HGOT: Hierarchical Graph of Thoughts for Retrieval-Augmented In-Context Learning in Factuality Evaluation

Yihao Fang[1,3], **Stephen W. Thomas**[1] **and Xiaodan Zhu**[2,3]

[1]Smith School of Business, Queen's University
[2]Department of Electrical and Computer Engineering, Queen's University
[3]Ingenuity Labs Research Institute, Queen's University
yihao.fang@gmail.com, {stephen.thomas, xiaodan.zhu}@queensu.ca

Abstract

With the widespread adoption of large language models (LLMs) in numerous applications, the challenge of factuality and the propensity for hallucinations has emerged as a significant concern. To address this issue, particularly in retrieval-augmented in-context learning, we introduce the hierarchical graph of thoughts (HGOT), a structured, multi-layered graph approach designed to enhance the retrieval of pertinent passages during in-context learning. The framework utilizes the emergent planning capabilities of LLMs, employing the divide-and-conquer strategy to break down complex queries into manageable sub-queries. It refines self-consistency majority voting for answer selection, which incorporates the recently proposed citation recall and precision metrics to assess the quality of thoughts, linking an answer's credibility intrinsically to the thought's quality. This methodology introduces a weighted system in majority voting, prioritizing answers based on the citation quality of their thoughts. Additionally, we propose a scoring mechanism for evaluating retrieved passages, considering factors such as citation frequency and quality, self-consistency confidence, and the retrieval module's ranking. Experiments indicate that HGOT excels as a versatile approach, outperforming competing models in FEVER by up to 7% and matching leading models such as Retrieve-then-Read in Open-SQuAD, and DSP in HotPotQA, demonstrating its efficacy in enhancing LLMs' factuality.

1 Introduction

The advancement of large language models (LLMs) (Devlin et al., 2019; Raffel et al., 2020; Radford et al., 2018, 2019; Brown et al., 2020) has revolutionized the field of NLP and artificial intelligence by offering unprecedented capabilities in natural language understanding and generation, leading to their widespread adoption in many applications. However, a critical challenge of these models is the tendency to "hallucinate" (Maynez et al., 2020; Raunak et al., 2021; Bouyamourn, 2023)—generating content that is factually incorrect or not grounded in reality. This issue raises significant concerns about the reliability and trustworthiness of LLMs, particularly in high-stakes applications. While numerous efforts have been made to address various aspects of this problem, a specific area that demands attention is retrieval-augmented in-context learning (Lazaridou et al., 2022; Izacard et al., 2022; Press et al., 2022; Khattab et al., 2022), a process where LLMs leverage external information to enhance their responses.

In response to the challenge of hallucinations, we introduce the hierarchical graph of thoughts (HGOT) framework, drawing inspiration from neuropsychological studies on the "hierarchy of goals" and working memory (Cowan, 2010; Jonides et al., 2008; Cowan, 2005). Our approach redefines how LLMs interact with and utilize external information sources. By constructing a structured, multi-layered graph (Ying et al., 2018; Chen et al., 2022), HGOT allows for a more organized and efficient way of sourcing and incorporating relevant information, thereby reducing the incidence of hallucinations in LLMs. Despite these advances, the challenges that we need to overcome involve dynamically constructing a hierarchical graph, as well as evaluating and ranking the qualities of thoughts and retrieved passages in this complex structure.

The HGOT framework places a strong emphasis on the dynamic creation of a hierarchical graph structure by exploring the applicability of the emergent planning capabilities of LLMs (Wang et al., 2023a; Valmeekam et al., 2023) in breaking down complex queries (higher in the hierarchy) into simpler sub-queries (lower in the hierarchy). This method employs a divide-and-conquer strategy, which simplifies the problem-solving process and improves the accuracy and relevance of the information retrieved by the LLM.

Another key feature of the HGOT framework is the improvement of the self-consistency majority voting mechanism (Wang et al., 2023b) used in LLMs, which enhances the quality assessment of thoughts or rationales. This improvement assesses the quality of thoughts or rationales generated by the LLMs. The method utilizes metrics such as citation recall and precision (Gao et al., 2023) to evaluate the quality of the information used by the LLMs in forming their responses. The underlying premise is that the quality of an LLM's response is directly related to the quality of its underlying thought. Therefore, in the majority voting process, responses are given weights based on the citation quality of their thoughts.

Furthermore, the HGOT framework proposes a scoring mechanism to evaluate the quality of retrieved passages. This mechanism takes into account various factors, including the frequency of passage citation, the citation quality (Gao et al., 2023) of the thought, self-consistency confidence score (Xiong et al., 2023; Wang et al., 2023b), and the retrieval module ranking. By considering these diverse factors, the mechanism ensures that the information utilized in the LLM's response generation is both relevant and of high quality.

To validate the effectiveness of the proposed method, we selected FEVER (Thorne et al., 2018), Open-SQuAD (Rajpurkar et al., 2016; Karpukhin et al., 2020), and HotPotQA (Yang et al., 2018) to evaluate the models' proficiency in fact retrieval and reasoning. We divided these datasets into three groups: "Long", "Medium", and "Short", according to the question length, emphasizing sampling from the tails of the distribution, a detail that is frequently overlooked in studies. Experiments show that HGOT outperforms existing retrieval-augmented in-context learning methods in FEVER by up to 7% and matching leading models such as Retrieve-then-Read (Lazaridou et al., 2022; Izacard et al., 2022) in Open-SQuAD, and Demonstrate-Search-Predict (DSP) (Khattab et al., 2022) in HotPotQA, underscoring its robustness and efficacy in enhancing LLMs' factuality.

In brief, we make the following contributions:

- We introduce HGOT and investigate LLM's (emergent) planning ability in breaking down complex queries for graph construction.
- **Thought Quality:** HGOT selects the best answer by voting which involves assessing thought quality with citation recall and precision metrics.
- **Retrieval Quality:** We propose a scoring mechanism for evaluating retrieved passages based on citation frequency and quality, self-consistency confidence, and retrieval module ranking.
- We conduct extensive experiments on FEVER, Open-SQuAD, and HotPotQA, emphasizing sampling from the extremes of the distribution. The results demonstrate HGOT's efficacy in enhancing LLMs' factuality.

2 Related Work

The "Retrieve-then-Read" pipeline (Lazaridou et al., 2022; Izacard et al., 2022) sends queries to a retrieval model (RM) to gather passages for a prompt that a language model (LM) uses for response generation. "Self-ask" (Press et al., 2022) and "Iterative Retriever, Reader, and Reranker" (IRRR) (Qi et al., 2020) improve upon this approach through multi-hop retrieval, enabling the LM to ask follow-up questions that the RM answers. These answers, combined with the original prompt, enhance the LM's ability to respond to the initial question.

"ReAct" (Yao et al., 2023b) uses LLMs to generate reasoning traces and task-specific actions in an interleaved manner. While reasoning traces help the model induce, actions allow it to interface with external sources. Baleen (Khattab et al., 2021) summarizes multiple passages of information in each hop to be used in subsequent iterations. The "Demonstrate-Search-Predict" (DSP) approach (Khattab et al., 2022) enhances the multi-hop methodologies by automatically annotating "chain-of-thought" (Wei et al., 2022) demonstrations. The potential weakness of those multi-hop pipelines lies in the generality and adaptability of their search operations. Especially, those pipelines face challenges when tasked with addressing inquiries that necessitate intricate planning for the retrieval of pertinent information.

Plan-and-Solve (PS) Prompting (Wang et al., 2023a) involves breaking down complex tasks into manageable subtasks and executing them according to a formulated plan, with PS+ prompting enhancing reasoning quality through detailed instructions. However, PS hasn't yet utilized LLMs' planning capabilities with retrieval-augmented in-context learning. Other methods such as the "tree of thoughts" (Yao et al., 2023a), "graph of thoughts" (Besta et al., 2023), and RECURRENTGPT (Zhou et al., 2023) explore reasoning via tree, graph, or recurrent structures to improve problem-solving,

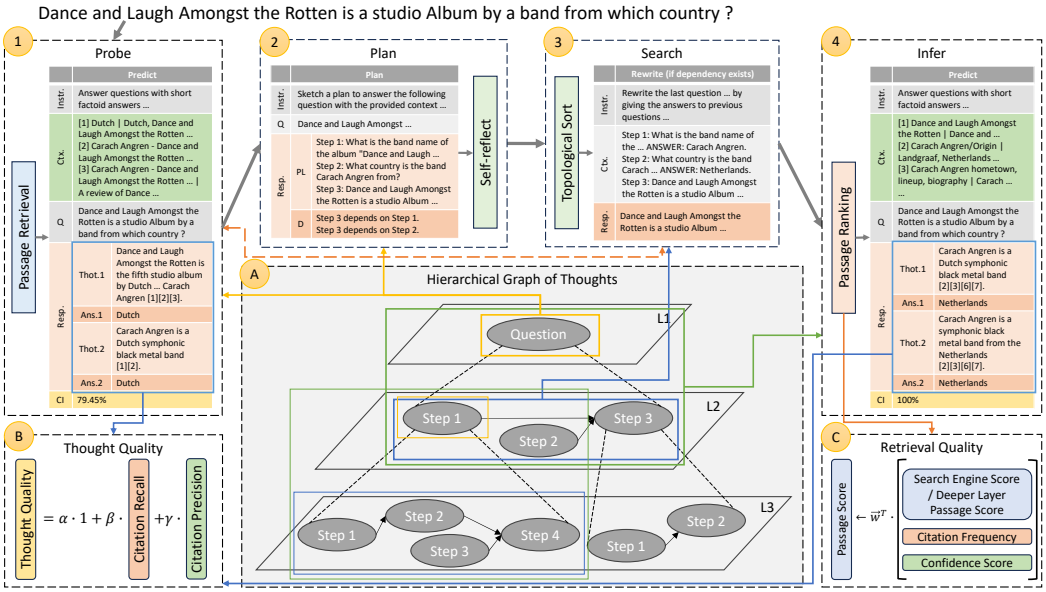

Figure 1: An illustrative example of HGOT in answering a factual question. (The abbreviations employed are as follows: Instr.: Instructions, Q: Question, Ctx.: Context or References, Resp.: ChatGPT's Response, PL: Plan, D: Dependencies, CI: Confidence, Ans.: Answer, Thot.: Thought)

but they face challenges in sourcing relevant information, suffering from drawbacks concerning the factual reliability of large language models.

3 Methodology

The HGOT framework involves creating a multi-layered graph that allows for a more organized and efficient sourcing and incorporation of relevant information. This structure aims to reduce the occurrence of hallucinations in LLMs. However, the initial challenges that we need to overcome involve dynamically constructing hierarchical graphs, along with assessing and ranking the qualities of thoughts and retrieved passages within this complex structure.

In terms of hierarchical graph construction, the HGOT framework utilizes the emergent planning ability of LLMs to break down complex queries into smaller, more manageable sub-queries (or steps), following a divide-and-conquer strategy.

To select the best answer for a query, the framework employs a method of improving self-consistency majority voting (Wang et al., 2023b). This involves assessing the quality of thoughts using citation recall and precision metrics and weighing answers based on the citation quality of their thoughts (Figure 1: Ⓑ).

Additionally, a scoring mechanism is proposed for evaluating the quality of retrieved passages. This mechanism takes into account various factors such as the frequency of passage citation, the quality of citations in the thoughts, a self-consistency confidence score adjusted for citation quality, and the retrieval module's ranking (Figure 1: Ⓒ).

3.1 Hierarchical Graph Construction, Search, and Inference

Graph Construction: When utilizing the emergent planning ability to break down a complex question into smaller, more manageable sub-queries or steps, it's crucial to recognize that these sub-queries or steps are not standalone. Instead, they often exhibit interconnections that contribute to forming a complete answer. These steps and their connections create a dependency graph within a deeper level of the hierarchical graph, which guides the exploration of the complex question. (In this framework, the dependency graph is designed as a directed acyclic graph to avoid circular dependencies.) Further, each sub-query can be extended into a more detailed dependency graph at even deeper levels of the hierarchy. For example, as illustrated in Figure 1: Ⓐ, a query at the initial layer (Layer 1 or L1) can be extended into a dependency graph at a subsequent layer (Layer 2 or L2). Within L2, the first step could unfold into a four-step dependency graph in the next layer (Layer 3 or L3), while the third step in L2 might lead to a two-step dependency graph at the same third layer (L3).

Establishing a precise dependency graph is essential before progressing to the subsequent stage,

as any error or ambiguity at this stage could significantly derail the solution path. To accurately infer this graph, there are several strategies that we can adopt. Initially, we employed the "Probe" procedure to gather references (referenced in Figure 1: ① and Appendix C.5). This involves collecting passages from the retrieval model and then scoring these passages by prompting LLM to probe for an answer. The specifics of how passages are scored will be discussed in Section 3.3.

Subsequently, we designed the prompt template for the "Plan" procedure (Figure 1: ② and Appendix C.1). This template incorporates instructions, demonstrations (see Appendix D), and the collected passages. The aim is to stimulate the LLM and guide it towards a holistic understanding of the question and its interconnected components.

Once the "Plan" procedure is complete, we introduce the self-reflection technique (Appendix C.2), inspired by the work of Shinn et al. (2023). This involves prompting the LLM again to double-check if the output dependencies are accurate and align with the question in each step. The method encourages the LLM to focus internally on the dependencies without external influence, by providing only related steps or sub-queries. Finally, we formalize these dependencies into a structure that is more compatible with programming language formats (Appendix C.3).

Search: A crucial aspect of this stage involves using topological sorting and rewriting, as shown in Figure 1: ③. Topological sorting within a dependency graph (i.e., a directed acyclic graph) ensures that steps influencing subsequent steps are processed in a sequential order. When evaluating a step or a sub-query, a "Probe" procedure is employed (refer to Figure 1: ①), which gathers passages from the retrieval model and instructs the LLM to search for an answer by using the sub-query. In the context of the dependency graph, when Step 2 is contingent on Step 1, the question in Step 2 is rewritten (see Appendix C.4) to include the sub-query from Step 1 along with the answer obtained from the "Probe" procedure. This process ensures that the interconnections are well-articulated and traceable within the graph.

The "Probe" procedure for each sub-query does more than seek answers; it also gathers and scores relevant passages. Additionally, the "Plan" procedure is applied to each sub-query to create a dependency graph at a deeper level. Following this, the "Search" procedure (Figure 1: ③) investigates the dependency graph topologically, and the "Infer" procedure (Figure 1: ④) is then utilized to calculate the final scores for all the passages collected in the earlier stages, to predict the answer, and to determine the confidence score. In each step or sub-query assessed during the "Search" procedure, the "Probe", "Plan", "Search", and "Infer" procedures are recursively executed until a specified depth of the graph is achieved, or the "Plan" procedure opts to stop further progression. Specifically, the termination condition is activated if the "Plan" procedure results in only a single step that closely resembles the sub-query being planned. The similarity between them is assessed using the cosine similarity of their BERT-based sentence embeddings (Reimers and Gurevych, 2019).

Algorithm 1 HGOT Traversal

▷ Let q be a question
▷ Let a be an answer. e.g., a_q is the answer to q
▷ Let $\mathbf{G}$ be a dependency graph (i.e., a directed acyclic graph)
▷ Let $\mathbf{CTX}$ be the context (incl. passages and scores)
▷ Let $\mathbf{CI}$ be a confidence score
▷ Let d be the level of depth in the hierarchical

procedure TRAVERSE(q, d)
  $a_q, \mathbf{CI}_q, \mathbf{CTX}_q \leftarrow$ PROBE(q)
  $\mathbf{G} \leftarrow$ PLAN($q, \mathbf{CTX}_q$)
  if STOP($q, \mathbf{G}, d$) **then**
    return $a_q, \mathbf{CI}_q, \mathbf{CTX}_q$
  else
    $\mathbf{CTX_G} \leftarrow$ SEARCH($\mathbf{G}, d+1$)
    $a_q, \mathbf{CI}_q, \mathbf{CTX} \leftarrow$ INFER($q, \mathbf{CTX}_q, \mathbf{CTX_G}$)
    return $a_q, \mathbf{CI}_q, \mathbf{CTX}$
  end if
end procedure

procedure SEARCH($\mathbf{G}, d$)
  $q_1, ..., q_r \leftarrow$ TOPOLOGICAL_SORT($\mathbf{G}$)
  for i **in** $1...r$ **do**
    $q_i \leftarrow$ REWRITE(q_i, IN_NEIGHBORS($q_i, \mathbf{G}$))
    $a_{q_i}, \mathbf{CI}_{q_i}, \mathbf{CTX}_{q_i} \leftarrow$ TRAVERSE(q_i, d)
  end for
  return $\mathbf{CTX}_{q_1}, ..., \mathbf{CTX}_{q_r}$
end procedure

Inference: Having the hierarchical graph of thoughts and their related passages collected from the retrieval model, the "Infer" procedure predicts the final answer to the query (Figure 1: ④). Specifically, this procedure ranks all passages retrieved during the examination of the query and its sub-queries, as will be explained in Section 3.3. It subsequently selects the top K passages with the highest rankings to use as the prompt for LLM. Along with demonstrations and instructions, the

"Infer" procedure asks LLM to think step by step, predicts the final answer, and estimates the confidence score (Appendix C.5 and Appendix D). The algorithm for recursive planning, searching, and inferring within HGOT is detailed in Algorithm 1.

3.2 Thought Quality

When assessing the quality of thoughts, we establish tuples $(\tau_1, a_1), ..., (\tau_m, a_m)$ as pairs of LLM-generated thoughts (rationales) and answers, as shown in Figure 1: ①, ④, and Ⓑ. The quality of a thought τ_i is determined by modifying the concepts of citation recall (REC) and citation precision (PREC) as introduced by Gao et al. (2023), in the following manner:

$$\rho_i \coloneqq \alpha \cdot 1 + \beta \cdot \text{REC}(\tau_i) + \gamma \cdot \text{PREC}(\tau_i) \quad (1)$$

Assuming there are d distinct responses $\hat{a}_1, ..., \hat{a}_d$, with d being less than or equal to m, we improve upon the self-consistency majority voting method (Wang et al., 2023b) by factoring in the thought qualities, defining the selected answer as:

$$\hat{a}^* = \underset{\hat{a}_h \in \{\hat{a}_1, ..., \hat{a}_d\}}{\arg\max} \sum_{i=1}^{m} \rho_i \delta(a_i, \hat{a}_h) \quad (2)$$

where δ is the Kronecker delta function, which equals 1 when the variables are the same and 0 otherwise.

Moreover, we develop the self-consistency confidence score (Xiong et al., 2023) by taking into account the thought qualities. This is defined as:

$$\textbf{CI} = \frac{\sum_{i=1}^{m} \rho_i \delta(a_i, \hat{a}^*)}{\sum_{i=1}^{m} \rho_i} \quad (3)$$

Note that when α equals 1 and both β and γ are zero, these equations are simplified to the prediction and calibration based on self-consistency (Wang et al., 2023b; Xiong et al., 2023).

3.3 Retrieval Quality

Assessing the quality of retrieved passages considers multiple aspects. These include how often the passage is cited, the quality of these citations (Gao et al., 2023), a self-consistency confidence score (Xiong et al., 2023), and the ranking given by the retrieval module (Figure 1: Ⓒ).

Assume p is a particular passage retrieved, which serves as a part of the context in the "Probe" or "Infer" procedures. The pairs $(\tau_1, a_1), ..., (\tau_m, a_m)$ represent the generated thoughts (rationales) and answers produced when using ChatGPT with a temperature greater than zero. Statements or sentences $s_1, ..., s_{l_{\tau_i}}$ are parts of τ_i. The process of natural language inference (denoted as a function NLI) and a citation marker at the end of each statement (denoted as M) work together to determine if a statement s_j cites passage p, resulting in a value of either true or false. This is formally expressed as:

$$\hat{\delta}(p, s_j) = \begin{cases} 1, & \text{if M}(p, s_j) \text{ or NLI}(p, s_j) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

We further define the "weighted citation frequency per thought" for a given passage p, as the total number of citations in τ_i, adjusted by the quality of the thought τ_i. Formally, it is presented as:

$$\nu(p, \tau_i) = \rho_i \sum_{j=0}^{l_{\tau_i}} \hat{\delta}(p, s_j) \quad (5)$$

The "weighted citation frequency" is the aggregate of these "weighted citation frequencies per thought" across all thoughts, and is denoted by:

$$\hat{\nu}(p) = \sum_{i=0}^{m} \nu(p, \tau_i) \quad (6)$$

Next, we normalize this "weighted citation frequency" so that the highest value among all passages from a specific retrieval P, to which p belongs, is equal to 1. The "normalized weighted citation frequency" is thus:

$$\bar{\nu}(p) = \frac{\hat{\nu}(p)}{\max_{p \in P} \hat{\nu}(p)} \quad (7)$$

Finally, during the "Probe" or "Infer" procedures, the quality score of the passage p is updated repetitively, starting with the initial score $\sigma(p, 0)$ provided by the search engine in the "Probe" procedure. The formula is expressed as follows:

$$\sigma(p, t+1) \leftarrow \vec{w}^T \cdot \begin{bmatrix} \sigma(p, t) \\ \bar{\nu}(p) \\ \textbf{CI} \end{bmatrix} \quad (8)$$

where $\vec{w} = (w_1, w_2, w_3)$ is a hyperparameter vector that can be tuned for different datasets, retrieval models and large language models.

4 Data

We evaluate HGOT across three datasets: FEVER (Thorne et al., 2018), Open-SQuAD (Rajpurkar

et al., 2016; Karpukhin et al., 2020), and HotPotQA (Yang et al., 2018). Considering the use of sentence length as a parameter for estimating complexity has been implemented in various NLP tasks (Platanios et al., 2019; Spitkovsky et al., 2010), to assess HGOT across different complexity levels, we stratify the three datasets based on sentence length, categorizing them into long, medium, and short.

Figure 2: The sentence length, measured by the number of tokens in a question, from the FEVER, Open-SQuAD, and HotPotQA datasets

The sentence length, measured by the number of tokens in a question, from the FEVER, Open-SQuAD, and HotPotQA datasets is illustrated in Figure 2. The median number of tokens in FEVER is 27, with a long tail of instances extending beyond the median (indicating possible complexity in reasoning, see Appendix B for a more in-depth examination of the data). Open-SQuAD and HotPotQA likewise exhibited a similar distribution. The training, development, and test distributions align well with each other, enabling the stratification of these datasets by sentence length.

Sent. Len.	FEVER			Open-SQuAD			HotPotQA		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
Long	1619	113	113	1174	121	118	1504	168	137
Medium	2182	150	150	1181	133	159	1628	181	148
Short	2182	150	150	1181	133	159	1628	181	148

Table 1: Count of examples across all three datasets and nine categories (Refer to Appendix A for summary statistics and Appendix B for data examples)

Questions from FEVER and Open-SQuAD that exceed the 98.5^{th} percentile in length are categorized as long, while for HotPotQA, this categorization applies to questions above the 98^{th} percentile. For questions of FEVER and Open-SQuAD that fall between the 1.5^{th} and 98.5^{th} percentiles, they are defined as medium length, and for HotPotQA, this range is from the 2^{nd} to the 98^{th} percentile. Within this group of medium-length questions, about 1.5% of those from FEVER and Open-SQuAD are randomly chosen for evaluation, compared to 2% of HotPotQA questions. Additionally, questions from FEVER and Open-SQuAD below the 1.5^{th} percentile are labelled as short, similar to those under the 2^{nd} percentile for HotPotQA questions. Lastly, Table 1 displays the total number of examples across all three datasets, spanning nine categories.

Metrics: For Open-SQuAD and HotPotQA, we utilize the Exact Match (EM) and F1 scores (Rajpurkar et al., 2016). The EM score identifies the proportion of predictions that precisely align with the correct answers, while the F1 score assesses the average token overlap between the prediction and the correct answer. For FEVER, we only use the EM score, considering the answers in FEVER being limited to three tokens or fewer.

5 Evaluation Setup

Baselines: Our benchmarking includes five approaches: "Vanilla LM" (Brown et al., 2020), "Retrieve-then-Read" (Lazaridou et al., 2022; Izacard et al., 2022), "Self-ask" (Press et al., 2022), "ReAct" (Yao et al., 2023b), and "Demonstrate-Search-Predict" (DSP) (Khattab et al., 2022). See Appendix E for further details.

Implementation Details: All approaches employed ChatGPT (gpt-3.5-turbo-1106) as the backbone LLM, with the exception of ReAct, which utilized text-davinci-002, given that the ReAct project[1] has not incorporated gpt-3.5-turbo-1106. For the retrieval model, we used the Google Search API provided by SerpApi.com, following the "Self-ask" approach (Press et al., 2022). HGOT[2] was implemented using Python language and the DSP framework (Khattab et al., 2022). Following Gao et al. (2023), We adopt a natural language inference (NLI) model (Honovich et al., 2022) in HGOT to measure thought quality and retrieval quality. Additionally, the topological sorting and deductions pertaining to HGOT were performed using the Python NetworkX[3] package.

6 Experimental Results

Findings and Analysis: The baseline models, referred to as "Vanilla LM", utilize few-shot in-context learning on ChatGPT without being augmented by retrieval models. These "Vanilla LM"

[1]https://github.com/ysymyth/ReAct
[2]https://github.com/fangyihao/hgot
[3]https://networkx.org/

Method	FEVER	Open-SQuAD		HotPotQA		FEVER	Open-SQuAD		HotPotQA	
	EM	EM	F1	EM	F1	EM	EM	F1	EM	F1
	Overall					Long				
Vanilla LM	54.72	17.43	33.91	33.58	43.93	43.36	16.10	34.22	24.09	38.15
Retrieve-then-Read	58.35	22.51	**38.81**	41.20	51.21	46.90	**29.66**	44.60	35.77	50.05
Self-ask	53.03	18.81	34.15	43.98	54.67	46.90	20.34	35.10	42.34	59.32
ReAct	45.04	-	-	35.47	42.18	34.51	-	-	17.52	24.62
DSP	55.45	20.65	36.09	47.23	**61.13**	47.79	23.73	39.08	**45.26**	**64.27**
HGOT+Sampling (Ours)	**61.50**	22.05	36.11	45.03	56.07	53.98	28.81	42.21	37.23	53.36
HGOT+KNN (Ours)	60.53	**24.10**	38.32	**47.37**	59.48	**54.87**	28.81	**46.27**	43.07	59.77
	Medium					Short				
Vanilla LM	54.00	26.42	41.10	29.73	40.63	64.00	9.43	26.49	44.59	51.59
Retrieve-then-Read	59.33	28.30	**43.14**	35.81	45.43	66.00	11.32	**30.12**	50.68	57.88
Self-ask	52.00	27.04	41.05	41.89	51.92	58.67	9.43	26.53	47.30	53.92
ReAct	45.33	-	-	33.11	40.69	52.67	-	-	51.35	56.89
DSP	55.33	28.93	42.51	41.89	57.17	61.33	10.06	27.41	**54.05**	**62.72**
HGOT+Sampling (Ours)	57.33	27.67	40.25	41.89	53.33	**71.33**	11.32	27.38	**54.05**	60.87
HGOT+KNN (Ours)	**61.33**	**31.45**	42.17	**46.62**	**59.21**	64.00	**13.21**	28.47	51.35	59.54

Table 2: A comparative analysis of Vanilla LM, Retrieve-then-Read, Self-ask, ReAct, DSP, and HGOT. The "Overall" section is derived by calculating the weighted average of metrics from the "Long", "Medium", and "Short" categories, using the number of examples in each category as weights.

models closely mirror the fundamental capabilities of ChatGPT as assessed in our factuality evaluation datasets. We observe that "Vanilla LM" generally excels at responding to short questions (or claims in FEVER), except when it comes to short Open-SQuAD questions (refer to Table 2). This exception is consistent with our dataset analysis (see Appendix B for details), where it is found that longer questions (or claims in FEVER) often demand the gathering of more facts and the undertaking of more complex reasoning. Conversely, questions of medium and short length in Open-SQuAD usually require identifying one or two specific pieces of knowledge. However, medium-length questions provide more context than the shorter ones.

Methods other than "Vanilla LM" include those that are augmented by retrieval mechanisms. In comparison, these retrieval-augmented approaches generally surpass the performance of "Vanilla LM", except in cases involving Self-ask and ReAct within the FEVER dataset (see the "Overall" section in Table 2). Additionally, the DSP method shows weaker performance in the FEVER dataset. This suggests that the ability to gather factual information is more crucial in FEVER than the capability for multi-hop reasoning. Our approaches, HGOT+Sampling and HGOT+KNN (with HGOT+Sampling and HGOT+KNN representing HGOT combined with the demonstration selection methods of "balanced sampling" or "k-nearest neighbors", as detailed in Appendix D), are versatile and exhibit strong performance across all three datasets, regardless of whether they prioritize the skill of accumulating factual data or conducting multi-hop comprehension and reasoning.

Specifically for the FEVER dataset, HGOT+Sampling secures the top position, with HGOT+KNN closely behind in second place. With a 61.50% EM score, HGOT+Sampling outperforms Retrieve-then-Read, which is third, by a margin of over 3% (refer to the "Overall" section in Table 2). In every length category of the FEVER dataset, namely "Long", "Medium", and "Short", either HGOT+Sampling or HGOT+KNN achieves the highest ranking. Notably, HGOT+Sampling exceeds DSP, the strongest baseline, by more than 7% in the "Long" category and surpasses Retrieve-then-Read by more than 5% in the "Short" category, where Retrieve-then-Read is the top among baselines. In the "Medium" category, Retrieve-then-Read competes closely with HGOT+KNN, underscoring the importance of fact-gathering over complex reasoning in FEVER, in line with findings in Appendix B. Moreover, both HGOT+Sampling and HGOT+KNN, on average, excel beyond Retrieve-then-Read's achievements in these scenarios.

Within the Open-SQuAD dataset, as detailed in Table 2's "Overall" section, HGOT+KNN stands out as the top performer, recording an EM score of 24.10%, which is over 1.5% higher than its nearest competitor, Retrieve-then-Read. HGOT+KNN also leads in EM scores for both the "Medium" and "Short" categories and achieves the highest F1 score in the "Long" category of the dataset. Retrieve-then-Read demonstrates strong competitiveness in the Open-SQuAD dataset, closely matching HGOT+KNN's performance across all categories,

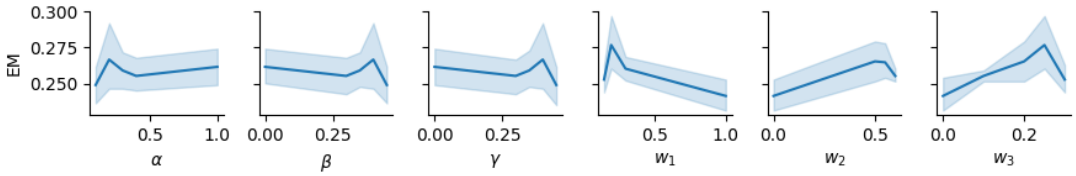

Figure 3: The visualizations of the hyperparameter searches are shown through pairwise relationships, featuring the EM score in the row and hyperparameters α, β, γ, w_1, w_2, and w_3 in the columns. Each subplot is represented as a line chart, aggregating the data to display the mean (solid blue line) and the 95% confidence interval (light blue area). Additionally, the optimal hyperparameters for attaining the highest EM score are indicated in each subplot.

in contrast to DSP, which shows weaker performance. This observation is consistent with our analysis in Appendix B, revealing that a large portion of the Open-SQuAD questions are designed to extract factual information, mainly asking "What", "How", and "When".

In the HotPotQA dataset, known for demanding multi-hop reasoning capabilities from models, HGOT+KNN achieved the top position in the total EM score. For the "Medium" category, HGOT+KNN recorded the highest EM score at 46.62%, surpassing the second-best performers, HGOT+Sampling, DSP, and Self-ask, by 4.73%. Additionally, in this category, HGOT+KNN led in F1 score, outperforming the second-ranked DSP by over 2%. DSP proved to be a strong contender across the board in the HotPotQA dataset, closely matching the performance of our HGOT+KNN model, whereas the Retrieve-then-Read model fell short. This performance trend corroborates our dataset examination in Appendix B, confirming the necessity for models to possess robust multi-hop reasoning skills for the HotPotQA dataset.

Ablation Study: We examine the effect of the presence or absence of thought quality and retrieval quality, as well as how HGOT's performance varies with different hyperparameters. More precisely, we explore how the EM score interacts with the hyperparameters α, β, and γ as shown in Equation 1, and also how EM score relates to each element of $\vec{w} = (w_1, w_2, w_3)$ as detailed in Equation 8. Specifically, setting $\alpha = 1$, $\beta = 0$, and $\gamma = 0$ in Equation 1 is equivalent to a situation where thought quality is not considered, reducing the model to rely solely on prediction and calibration through self-consistency, as discussed in Wang et al. (2023b). Similarly, when $w_1 = 1$, $w_2 = 0$, and $w_3 = 0$ in Equation 8, it simulates a condition where retrieval quality is disregarded, with the ranking of retrieved passages depending only on the search engine's score.

We include hyperparameter settings of $\alpha = 1$, $\beta = 0$, and $\gamma = 0$, alongside $w_1 = 1$, $w_2 = 0$, and $w_3 = 0$, to equalize the absence of thought quality and to simulate the absence of retrieval quality when searching for HGOT+KNN's optimal hyperparameter configurations for the medium-length category in the Open-SQuAD dataset. Figure 3 illustrates the EM scores associated with varying values of each hyperparameter. It is observed that the optimal EM score is attained with hyperparameter values of $\alpha = 0.2$, $\beta = 0.4$, $\gamma = 0.4$, $w_1 = 0.2$, $w_2 = 0.55$, and $w_3 = 0.25$, as detailed in Table 7 in Appendix F. This suggests that the optimal combination of hyperparameters can be identified with the presence of thought quality and retrieval quality, emphasizing the significance of introducing these qualities into the model (see Appendix F for additional results from the ablation study).

7 Conclusion

In our factuality evaluation, we chose FEVER, Open-SQuAD, and HotPotQA to assess models' abilities in both fact retrieval and reasoning. We segmented the datasets FEVER, Open-SQuAD, and HotPotQA into three categories: "Long", "Medium", and "Short", based on the length of their questions. This categorization emphasizes the significance of examining both extremely short and long questions, an aspect often overlooked in research. We introduced HGOT. This approach structures thoughts in a hierarchical graph format, leveraging emergent planning capabilities. It evaluates thoughts and retrieved passages by introducing metrics for thought and retrieval qualities, thereby safeguarding HGOT's capabilities in reasoning and fact-finding. Experiments show that HGOT stands out as a versatile approach, surpassing other models in FEVER and matching leading models such as Retrieve-then-Read in Open-SQuAD, and DSP in HotPotQA.

Limitations

HGOT employs OpenAI's ChatGPT for its language model, whereas alternative models such as Google's Gemini and Meta's Llama 2 have not been explored. HGOT's evaluation is conducted using the Google Search API from SerpApi.com as its retrieval model. Its performance could vary, either improve or decline, when used in conjunction with other search engines such as Microsoft Bing, Yahoo, and Baidu. Additionally, the retrieval model for HGOT could potentially include various domain-specific data sources, for example, this could involve aligning queries with pertinent information in relational databases such as Oracle and IBM's DB2, which are widely used in the finance industry. However, the effectiveness of these variant implementations has not been examined.

Ethics Statement

We ensure that all data utilized is publicly available and refrain from involving any private data. We affirm that our research focuses on assessing factuality and deliberately avoids producing harmful or undesirable content.

References

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michal Podstawski, Hubert Niewiadomski, Piotr Nyczyk, et al. 2023. Graph of thoughts: Solving elaborate problems with large language models. *arXiv preprint arXiv:2308.09687*.

Adam Bouyamourn. 2023. Why LLMs hallucinate, and how to get (evidential) closure: Perceptual, intensional, and extensional learning for faithful natural language generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3181–3193, Singapore. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Cen Chen, Kenli Li, Wei Wei, Joey Tianyi Zhou, and Zeng Zeng. 2022. Hierarchical graph neural networks for few-shot learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(1):240–252.

Nelson Cowan. 2005. *Working memory capacity*. Psychology press.

Nelson Cowan. 2010. Multiple concurrent thoughts: The meaning and developmental neuropsychology of working memory. *Developmental neuropsychology*, 35(5):447–474.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. *arXiv preprint arXiv:2305.14627*.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. TRUE: Re-evaluating factual consistency evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3905–3920, Seattle, United States. Association for Computational Linguistics.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2022. Few-shot learning with retrieval augmented language models. *arXiv preprint arXiv:2208.03299*.

John Jonides, Richard L Lewis, Derek Evan Nee, Cindy A Lustig, Marc G Berman, and Katherine Sledge Moore. 2008. The mind and brain of short-term memory. *Annu. Rev. Psychol.*, 59:193–224.

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

Omar Khattab, Christopher Potts, and Matei Zaharia. 2021. Baleen: Robust multi-hop reasoning at scale via condensed retrieval. *Advances in Neural Information Processing Systems*, 34:27670–27682.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2022. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive nlp. *arXiv preprint arXiv:2212.14024*.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot

prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

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

Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom Mitchell. 2019. Competence-based curriculum learning for neural machine translation. 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 1162–1172, Minneapolis, Minnesota. Association for Computational Linguistics.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*.

Peng Qi, Haejun Lee, Oghenetegiri Sido, Christopher D Manning, et al. 2020. Answering open-domain questions of varying reasoning steps from text. *arXiv preprint arXiv:2010.12527*.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

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*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Vikas Raunak, Arul Menezes, and Marcin Junczys-Dowmunt. 2021. The curious case of hallucinations in neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1172–1183, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik R Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Valentin I. Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. 2010. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 751–759, Los Angeles, California. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Karthik Valmeekam, Matthew Marquez, Sarath Sreedharan, and Subbarao Kambhampati. 2023. On the planning abilities of large language models–a critical investigation. *arXiv preprint arXiv:2305.15771*.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023a. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. *arXiv preprint arXiv:2305.04091*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William 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*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. *arXiv preprint arXiv:2305.10601*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. 2018. Hierarchical graph representation learning with differentiable pooling. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Wangchunshu Zhou, Yuchen Eleanor Jiang, Peng Cui, Tiannan Wang, Zhenxin Xiao, Yifan Hou, Ryan Cotterell, and Mrinmaya Sachan. 2023. Recurrentgpt: Interactive generation of (arbitrarily) long text. *arXiv preprint arXiv:2305.13304*.

A Dataset Summary Statistics

Table 3 presents a comparison of the FEVER, Open-SQuAD, and HotPotQA datasets across nine evaluated categories in our experiments. For each category, we assess the total number of instances, as well as the maximum, minimum, and median lengths of questions, in addition to calculating the mean and standard deviation for question lengths. It is noted that the question lengths in all three categories of the Open-SQuAD dataset are generally shorter compared to the equivalent categories in the FEVER and HotPotQA datasets. Furthermore, the "Long" and "Medium" categories exhibit larger standard deviations in question length across all three datasets when compared to the "Short" categories.

Dataset	Sentence Length	Split	Number of Examples	Maximum Length	Minimum Length	Median	Mean	Standard Deviation
FEVER	Long	Train	1619	125	38	40	44.33	12.89
		Dev	113	57	37	38	39.22	3.58
		Test	113	53	39	41	42.33	3.24
	Medium	Train	2182	37	24	27	27.51	2.82
		Dev	150	36	24	27	27.49	2.63
		Test	150	37	24	27	27.81	2.90
	Short	Train	2182	23	21	23	22.81	0.40
		Dev	150	23	21	23	22.81	0.41
		Test	150	23	22	23	22.76	0.43
Open-SQuAD	Long	Train	1174	60	22	23	24.42	3.18
		Dev	121	36	22	24	24.55	2.86
		Test	118	34	23	24	25.02	2.55
	Medium	Train	1181	21	6	11	11.26	3.29
		Dev	133	20	6	11	11.41	3.29
		Test	159	19	6	11	11.53	3.34
	Short	Train	1181	5	1	5	4.72	0.57
		Dev	133	5	4	5	4.83	0.38
		Test	159	5	3	5	4.79	0.47
HotPotQA	Long	Train	1504	128	58	66	69.46	10.96
		Dev	168	120	59	66	69.12	10.31
		Test	137	57	34	36	37.66	3.98
	Medium	Train	1628	57	10	17	19.49	8.33
		Dev	181	58	10	18	20.23	9.80
		Test	148	33	10	17	17.71	5.43
	Short	Train	1628	9	4	9	8.43	0.91
		Dev	181	9	5	9	8.43	0.90
		Test	148	9	7	9	8.57	0.65

Table 3: Summary statistics across three datasets FEVER, Open-SQuAD, and HotPotQA and nine categories

B Dataset Examples and Examination

B.1 FEVER Data Examples and Examination

The FEVER dataset necessitates that the model gathers relevant background information or context regarding the subject, such as knowing what the Boeing 767 is as stated in the claim "The Boeing 767 became the most frequently used airliner for transatlantic flights between North America and Europe in the 1990s" (Table 4). Subsequently, it is required to conduct logical analysis on all the specific facts collected. Claims that are longer typically require the accumulation of more facts and knowledge, as well as the undertaking of more sophisticated reasoning. As a result, the complexity of a claim is often proportional to its length.

Sentence Length	Claim	Answer
Long	The Boeing 767 became the most frequently used airliner for transatlantic flights between North America and Europe in the 1990s.	SUPPORTS
	In Kentucky, the electric chair has been kept in operation except for those whose capital crimes were committed prior to March 31, 1998, and who choose electrocution.	REFUTES
	The House of the Spirits is about the life of a young lady named Clara during the military dictatorship in Algeria.	REFUTES
	One Flew Over the Cuckoo's Nest won the five major Academy Awards the year it was released, the second film to do so.	NOT ENOUGH INFO
	In 2012, Simi Valley, California, reported a higher median household income than that of the nation overall.	SUPPORTS
Medium	Planet Hollywood Las Vegas is operated by all entities except an American gaming corporation.	REFUTES
	Chris Bosh plays in the National Basketball Association as a professional basketball player.	SUPPORTS
	Pierce County, Washington is the location of the lowest mountain in Washington.	NOT ENOUGH INFO
	The Airbus A380 entered commercial service on October 25, 2017.	REFUTES
	The Nobel Prize in Chemistry was awarded to a person from the Kingdom of the Netherlands.	SUPPORTS
Short	Estonia is a country.	SUPPORTS
	Edward Cullen was created.	NOT ENOUGH INFO
	Dopamine prevents neuromodulation.	REFUTES
	Backing vocalists are performers.	SUPPORTS
	Reanimation is a book.	NOT ENOUGH INFO

Table 4: FEVER data examples

B.2 Open-SQuAD Data Examples and Examination

As demonstrated in Table 5 of the Open-SQuAD dataset, the bulk of questions are focused on "What", "How", "When", and "Why", requiring the accumulation of factual data for answers. Additionally, questions of medium and short length typically need the collection of one or two specific pieces of information or knowledge. For instance, the question "In what geographical portion of Wales is Abercynon located?" necessitates identifying the specific location of Abercynon within Wales. Notably, medium-length questions tend to offer more context for information retrieval compared to those in the "Short" category, such as "What is septicemia?". Thus, the inclusion of "Short" category questions in Open-SQuAD doesn't suggest they are easy to answer, especially for models that find it challenging to gather factual data. Conversely, "Long" category questions usually demand more extensive fact-finding and

complex reasoning.

Sentence Length	Question	Answer
Long	What was the number of times the Denver Broncos played in a Super Bowl by the time they reached Super Bowl 50?	eight
	What is the application of prime numbers used in information technology which utilizes the fact that factoring very large prime numbers is very challenging?	public-key cryptography
	When did the UMC's General Board of Church and Society call on all United Methodists to abstain from alcohol for Lent?	2011 and 2012
	What is the minimum distance between a patient's home and the nearest pharmacy that allows a physician in Austria to give out medicine?	more than 4 kilometers
	Approximately how many names were signed on an online petition on the Parliamentary website in response to the closing of the Musical Instruments gallery?	over 5,100
Medium	In what geographical portion of Wales is Abercynon located?	south
	How long has the Doctor Who Magazine been in circulation?	since 1979
	What social construct did Huguenot refugees in Canterbury practice?	economic separation
	Why were Johann Esch and Heinrich Voes executed by the Catholic Church?	for Lutheran views
	Who was the first known European to visit China and return?	Marco Polo
Short	What is septicemia?	a type of "blood poisoning"
	What shape are Plastoglobuli?	spherical bubbles
	What do carotenoids absorb?	light energy
	What is a prasinophyte?	a green algal derived chloroplast
	What was Apple Talk	a proprietary suite of networking protocols developed by Apple Inc

Table 5: Open-SQuAD data examples

B.3 HotPotQA Data Examples and Examination

HotPotQA questions typically demand from the model not only the skill to accumulate factual data but, more importantly, the capability for multi-hop comprehension and reasoning, particularly with long questions. For instance, to answer the question (refer to Table 6), "What is the genus of the viral disease that has symptoms such as fever, chills, loss of appetite, nausea, muscle pains, and headaches, and has a chance of causing liver damage?" the model is required to initially identify details about "the viral disease

that has symptoms such as fever, chills, loss of appetite, nausea, muscle pains, and headaches" alongside information on "the viral disease that has a chance of causing liver damage", before determining the genus of the virus in question. Therefore, the degree of complexity for a HotPotQA question often correlates with its length.

Sentence Length	**Question**	**Answer**
Long	Out of two American colonies that had a series of skirmishes and raids between 1701 and 1765 at the disputed border, which British proprietary colony became a royal colony on the northeast coast of North America?	Province of New York
	Which Captain launched the attack which led to more casualties than any other incident in the war fought between the settlers of the nascent colony of New Netherland and the native Lenape population?	Captain John Underhill
	Lost Kingdom Adventure is a dark ride located at four Legoland theme parks, including which park, which is the original Legoland park, that was opened on June 7th, 1968?	Legoland Billund
	What is the genus of the viral disease that has symptoms such as fever, chills, loss of appetite, nausea, muscle pains, and headaches, and has a chance of causing liver damage?	Flavivirus
	Until what year did the Chief of Justice of the Supreme Court that administered the presidential oath of office to Abraham Lincoln on his first inauguration as the 16th President of the United States hold that office?	1864
Medium	The Last Run is a drama film that stars which Lithuanian-American actor?	Vyto Ruginis
	What part of Australia is Alice River and Rupertswood in?	Victoria
	What was the nationality of the composer of Chaconne in F minor?	German
	What was the breakthrough role of the actor starring in Good Boy! and was a native of Atlanta?	Tai Frasier in "Clueless"
	Who played the role of Nettie Harris in the 1985 film directed by Steven Spielberg?	Akosua Gyamama Busia
Short	What empire was Aleksei Gen born into?	Russian Empire
	Romans stars which Tamil and Telugu actress?	Nivetha Thomas
	Are Ari Up and Boz Burrell both guitarists?	no
	Are Tetrastigma and Spruce both types of plants?	yes
	What did Karan Kapoor's maternal grandfather deliver?	Shakespeare performances

Table 6: HotPotQA data examples

C Prompt and Response Examples

C.1 Prompt and Response of the "Plan" Procedure

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ PROMPT ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
.................................user.................................
Sketch a plan to answer the following question with the provided context. List only
    ↪  the essential steps which can be answered by search engines. Express each
    ↪ step as a standalone search question. Highlight interdependencies if any.
    ↪ Higher number steps can depend on lower number steps, while the reverse is
    ↪ not possible.

---

Follow the following format.

Context:
${sources that may contain relevant content. e.g., [1] Passage 1. [2] Passage 2.
    ↪ [3] Passage 3.}

Question: ${the question to be answered}

Plan:
Step 1: ${a standalone search question. e.g., ...?} Step 2: ${a standalone search
    ↪ question. e.g., ...?} ... Step n: ${a standalone search question. e.g.,
    ↪ ...?}

Dependencies: ${interdependencies among multiple steps. e.g., Step ... depends on
    ↪ Step ... .}

---

Context:
[1] Steve Masiello | (born September 2, 1977) is an American college basketball
    ↪ coach and a former player. He most recently served as men's head coach at
    ↪ Manhattan College.
[2] Jaspers' new coach hopes to recapture MC's past glory | Manhattan College
    ↪ introduced Steve Masiello, center, who will take over as the Jaspers' new
    ↪ men's basketball coach.
[3] Steve Masiello (St. John's Red Storm) | Steve Masiello (born September 2, 1977)
    ↪ . Current position: Associate head men's basketball coach. Current team: St.
    ↪  John's Red Storm (Head ...

Question: Which of the Manhattan Jaspers basketball team head coach was born in
    ↪ September 2, 1977?

Plan:
Step 1: Who is the head coach of the Manhattan Jaspers basketball team? Step 2:
    ↪ When was the head coach born?

Dependencies: Step 2 depends on Step 1.
```

```
---

Context:
[1] Phil Cutchin | Phil Cutchin (September 9, 1920 - January 7, 1999) was an
    ↪ American football player and coach. He served as the head football coach at
    ↪ Oklahoma State ...
[2] Former OSU Football Coach Cutchin Dies | In life, Phil Cutchin captained a Paul
    ↪  "Bear" Bryant football team, was an Army officer in two wars, a football
    ↪ coach and a stock broker.
[3] Phil Cutchin | American Football Database | Fandom | Phil Cutchin (September 9,
    ↪  1920 - January 7, 1999) was an American football player and coach. He
    ↪ served as the head football coach at Oklahoma State ...

Question: Coach Phil Cutchin served as the head football coach at Oklahoma State-
    ↪ University-Stillwater, which was originally known as what?

Plan:
Step 1: What was Oklahoma State University-Stillwater originally known as? Step 2:
    ↪ When did Phil Cutchin serve as the head football coach at Oklahoma State
    ↪ University-Stillwater?

Dependencies: Step 2 depends on Step 1.

---

Context:
Todd Boehly | Todd Boehly is an American businessman and investor. He is the co-
    ↪ founder, chairman, chief executive officer and controlling member of
    ↪ Eldridge Industries, ...

Question: What was Todd Boehly's former position at the firm where Mark Walter is
    ↪ the CEO?

Plan:
---------------------------------- RESPONSE ----------------------------------
---------------------------------- CHOICE 0 ----------------------------------
.................................assistant.................................
Step 1: What is the name of the firm where Mark Walter is the CEO? Step 2: What was
    ↪  Todd Boehly's former position at the firm where Mark Walter is the CEO?

Dependencies: Step 2 depends on Step 1.
```

C.2 Prompt and Response of the "Self-reflect" Procedure

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ PROMPT ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
.................................user.................................
Highlight interdependencies among the steps below if any. Higher number steps can
    ↪ depend on lower number steps, while the reverse is not possible.

---

Follow the following format.
```

```
Plan:
Step 1: ${a standalone search question. e.g., ...?} Step 2: ${a standalone search
    ↪ question. e.g., ...?} ... Step n: ${a standalone search question. e.g.,
    ↪ ...?}

Dependencies: ${interdependencies among multiple steps. e.g., Step ... depends on
    ↪ Step ... .}

---

Plan:
Step 1: Who is the head coach of the Manhattan Jaspers basketball team? Step 2:
    ↪ When was the head coach born?

Dependencies: Step 2 depends on Step 1.

---

Plan:
Step 1: What was Oklahoma State University-Stillwater originally known as? Step 2:
    ↪ When did Phil Cutchin serve as the head football coach at Oklahoma State
    ↪ University-Stillwater?

Dependencies: Step 2 depends on Step 1.

---

Plan:
Step 1: What is the name of the firm where Mark Walter is the CEO? Step 2: What was
    ↪  Todd Boehly's former position at the firm where Mark Walter is the CEO?

Dependencies:
---------------------------------- RESPONSE ----------------------------------
---------------------------------- CHOICE 0 ----------------------------------
.................................assistant.................................
Step 2 depends on Step 1.
```

C.3 Prompt and Response of the "Formalize" Procedure

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ PROMPT ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
.................................user.................................
Express the dependencies in formal language by giving the descriptions below.

---

Follow the following format.

Descriptions: ${descriptions of dependencies}
Dependencies: ${e.g., If Step 2 depends on Step 1, then write Step 1 -> Step 2; If
    ↪ Step 2 and Step 3 depend on Step 1, then write Step 1 -> (Step 2 and Step 3)
    ↪ ; If Step 3 depends on Step 1 and Step 2, then write (Step 1 and Step 2) ->
```

```
    ↪ Step 3}

---

Descriptions: Step 2 depends on Step 1.
Dependencies:
----------------------------------- RESPONSE -----------------------------------
---------------------------------- CHOICE 0 ------------------------------------
.................................assistant.....................................
Step 1 -> Step 2
```

C.4 Prompt and Response of the "Rewrite" Procedure

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ PROMPT ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
.................................user.....................................
Rewrite the last question in a standalone manner by giving the answers to previous
    ↪ questions. Do not consider answers that were not specified. Only show the
    ↪ last question after the rewrite.

---

Follow the following format.

Context:
${previous questions and answers}

Rewrite: ${the last question after the rewrite}

---

Context:
Step 1: Who is the head coach of the Manhattan Jaspers basketball team? ANSWER:
    ↪ John Gallagher. Step 2: When was the head coach born?

Rewrite: When was the head coach of the Manhattan Jaspers basketball team born?

---

Context:
Step 1: What was Oklahoma State University-Stillwater originally known as? ANSWER:
    ↪ Oklahoma Agricultural and Mechanical College. Step 2: When did Phil Cutchin
    ↪ serve as the head football coach at Oklahoma State University-Stillwater?

Rewrite: When did Phil Cutchin serve as the head football coach at Oklahoma State
    ↪ University-Stillwater?

---

Context:
Step 1: What is the name of the firm where Mark Walter is the CEO? ANSWER:
    ↪ Guggenheim Partners. Step 2: What was Todd Boehly's former position at the
    ↪ firm where Mark Walter is the CEO?
```

```
Rewrite:
---------------------------------- RESPONSE ----------------------------------
---------------------------------- CHOICE 0 ----------------------------------
.................................assistant.................................
What was Todd Boehly's former position at Guggenheim Partners?
```

C.5 Prompt and Response of the "Predict" Procedure

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ PROMPT ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
.................................user.................................
Answer questions with short factoid answers.

---

Follow the following format.

Context:
${sources that may contain relevant content. e.g., [1] Passage 1. [2] Passage 2.
    ↪ [3] Passage 3.}

Question: ${the question to be answered}

Rationale: Let's think step by step. ${a step-by-step deduction that identifies the
    ↪  correct response, which will be provided below. Every statement in the "
    ↪ Rationale" section should be attributable to the passages provided in the "
    ↪ Context" section. e.g., ...[1][2].}

Answer: ${a short factoid answer, often between 1 and 5 words}

---

Context:
[1] List of Manhattan Jaspers men's basketball head coaches | Manhattan's current
    ↪ head coach is John Gallagher. He was hired in March 2023, replacing RaShawn
    ↪ Stores, who was not promoted to the full-time position after ...
[2] Steve Masiello | Stephen John Masiello Jr. (born September 2, 1977) is an
    ↪ American college basketball coach and a former player. He most recently
    ↪ served as men's head coach ...
[3] Steve Masiello | (born September 2, 1977) is an American college basketball
    ↪ coach and a former player. He most recently served as men's head coach at
    ↪ Manhattan College.
[4] Manhattan College Appoints John Gallagher to Lead Men's ... | - John Gallagher
    ↪ has been named the new Head Men's Basketball Coach at Manhattan College, it
    ↪ was announced today by Director of Athletics ...
[5] List of Manhattan Jaspers men's basketball head coaches | Manhattan's current
    ↪ head coach is John Gallagher. He was hired in March 2023, replacing RaShawn
    ↪ Stores, who was not promoted to the full-time position after ...
[6] Jaspers' new coach hopes to recapture MC's past glory | Manhattan College
    ↪ introduced Steve Masiello, center, who will take over as the Jaspers' new
    ↪ men's basketball coach.
```

```
[7] Men's Basketball Coaches | Head Coach, 718-862-7533 718-862-7533 .
    ↪ jgallagher06@manhattan.edu, First Year ; Assistant Coach, 718-862-7533
    ↪ 718-862-7533 . tim.brooks@manhattan.edu, First ...

Question: Which of the Manhattan Jaspers basketball team head coach was born in
    ↪ September 2, 1977?

Rationale: Let's think step by step. Steve Masiello was born on September 2, 1977
    ↪ [2][3]. John Gallagher is the current head coach of the Manhattan Jaspers
    ↪ basketball team [1][4][5].

Answer: Steve Masiello

---

Context:
[1] Oklahoma Agricultural and Mechanical College | Oklahoma Agricultural and
    ↪ Mechanical College, Founded on Christmas Day in 1890 under the Morrill Act
    ↪ as Oklahoma Agricultural and Mechanical College, Oklahoma State University
    ↪ has grown through its traditions and culture to become one of America's
    ↪ premier land-grant universities., Oklahoma Agricultural and Mechanical
    ↪ College
[2] Oklahoma State University-Stillwater | OSU was founded in 1890 under the
    ↪ Morrill Act. Originally known as Oklahoma Agricultural and Mechanical
    ↪ College (Oklahoma A&M), it is the flagship institution ...
[3] 1963 to 1968 | 1963 to 1968, Phil Cutchin (September 9, 1920 - January 7, 1999)
    ↪  was an American football player and coach. He served as the head football
    ↪ coach at Oklahoma State University-Stillwater from 1963 to 1968, compiling a
    ↪  record of 19-38-2., 1963 to 1968
[4] Former OSU Football Coach Cutchin Dies | Cutchin was head football coach at
    ↪ Oklahoma State from 1963 to 1968. He won only 19 games, but most all of his
    ↪ 40 defeats were given up ...
[5] Phil Cutchin | Phil Cutchin (September 9, 1920 - January 7, 1999) was an
    ↪ American football player and coach. He served as the head football coach at
    ↪ Oklahoma State ...
[6] OSU History | The college's first students attended classes in the Stillwater
    ↪ Congregational Church. The original campus consisted of 200 acres of prairie
    ↪  that were ...
[7] Phil Cutchin | American Football Database | Fandom | He served as the head
    ↪ football coach at Oklahoma State University-Stillwater from 1963 to 1968,
    ↪ compiling a record of 19-38-2. Although he never had a winning ...

Question: Coach Phil Cutchin served as the head football coach at Oklahoma State-
    ↪ University-Stillwater, which was originally known as what?

Rationale: Let's think step by step. Oklahoma Agricultural and Mechanical College
    ↪ [1][2].

Answer: Oklahoma Agricultural and Mechanical College

---
```

```
Context:
[1] Unions file lawsuit challenging Wisconsin Act 10 | Former Republican Gov. Scott
    ↪  Walker signed the law in 2011 despite some of the largest protests in state
    ↪  history, and the law has since shaped the state's political landscape.,
    ↪ Scott Walker
[2] Act 10 turns 10: Four takeaways from the law that shook ... | Here's a look at
    ↪ how the law limiting collective bargaining for most public workers has
    ↪ played out.
[3] Act 10 turns 10: Four takeaways from the law that shook ... | Act 10 ended the
    ↪ ability of public-sector unions to negotiate over any issues other than
    ↪ raises, and those raises were capped at the rate of ...
[4] Wisconsin Teachers Sue to Restore Collective Bargaining ... | The law, which
    ↪ was championed by former Republican Gov. Scott Walker, has been challenged
    ↪ unsuccessfully in court before. But the political context has changed: The
    ↪ Wisconsin Supreme Court recently flipped to liberal control for the first
    ↪ time in 15 years., Scott Walker
[5] Wis. governor officially cuts collective bargaining | Scott Walker has
    ↪ officially taken away nearly all collective bargaining rights from the vast
    ↪ majority of the state's public employees. Walker ...
[6] 10 years later, Wisconsinites are still divided over Act 10 | Former Gov. Scott
    ↪  Walker's landmark legislation required public employees to pay more for
    ↪ their pensions and health care and limited their ...
[7] Wisconsin's Act 10 limitations on collective bargaining | With its 5-2 vote
    ↪ upholding the law, the Wisconsin Supreme Court gave an important nod towards
    ↪  the constitutionality of limits of collective bargaining rights ...

Question: Which Wisconsin state governor oversaw a vote to significantly limit
    ↪ public employee collective bargaining?

Rationale: Let's think step by step. Former Republican Governor Scott Walker
    ↪ oversaw a vote to significantly limit public employee collective bargaining
    ↪ [1][4][5][6][7].

Answer: Scott Walker

---

Context:
[1] Mark Walter | 184 Mark Walter on the 2023 Forbes 400 - Mark Walter is CEO of
    ↪ investment firm Guggenheim Partners, which has over $300 billion in assets
    ↪ under management.
[2] Todd Boehly - Milken Institute | Boehly was the President of Guggenheim
    ↪ Partners. He received his B.B.A. from the College of William & Mary, where
    ↪ he later founded the Boehly Center for Excellence in Finance, and studied at
    ↪  the London School of Economics., President
[3] Katie & Todd Boehly | Prior to founding Eldridge Industries, Todd was President
    ↪  of Guggenheim Partners and founded the credit business at Guggenheim. He
    ↪ received his B.B.A. from The College of William & Mary, where he later
    ↪ founded The Boehly Center for Excellence in Finance, and studied at the
    ↪ London School of Economics., Prior to founding Eldridge Industries, Todd was
    ↪  President of Guggenheim Partners and founded the credit business at
    ↪ Guggenheim
```

```
[4] Katie & Todd Boehly | Prior to founding Eldridge Industries, Todd was President
    ↪  of Guggenheim Partners and founded the credit business at Guggenheim.,
    ↪ President of Guggenheim Partners
[5] Guggenheim Partners | Guggenheim Partners, In 2000, he helped found Guggenheim
    ↪ Partners. Today, he is the CEO of Guggenheim Partners, which has rapidly
    ↪ grown into a global, diversified financial services firm with more than $260
    ↪  billion in assets under management, 2,300 employees, and 25 offices in six
    ↪ countries around the world., Guggenheim Partners
[6] Todd Boehly | Todd Boehly cofounded holding company Eldridge in 2015, after
    ↪ building Guggenheim Partners' credit business and serving as president.
    ↪ Boehly acquired some ...
[7] Todd L. Boehly | Boehly was President of Guggenheim Partners and founded the
    ↪ credit business at Guggenheim. He received his B.B.A. from The College of
    ↪ William & Mary, where ...

Question: What was Todd Boehly's former position at the firm where Mark Walter is
    ↪ the CEO?

Rationale: Let's think step by step.
---------------------------------- RESPONSE ----------------------------------
---------------------------------- CHOICE 0 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------------------------------- CHOICE 1 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------------------------------- CHOICE 2 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------------------------------- CHOICE 3 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------------------------------- CHOICE 4 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][7].

Answer: President
---------------------------------- CHOICE 5 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][7].

Answer: President
---------------------------------- CHOICE 6 ----------------------------------
.................................assistant.................................
```

Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 7 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 8 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 9 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 10 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 11 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][7].

Answer: President
---------- CHOICE 12 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 13 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 14 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 15 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------- CHOICE 16 ----------
.....assistant.....
Todd Boehly was the President of Guggenheim Partners [2][3][4][7].

```
Answer: President
---------------------------------- CHOICE 17 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------------------------------- CHOICE 18 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------------------------------- CHOICE 19 ----------------------------------
.................................assistant.................................
Todd Boehly was the President of Guggenheim Partners [2][3][4][6][7].

Answer: President
---------------------------------- ANSWER -----------------------------------
President
---------------------------------- CONFIDENCE ---------------------------------
1.0
```

D Automated Annotated Demonstrations

Following DSP (Khattab et al., 2022), a demonstration is defined as a training example crafted to illustrate particular behaviors expected from the LLM. A qualifying example of such a demonstration occurs when the model's prediction for this example aligns with the actual correct answer. We extend DSP's approach by incorporating additional considerations into the automated creation of demonstrations.

In the automated creation of demonstrations for use in the "Probe" and "Infer" procedures, we adjust citation marks using regular expressions. We employ the regular expression `(\[[0-9]+\])+` to identify citation marks and ensure they are placed at the end of each sentence or statement, if they are not already. To verify that all sentences or statements adhere to this format, we use the regular expression `^([^\[\.]+(\[[0-9]+\])*\.)+$`. This standardized format aids in accurately tallying the total count of cited passages.

For demonstrations intended for the "Plan" procedure, we select premium dependency rules utilizing regular expressions. The regular expression `None|((\s*([Ss]tep [0-9]+) depends on ([Ss]tep [0-9]+)\.\s*)+)` is used to ensure that dependencies in the dependency graph, generated by LLM, conform to a particular format. This assists in the precise identification of these relationships.

During our observations in automated annotated demonstrations for the "Plan" procedure, we have noticed that overly long sub-queries or steps produced by LLM often erroneously repeat the original, more complex question, deviating from the divide-and-conquer strategy of breaking down a complex question into smaller sub-queries. To address this, we implement the outlier detection method known as the interquartile range (IQR) to identify and disqualify any excessively long sub-query or step.

In selecting demonstrations for a prompt, we utilize two different approaches: balanced sampling and k-nearest neighbors (KNN). Balanced sampling involves randomly selecting from training examples while making sure to maintain an even distribution of answers (classes). KNN, on the other hand, makes use of sentence representations[4] to identify and select the k training examples closest to the input question (or claim, as in the case of FEVER). This approach was investigated by Liu et al. (2022).

[4]https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

E Baselines

Our benchmarking encompasses five methods: "Vanilla LM" as outlined by Brown et al. (2020), "Retrieve-then-Read" as discussed in the works of Lazaridou et al. (2022) and Izacard et al. (2022), "Self-ask" introduced by Press et al. (2022), "ReAct" described by Yao et al. (2023b), and "Demonstrate-Search-Predict" (DSP) presented by Khattab et al. (2022).

- Vanilla LM: The "Vanilla LM" baselines employ the few-shot in-context learning approach as proposed by Brown et al. (2020). These basic benchmarks don't engage in retrieving text passages pertinent to the input query.
- Retrieve-then-Read: The "Retrieve-then-Read" benchmarks utilize the retrieval model (RM) to support each instance with a possibly relevant text passage prior to presenting the prompt to the language model (LM).
- Self-ask: The "Self-ask" baselines involve the LM posing additional "follow-up questions" that are then directed to a retrieval model. Adhering to Khattab et al. (2022), we alter the Self-ask's prompt design by: (i) merging few-shot training instances from the task, such as question-answer pairs, at the beginning of the prompt, (ii) instructing the model to produce a brief initial answer at each retrieval phase, and (iii) specifically commanding the model to generate a subsequent "search query" at each stage.
- ReAct: The ReAct method utilizes LLMs to concurrently create reasoning traces and task-specific actions. We test ReAct using the "text-davinci-002" backbone LLM, focusing on the FEVER and HotPotQA datasets. However, the ReAct project has not incorporated the Open-SQuAD dataset and the "gpt-3.5-turbo-1106" backbone LLM, thus these have not been subjected to evaluation.
- Demonstrate-Search-Predict (DSP): The DSP method initiates pipeline-aware demonstrations, seeks out related passages, and creates predictions rooted in evidence. Following Khattab et al. (2022), we utilize random sampling to select and annotate examples, and then employ them as demonstrations.

F Extended Ablation Study

α	β	γ	w_1	w_2	w_3	**EM**	**F1**
0.1	0.45	0.45	0.15	0.55	0.3	25.16	36.55
0.1	0.45	0.45	0.2	0.55	0.25	27.04	39.34
0.1	0.45	0.45	0.3	0.5	0.2	24.53	35.20
0.1	0.45	0.45	0.3	0.6	0.1	25.16	35.35
0.1	0.45	0.45	1	0	0	22.64	34.15
0.2	0.4	0.4	0.15	0.55	0.3	25.16	36.55
0.2	0.4	0.4	0.2	0.55	0.25	31.45	42.17
0.2	0.4	0.4	0.3	0.5	0.2	27.67	41.44
0.2	0.4	0.4	0.3	0.6	0.1	25.16	35.40
0.2	0.4	0.4	1	0	0	23.90	35.27
0.3	0.35	0.35	0.15	0.55	0.3	23.90	37.03
0.3	0.35	0.35	0.2	0.55	0.25	25.79	36.78
0.3	0.35	0.35	0.3	0.5	0.2	28.30	40.67
0.3	0.35	0.35	0.3	0.6	0.1	25.16	37.23
0.3	0.35	0.35	1	0	0	26.42	38.00
0.4	0.3	0.3	0.15	0.55	0.3	25.16	38.50
0.4	0.3	0.3	0.2	0.55	0.25	25.79	38.37
0.4	0.3	0.3	0.3	0.5	0.2	27.67	41.06
0.4	0.3	0.3	0.3	0.6	0.1	25.79	38.58
0.4	0.3	0.3	1	0	0	23.27	35.46
1	0	0	0.15	0.55	0.3	27.04	39.47
1	0	0	0.2	0.55	0.25	28.30	38.12
1	0	0	0.3	0.5	0.2	24.53	37.02
1	0	0	0.3	0.6	0.1	26.42	35.89
1	0	0	1	0	0	24.53	37.76

Table 7: An elaborate overview of HGOT+KNN's various hyperparameter combinations being explored, along with their corresponding EM and F1 scores, within the medium-length category of the Open-SQuAD dataset.

α	β	γ	w_1	w_2	w_3	**EM**
0.1	0.45	0.45	0.15	0.55	0.3	53.33
0.1	0.45	0.45	0.2	0.55	0.25	54.00
0.1	0.45	0.45	0.3	0.5	0.2	57.33
0.1	0.45	0.45	0.3	0.6	0.1	54.67
0.1	0.45	0.45	1	0	0	61.33
0.2	0.4	0.4	0.15	0.55	0.3	51.33
0.2	0.4	0.4	0.2	0.55	0.25	56.67
0.2	0.4	0.4	0.3	0.5	0.2	52.00
0.2	0.4	0.4	0.3	0.6	0.1	59.33
0.2	0.4	0.4	1	0	0	57.33
0.3	0.35	0.35	0.15	0.55	0.3	57.33
0.3	0.35	0.35	0.2	0.55	0.25	57.33
0.3	0.35	0.35	0.3	0.5	0.2	61.33
0.3	0.35	0.35	0.3	0.6	0.1	56.67
0.3	0.35	0.35	1	0	0	61.33
0.4	0.3	0.3	0.15	0.55	0.3	59.33
0.4	0.3	0.3	0.2	0.55	0.25	56.67
0.4	0.3	0.3	0.3	0.5	0.2	60.00
0.4	0.3	0.3	0.3	0.6	0.1	56.67
0.4	0.3	0.3	1	0	0	60.67
1	0	0	0.15	0.55	0.3	58.00
1	0	0	0.2	0.55	0.25	58.00
1	0	0	0.3	0.5	0.2	54.67
1	0	0	0.3	0.6	0.1	52.67
1	0	0	1	0	0	58.00

Table 8: A detailed examination of the numerous hyperparameter configurations tested for HGOT+KNN, together with their respective EM scores, specifically within the medium-length category of the FEVER dataset.

α	β	γ	w_1	w_2	w_3	**EM**	**F1**
0.1	0.45	0.45	0.15	0.55	0.3	42.57	54.49
0.1	0.45	0.45	0.2	0.55	0.25	39.19	51.58
0.1	0.45	0.45	0.3	0.5	0.2	40.54	52.91
0.1	0.45	0.45	0.3	0.6	0.1	39.86	51.94
0.1	0.45	0.45	1	0	0	43.92	54.63
0.2	0.4	0.4	0.15	0.55	0.3	43.24	55.93
0.2	0.4	0.4	0.2	0.55	0.25	39.86	53.81
0.2	0.4	0.4	0.3	0.5	0.2	41.22	53.63
0.2	0.4	0.4	0.3	0.6	0.1	40.54	52.39
0.2	0.4	0.4	1	0	0	43.92	54.63
0.3	0.35	0.35	0.15	0.55	0.3	41.89	54.58
0.3	0.35	0.35	0.2	0.55	0.25	39.86	53.25
0.3	0.35	0.35	0.3	0.5	0.2	41.22	54.17
0.3	0.35	0.35	0.3	0.6	0.1	40.54	52.17
0.3	0.35	0.35	1	0	0	43.92	54.63
0.4	0.3	0.3	0.15	0.55	0.3	41.89	54.58
0.4	0.3	0.3	0.2	0.55	0.25	38.51	52.35
0.4	0.3	0.3	0.3	0.5	0.2	41.22	53.95
0.4	0.3	0.3	0.3	0.6	0.1	40.54	52.79
0.4	0.3	0.3	1	0	0	43.92	54.63
1	0	0	0.15	0.55	0.3	40.54	54.20
1	0	0	0.2	0.55	0.25	39.86	53.47
1	0	0	0.3	0.5	0.2	40.54	52.98
1	0	0	0.3	0.6	0.1	39.86	53.02
1	0	0	1	0	0	43.92	55.08

Table 9: A comprehensive review of the different hyperparameter combinations tested on HGOT+KNN, including both their EM and F1 scores, within the medium-length category of the HotPotQA dataset.