Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25)
Demonstrations Track

GE-Chat: A Graph Enhanced RAG Framework for Evidential Response Generation of LLMs

Longchao Da[1] , **Parth Mitesh Shah**[1] , **Kuan-Ru Liou**[1] , **Jiaxing Zhang**[2] , **Hua Wei**[1*]
[1]Arizona State University
[2]New Jersey Institute of Technology
{longchao, pshah113, kliou, hua.wei}@asu.edu, jz48@njit.edu

Abstract

Large Language Models (LLMs) have become integral to human decision-making processes. However, their outputs are not always reliable, often requiring users to assess the accuracy of the information provided manually. This issue is exacerbated by hallucinated responses, which are frequently presented with convincing but incorrect explanations, leading to trust concerns among users. To address this challenge, we propose GE-Chat, a knowledge Graph-enhanced retrieval-augmented generation framework designed to deliver Evidence-based responses. Specifically, when users upload a document, GE-Chat constructs a knowledge graph to support a retrieval-augmented agent, enriching the agent's responses with external knowledge beyond its training data. We further incorporate Chain-of-Thought (CoT) reasoning, n-hop subgraph searching, and entailment-based sentence generation to ensure accurate evidence retrieval. Experimental results demonstrate that our approach improves the ability of existing models to identify precise evidence in free-form contexts, offering a reliable mechanism for verifying LLM-generated conclusions and enhancing trustworthiness.

1 Introduction

Large Language Models (LLMs) have shown exceptional performance in tasks such as multi-round conversational interactions, question understanding, response generation, and reasoning based on provided contexts [Hu *et al.*, 2024], [Yuan *et al.*, 2024], [Zhang *et al.*, 2024], etc. These advancements have enabled diverse applications across domains, including customer support [Følstad and Skjuve, 2019], virtual assistance [Wei *et al.*, 2024], and augmented agents with tool-usage capabilities [Da *et al.*, 2024].

Despite their extensive training on large-scale expert corpora, LLMs are prone to generating incorrect or misleading information [Ji *et al.*, 2023]. Prominent LLM providers like OpenAI ChatGPT and Claude caution users with disclaimers such as "*LLMs can make mistakes. Check important information carefully*", emphasizing the ongoing challenge of ensuring response reliability [Liu *et al.*, 2025].

Existing solutions to reduce the generation of misinformation primarily focus on grounding LLM outputs in factual data through either fine-tuning [Wang *et al.*, 2024] or retrieval-augmented methods [Shuster *et al.*, 2021]. While fine-tuning reduces hallucinations, it demands significant computational resources and is impractical for proprietary black-box models. Retrieval-augmented approaches rely on external sources for factuality verification [Ding *et al.*, 2024], requiring multiple queries to external APIs or knowledge bases, which can increase latency and operational complexity.

To address these limitations, recent efforts have explored evidence-based response generation by matching model outputs to relevant source documents. For example, when users upload a document and pose a query, systems highlight raw contextual segments most relevant to the response, thus guiding the user to understand where the conclusion is drawn from the original document. However, the current method, such as [Lin, 2024] and [Han *et al.*, 2024] use the direct LLM-responsed source of the evidence [Saad-Falcon *et al.*, 2023]. It has two shortcomings, First, when faced with a redundant response, it can only perform on the chunk-level resource highlight, which gives a whole paragraph of relevant context without fine concentration. Second, the ability of LLMs to reflect their source evidence varies significantly on their instruction-following capability. Smaller models that lack instruction fine-tuning often struggle to highlight relevant information for users to refer to effectively.

To resolve the above issues, this paper proposes a framework named GE-Chat that provides users with evidence of an LLM's response in a more accurate and generalizable way. Different from existing work, this method not only poses constraints on the derived source that it must come from the original context, but also provides sentence-level fine-grained identification to accurately mark out the evidence supporting the LLMs' conclusion. This framework can also be applied to any LLM with outstanding evidence retrieval ability (related to the requirement of instruction-following ability). We compare with the baseline that gets source evidence by direct reflection, GE-Chat improves on retrieval performance.

2 Approach

We discuss the details of the **GE-Chat** framework, which builds upon the graph-based retrieval augmented generation agent (Graph-RAG), and then innovates on a three-step

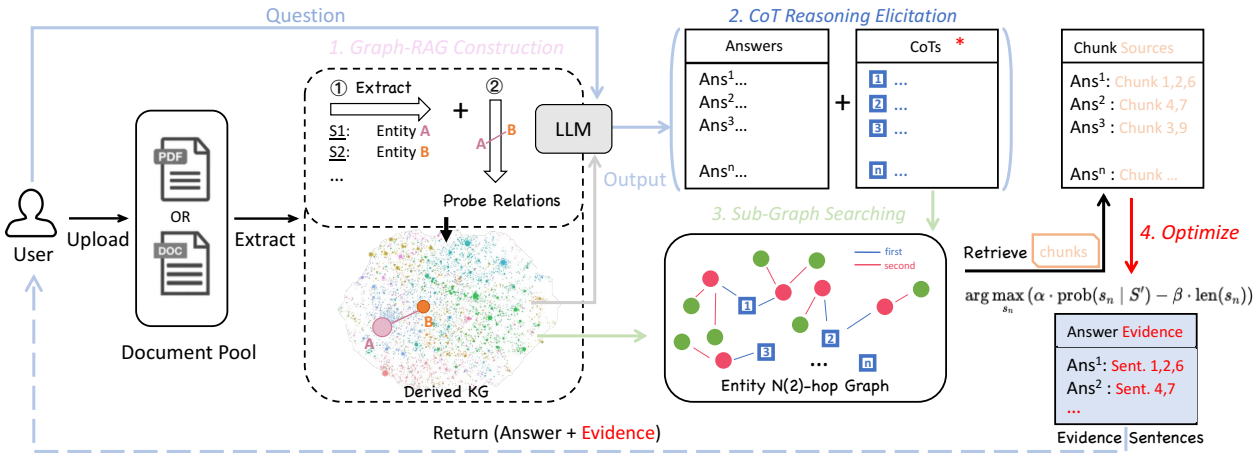


Figure 1: The overview of the GE-Chat framework. As shown in this pipeline, when user uploads the document, it is used for the 1. Graph-RAG Construction, which contains two main steps using LLMs, ① Extract the entities A, B, etc., from the document chunks, then ② Probe the contextual relations between these entities. Then a derived Knowledge Graph is formed and used for question answering. In order to realize evidence generation on Graph-RAG, 2. CoT Reasoning Elicitation is proposed to elicit the reasoning chain for answers. Then we have 3. Sub-Graph Searching based on Entity Matching and N-hop Relations Probing, this sub-graph contains entities and relations used to retrieve the source chunks, guaranteeing originality of content. We then 4. Optimize toward balancing meaningfulness and conciseness.

paradigm for better evidential response generation; we introduce from constructing the RAG agent Sec. 2.1), and then explain three components in our framework (Sec. 2.2, 2.4).

2.1 Graph-RAG Construction

The Graph-RAG [Larson and Truitt, 2024] is unique in its ability to integrate external information through a structured knowledge graph, thus supporting graph-based queries and allowing for relational reasoning. Besides, it also does well in handling multi-hop reasoning, this feature helps to find more than one-hop-related entities in the knowledge base, extract sub-graphs, and capture the relational semantics clusters. After a user uploads a document in (.TXT/.DOC/.PDF) format, the metadata will be cut into corpus chunks to temporarily store the file, then we construct a knowledge graph $\mathcal{G}$ by two steps: ① extract entities from the chunks, and ② probe the relations among the entities. The LLM is used to achieve the knowledge graph (KG) and KG is used back to the LLM as external information to make responses. We constructed a fast and lightweight Graph-RAG following the implementations [Guo *et al.*, 2024].

2.2 CoT Reasoning Elicitation

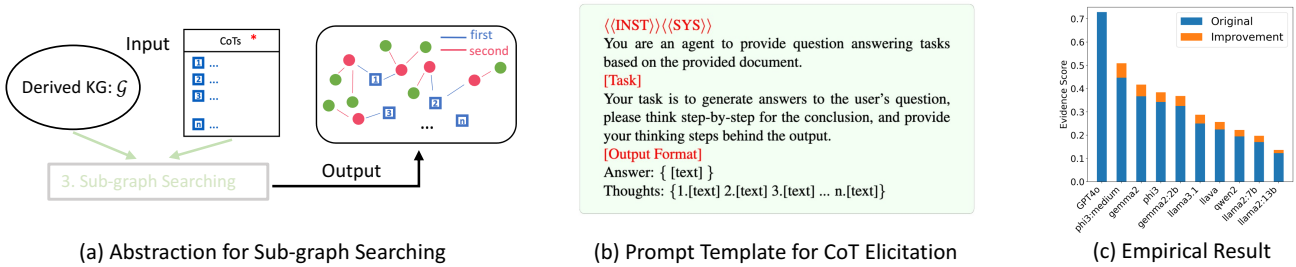
This section introduces the Chain-of-Thought (CoT) reasoning inducer, which serves as a primary step in deriving the reasoning process. It is well acknowledged that the majority of the LLMs can automatically perform CoT [Wei *et al.*, 2022] to elicit the reasoning process, i.e., how they draw the conclusion step by step. We follow the same idea to induce the logic steps $Logic\_steps\{step_1, step_2, ..., step_n\}$ from the LLM given a question Q on a document. By designing a CoT template, we tend to achieve the following:

$$Answer, Logic\_steps = \text{CoT\_template}(Q, Doc) \quad (1)$$

The template is inspired by work [Zhang *et al.*, 2022] and is shown in the above green block. This step corresponds to the upper right part of Fig. 1, `CoT Reasoning Elicitation`, where each answer is associated with a CoT chain that explains the reasoning process step by step. These CoT chains, generated by RAG models, provide a logical structure to responses, but may not inherently align with the raw content of the submitted document. To ensure the evidence strictly originates from the provided source, a critical grounding step is introduced through sub graph searching based on entity matching. This process anchors CoT reasoning to specific entities and relationships within the knowledge graph, bridging the gap between generated content and its original context. By doing so, we enhance the trustworthiness and accuracy of the responses while maintaining a clear traceability to the original document.

2.3 Efficient Sub-Graph Searching

The sub-graph searching is conducted based on two resources: Derived KG $\mathcal{G}$ and CoTs as in Figure 2. For each of the CoT results: $c_i \in \{c_1, c_2, c_3, ..., c_n\}$, the c_i will be used to match the most relevant graph entities (Entity Matching), as shown in the output part, the blue boxes are the identical c_i, and the ● is connected by blue edges, which is the first hop most relevant entities, this is what c_i has been involved in the LLM's answer, then we relax this relation to further second-hop as shown in ● green dot. This search is efficient for the CoT guidance and pre-calculated $\mathcal{G}$ for n-hop relationships probing (in contrast to the whole document-range global search). Finding this entity sub-graph is like finding an anchor that leads to the original source chunks, we can perform `Source Chunk Retrieval` to get several chunks for each c_i in CoTs. This step bridges the made-up CoT content with the source content of documents by finding the anchor entity. However, chunk-level descriptions leave space for more fine-grained evidence sentences [Zhao *et al.*, 2024], so we employ an optimization objective in the next section.



(a) Abstraction for Sub-graph Searching (b) Prompt Template for CoT Elicitation (c) Empirical Result

Figure 2: The illustration of (a) sub-graph searching, (b) prompt templates for CoT, and (3) the comparison between the original LLMs evidence generation and LLMs with GE-Chat framework. GE-Chat outperforms existing LLMs in evidence generation.

2.4 Evidence Content Optimization

As in Figure 1, based on the chunk sources (e.g.,chunk1,2,6), for each of the c_i in CoTs, we expect to find a finer sentence from a certain chunk that best supports the answer sentence S' with minimal redundant information. We first formalize this problem, and then provide a solution in corresponds to 4. Optimize in the Fig. 1 uses the entailment probability.

Problem Setting: Given a chunk of text containing n sentences, $S = \{s_1, s_2, \ldots, s_n\}$, and a target sentence S' which represents the sentence in the answering content, the objective is to find the best sentence $s_{\text{best}} \in S$ that: 1. Maximizes the entailment probability $\text{prob}(s_n \mid S')$, which measures how strongly s_n entails S', 2. Minimizes the sentence length $\text{len}(s_n)$, encouraging concise representations.

In order to balance these two criteria, we define an objective function $\mathcal{F}(s_n)$, which assigns a score to each sentence s_n based on its contained meaning and conciseness. The score for each sentence s_n is given by: $\mathcal{F}(s_n) = \alpha \cdot \text{prob}(s_n| - S') - \beta \cdot \text{len}(s_n)$, where α and β are set as 0.5 to control the weight of the entailment probability, and penalty for longer sentences, respectively. We want to measure how much the generated evidence means similarly to the answer, a rational way is to calculate the entailment probability $\text{prob}(s_n| - S')$. We achieve this by using NLI model [1], which provides a three-element tuple by taking two text pairs s_n and S': $[logit_{cont}, logit_{neut}, logit_{ent}] = \overrightarrow{NLI}(s_n, S')$. The output is processed by transforming into the probability through $\mathbf{p} = Softmax(logit_{cont}, logit_{neut}, logit_{ent})$. Then we can calculate the $\overrightarrow{p_{ent}}(s_n, S') = p(s_n \vdash S') = \mathbf{p}_3$ as the entailment probability. The optimal sentence s_{best} is the one that maximizes $\mathcal{F}(s_n)$: $s_{\text{best}} = \arg\max_{s_n \in S}(\mathcal{F}(s_n))$ Using this objective, we can find the best evidence that supports the answers in the LLM's responses, and this action is performed in a small chunk, which is not computationally expensive and can be deployed in real time.

The s_{best} will be calculated for each of the answers, such as in Figure 1, the best evidence output for `Ans`[1] is a combination of sentences `Sent.1, 2, 6`. This will be returned back to users together for users to understand which part of the answer comes with the evidence supported and which part lacks such trustworthy information, helping practitioners understand the reliability of generated content.

[1]we use off-the-shelf DeBERTa-large model [He *et al.*, 2021]

3 Experiment

Dataset: To address the scarcity of evidence sources in prior research, we created a dataset with 1000 cases to evaluate evidence generation quality across 10 categories: Biology, Chemistry, etc, with three dimensions: (1) PDF length—short (<10), medium (10-100), and long (>100 pages); (2) Question types—synthesis, and (3) human-annotated answers with corresponding evidence sentences, ensuring reliability and comprehensiveness. We tested our method on this dataset and have released the dataset and videos [2] for public use with standard questions, groundtruth answers, and evidence for reference. **Evaluation Metric:** Following existing work [LangChain, 2023], we use the cosine similarity to evaluate the relevance of the generated text (evidence) with the correct evidence and use the conciseness score [Ragas authors, 2024] to quantify how precise response is. In combination, we have the following score(↑):

$$\text{Evid.}_{\text{score}} = \frac{1}{N}\sum_{i=1}^{N}\left[\cos(E_i, E_{gt_i}) \cdot \min\left(1, \frac{L_{gt_i}}{L_i}\right)\right] \quad (2)$$

where the E_i is the embedding of the generated evidence for question i, and E_{gt_i} is the groundtruth evidence. The first term measures the cosine similarity of two evidence (↑), and L_{gt_i} is the length of text for groundtruth evidence while the L_i is the generated evidence, $\frac{L_{gt_i}}{L_i}$ measures conciseness.

3.1 Experiment Results and Conclusion

In the experiment, the direct evidence retrieval ability of GPT4o is the best, while other models perform worse, especially lacking conciseness. After applying GE-Chat to existing models except for GPT4o (GPT4o is involved in the process of ground-truth 'reference' generation). The results are shown in Fig. 2 (c). In summary, we presented GE-Chat, a novel framework addressing the trustworthiness of LLMs by evidence retrieval and verification. Through constraints on source derivation and sentence-level highlight capabilities, GE-Chat significantly enhances the reliability of LLM-generated responses. By offering a transparent and user-friendly approach, GE-Chat contributes to making AI systems more reliable and trustworthy, paving the way for responsible deployment in critical decision-making processes.

[2]Click the link to the testset and videos.

Acknowledgments

The work was partially supported by NSF awards #2421839, NAIRR #240120 and used AWS through the CloudBank project, which is supported by NSF grant #1925001. The views and conclusions in this paper are those of the authors and should not be interpreted as representing any funding agencies. Thank OpenAI for providing API credits under the Researcher Access program and Amazon Research Awards.

References

[Da *et al.*, 2024] Longchao Da, Kuanru Liou, Tiejin Chen, Xuesong Zhou, Xiangyong Luo, Yezhou Yang, and Hua Wei. Open-ti: Open traffic intelligence with augmented language model. *International Journal of Machine Learning and Cybernetics*, pages 1–26, 2024.

[Ding *et al.*, 2024] Hanxing Ding, Liang Pang, Zihao Wei, Huawei Shen, and Xueqi Cheng. Retrieve only when it needs: Adaptive retrieval augmentation for hallucination mitigation in large language models. *arXiv preprint arXiv:2402.10612*, 2024.

[Følstad and Skjuve, 2019] Asbjørn Følstad and Marita Skjuve. Chatbots for customer service: user experience and motivation. In *Proceedings of the 1st international conference on conversational user interfaces*, 2019.

[Guo *et al.*, 2024] Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. Lightrag: Simple and fast retrieval-augmented generation. *arXiv preprint arXiv:2410.05779*, 2024.

[Han *et al.*, 2024] Rujun Han, Yuhao Zhang, Peng Qi, Yumo Xu, Jenyuan Wang, Lan Liu, William Yang Wang, Bonan Min, and Vittorio Castelli. Rag-qa arena: Evaluating domain robustness for long-form retrieval augmented question answering. 2024.

[He *et al.*, 2021] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021.

[Hu *et al.*, 2024] Wenbo Hu, Yifan Xu, Yi Li, Weiyue Li, Zeyuan Chen, and Zhuowen Tu. Bliva: A simple multimodal llm for better handling of text-rich visual questions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2256–2264, 2024.

[Ji *et al.*, 2023] Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. Towards mitigating llm hallucination via self reflection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1827–1843, 2023.

[LangChain, 2023] LangChain. Evaluating rag pipelines with ragas + langsmith. https://blog.langchain.dev/evaluating-rag-pipelines-with-ragas-langsmith/, 2023. Accessed: 2025-02-11.

[Larson and Truitt, 2024] Jonathan Larson and Steven Truitt. Graphrag: Unlocking llm discovery on narrative private data, 2024.

[Lin, 2024] Demiao Lin. Revolutionizing retrieval-augmented generation with enhanced pdf structure recognition. *arXiv preprint arXiv:2401.12599*, 2024.

[Liu *et al.*, 2025] Xiaoou Liu, Tiejin Chen, Longchao Da, Chacha Chen, Zhen Lin, and Hua Wei. Uncertainty quantification and confidence calibration in large language models: A survey. *arXiv preprint arXiv:2503.15850*, 2025.

[Ragas authors, 2024] Ragas authors. *conciseness*, 2024. Accessed: 2025-02-11.

[Saad-Falcon *et al.*, 2023] Jon Saad-Falcon, Joe Barrow, Alexa Siu, Ani Nenkova, Ryan A Rossi, and Franck Dernoncourt. Pdftriage: question answering over long, structured documents. *arXiv preprint arXiv:2309.08872*, 2023.

[Shuster *et al.*, 2021] Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*, 2021.

[Wang *et al.*, 2024] Lei Wang, Jiabang He, Shenshen Li, Ning Liu, and Ee-Peng Lim. Mitigating fine-grained hallucination by fine-tuning large vision-language models with caption rewrites. In *International Conference on Multimedia Modeling*, pages 32–45. Springer, 2024.

[Wei *et al.*, 2022] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[Wei *et al.*, 2024] Rongxuan Wei, Kangkang Li, and Jiaming Lan. Improving collaborative learning performance based on llm virtual assistant. In *2024 13th International Conference on Educational and Information Technology (ICEIT)*, pages 1–6. IEEE, 2024.

[Yuan *et al.*, 2024] Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Jia Deng, Boji Shan, Huimin Chen, Ruobing Xie, Yankai Lin, et al. Advancing llm reasoning generalists with preference trees. *arXiv preprint arXiv:2404.02078*, 2024.

[Zhang *et al.*, 2022] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*, 2022.

[Zhang *et al.*, 2024] Jiaxing Zhang, Jiayi Liu, Dongsheng Luo, Jennifer Neville, and Hua Wei. Llmexplainer: Large language model based bayesian inference for graph explanation generation, 2024.

[Zhao *et al.*, 2024] Jihao Zhao, Zhiyuan Ji, Pengnian Qi, Simin Niu, Bo Tang, Feiyu Xiong, and Zhiyu Li. Metachunking: Learning efficient text segmentation via logical perception. *arXiv preprint arXiv:2410.12788*, 2024.