_t|} \geq \theta_{precision}$ ($\theta = 0.98$ empirically)
or maximum iteration thresholds.

A.1.4 Final Query Synthesis

Through combining LLM-based semantic parsing with knowledge-grounded verification, we converge on an optimized SPARQL template satisfying both syntactic validity and functional correctness requirements for structured knowledge extraction.

A.2 Optimization

Two aspects of optimization are included in MEBench system to enhance the overall performance:

Model Selection. Model selection is straightforward yet highly effective for optimization Liu et al. (2024). Our system comprises multiple tasks, necessitating the selection of the most suitable model for different tasks. For basic tasks, more affordable and faster LLMs can suffice, while utilization of the most advanced LLMs is essential in more complex tasks to ensure optimal performance. Specifically, our system employs powerful yet resource-intensive GPT-4 for tasks such as semantic analysis or generation of table schemas and SQL queries. In contrast, for more basic information extraction, we utilize open-source Mistral-7B, thereby achieving a balance between cost efficiency and functional performance.

LLM Input/Output Control SplitWise Patel et al. (2023) shows that LLM inference time is generally proportional to the size of input and output tokens. Since GPT models decide the cost based on the input token, we try to minimize the input of large models. Meanwhile, we use the instructive prompt to reduce the size of the outputs generated

Table 5: Example Topics and Their Entities Attributions.

Topics	Entities Attributions	#-Entities
ACM fellow	nationality, field of study, affiliation	1115
Presidents of the US	term lengths, political parties, vice-presidents, birth states, previous occupations	55
Chemical Elements	atomic number, atomic mass, boiling point, melting point, electron configuration	166
Summer Olympic Games	host cities, number of participating countries, total number of events, medal tally, records broken	35
Nobel Prize in Chemistry	categories, year of award, country of origin, field of contribution.	194
Cities of the World	population, geographic coordinates, altitude, GDP	7040

Table 6: Template example for questions generated by the LLM (GPT-4).

Types	Sub-types	Template Examples
Comparison	Intercomparison	Which has high [property], [entity A] or [entity B]?
	Superlative	Which [entity] has the highest/lowest [property]?
Statistics	Aggregation	How many [entities] have [specific property value]?
	Distribution Compliance	Does [property] follow a normal distribution?
	Correlation Analysis	Is there a linear relationship between [property A] and [property B]?
	Variance Analysis	Are the variances in [property A] and [property B] significantly different?
Relationship	Descriptive Relationship	How is [entity A] related to [entity B]?
	Hypothetical Scenarios	What would be the impact if [entity A] collaborates with [entity B]?

by LLM without changing the quality of these outputs. The example of prompt is in Appendix A.2.1.

A.2.1 Prompt for Output Control

> Review your output to ensure it meets all the above criteria. Your goal is to produce a clear, accurate, and well-structured output. Just output the result, no other word or symbol.

A.2.2 Quality Control

We devise several strategies to ensure the integrity and effectiveness of questions.

Question Templates. The use of templates ensures that every question is crafted with a clear structure, making it easier for respondents to understand and answer them accurately. For relationship and complex statistic questions we turn the questions in a closed-ended style, as they require a specific response of either "yes" or "no", which make the answer in a standardized format. The examples of Question Templates is in the Appendix 6.

Question Refinement. After initial development, each question undergoes a refinement process which we used GPT-3.5-Turbo. This stage is critical for enhancing the clarity, relevance, and neutrality of the questions. It involves reviewing the questions for bias. This strategy helps in reduc-

ing misunderstandings and improving the overall quality of the questions.

Manual review. We assess the questions for accuracy, ensuring they are factually correct and relevant to our purpose. Manual reviews can also provide insights into whether the questions are likely to effectively elicit the intended information from answers, thereby contributing to the reliability and validity of the benchmark.

A.3 Tables

Table 5 shows examples of topics and their entities' attributions. Table 6 shows examples of question templates to synthesize questions.