Enhancing Post-Hoc Attributions in Long Document Comprehension via Coarse Grained Answer Decomposition

Pritika Ramu, Koustava Goswami, Apoorv Saxena, Balaji Vasan Srinivavsan
Adobe Research, India
{pramu, koustavag, apoorvs, balsrini}@adobe.com

Abstract

Accurately attributing answer text to its source document is crucial for developing a reliable question-answering system. However, attribution for long documents remains largely unexplored. Post-hoc attribution systems are designed to map answer text back to the source document, yet the granularity of this mapping has not been addressed. Furthermore, a critical question arises: What exactly should be attributed? This involves identifying the specific information units within an answer that require grounding. In this paper, we propose and investigate a novel approach to the factual decomposition of generated answers for attribution, employing template-based in-context learning. To accomplish this, we utilize the question and integrate negative sampling during few-shot in-context learning for decomposition. This approach enhances the semantic understanding of both abstractive and extractive answers. We examine the impact of answer decomposition by providing a thorough examination of various attribution approaches, ranging from retrieval-based techniques to LLM-based attributors.

1 Introduction

The rise of Large Language Models (LLMs) and GenAI-based technologies has greatly increased their usability across various sectors, notably in grounded question-answering systems. However, to establish trust, it's crucial to attribute information obtained from source documents, especially given the tendency of these models to generate texts from their own knowledge bases (Huang et al., 2023). With opaque LLMs like ChatGPT, there's a need to explore post-hoc attribution methods to enhance reliability. Targeted attribution is necessary in chat-based question-answering systems to improve user experience, as not every part of an answer requires attribution.

Prior research addresses generating attributions alongside responses in open domains (Gao et al.,

Figure 1: An example from Verifiability dataset. The input to the post-hoc attribution system is the question, document and answer. The output is evidence sentences from the document. Text marked in red do not require attribution.

2023b,a), either per sentence or per paragraph (Bohnet et al., 2023). However, attributing responses in long document sequences is challenging, and inline attribution falls short (Gao et al., 2023b). Additionally, determining *when to cite* is crucial, as inappropriate or excessive citations can lead to redundancy (Huang and Chang, 2024). Moreover, excessive in-line citations can diminish creativity in generated content (Huang and Chang, 2024). (Guo et al., 2022) explores fine-tuning to enhance attributions, but these models are domain-specific, limiting their adaptability.

With rise of LLMs, the attribution task is viewed as a retrieval and mapping task (Li et al., 2022). This shift presents challenges, especially in abstractive question-answering scenarios, where retrieving context is not straightforward, and semantic relatedness must often be inferred from the entire document context. Notably, (Gao et al., 2023a) has underscored the limitations of retriever-based attribution engines, particularly in handling out-of-distribution knowledge during context processing.

Despite these efforts, existing methods treat an-

swers as singular attributable elements and strive to map them back to long sequence contexts. However, answers can encompass multiple facts and are contextually dependent on the posed question. Existing fine-tuning or retriever-based systems face challenges in identifying the specific attributable components within an answer. In this paper, we formulate post-hoc attribution as the task of identifying source sentences from a document that support *attributable* parts of an answer when our input consists of a question, answer, and the document from which the answer is obtained.

Firstly, we propose using template-based in-context learning as a method to achieve question-contextualized decomposition of answers. This technique helps break down answers that aligns with the question, which aids identifying references in the document context. This alignment is crucial as it enhances the identification of the specific information units necessary to answer the question effectively. Empirical observations suggest that guiding a language model for specific tasks in few-shot settings can lead to improved adaptability (Ma et al., 2023). Building upon this insight, we recognize the significance of guiding a language model for in-context decomposition. Additionally, we introduce *negative sampling*-based in-context learning to enable language models to discern between good and bad decomposition.

Secondly, we analyze the use of different retrieval methods as attributors and compare their performance using four decomposition methods. We observe that, on average, question-contextualized coarse-grain decomposition results are comparable, and in some settings, better than using non-decomposed sentences. This serves as a regression test for our methodology without degrading retriever performance. Moreover, our experimental results on the Citation Verifiability dataset highlight that contextualized coarse-grain decomposition for retriever-based attributors (BM25, GTR, MonoT5) achieves, on average, a **3%** gain in precision over baseline models, emphasizing the efficacy of decomposition for attributions (Table 1).

Thirdly, we examine the use of LLMs as post-hoc attributors and the advantages they offer by considering the context of the question and the decomposed answer. We observe a significant improvement in performance both empirically and qualitatively when providing question-contextualized coarse-grain decomposition of answers to LLMs.

By leveraging LLMs as attributors and incorporating question-contextualized coarse-grain decomposition, we achieve state-of-the-art performance on the QASPER and Verifiability datasets (Table 2).

2 Related Work

Multiple works have established the need for ensuring a trustworthy model and how citations and attributions can help ensure that reliable information is provided or to detect if any information is hallucinated (Huang and Chang, 2023; Litschko et al., 2023; Rashkin et al., 2022; Venkit et al., 2024). Efforts to create truthful AI have been highlighted, emphasizing the importance of fact-checking and claim verification, even in the pre-LLM era (Evans et al., 2021). Notably, (Petroni et al., 2022) focuses on improving the identification of claims from Wikipedia that lack support from citations, with a specific emphasis on identifying supporting evidence at a paragraph level. In addition to unimodal systems, multimodal systems, with the introduction of generative AI, are becoming increasingly important to attribute contextual parts and build trustworthy AI systems (Ancona et al., 2017; Holzinger et al., 2021; Zhao et al., 2023; Phukan et al., 2024).

In the context of dialogue systems requiring background knowledge, issues with spurious correlations have been acknowledged, necessitating a more robust method for identifying what can be attributed in an abstractive setting (Dziri et al., 2022). (Mei et al., 2023) introduces the identification of missing information and providing appropriate attributions to mitigate potential dangers due to misinformation. Meanwhile, (Sarti et al., 2023a,b) delve into the analysis of feature attribution and interpretability of language models. A unique task is undertaken by (Funkquist et al., 2023), which addresses the creation of inline citations for scientific articles from abstracts of other scientific articles.

(Sancheti et al., 2024) introduced fine-grained attribution of answers but did not cover fact-based attribution. The task of fact-based attribution—assigning facts from answers to a list of evidence sentences rather than paragraphs—is underexplored in the existing literature. This underscores the need for further exploration in fact-based attribution, particularly concerning sentence-level source attributions and the challenges posed by long document sequences in question-answering systems.
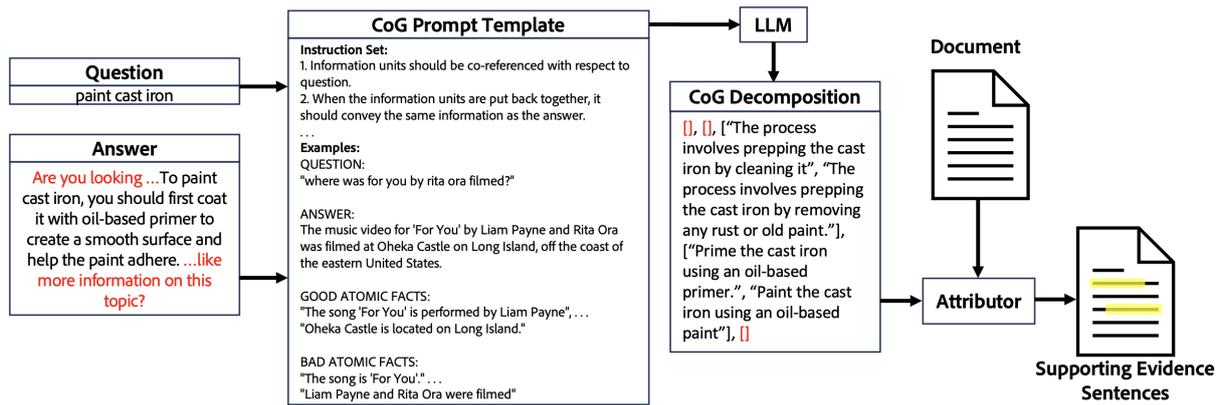
Figure 2: Pipeline for attribution: Answers are decomposed and sent to the attributor for identifying evidences.

3 Method

We propose a 2-fold process: First, decompose the answer into smaller information units, then employ an attributor to map these units to evidence from the source document (Fig. 2). We investigate using retrieval methods and various LLMs for attribution.

3.1 Task Definition

Given a question Q, a collection of evidence sentences $E = e_1, ..., e_n$ extracted from document D, and an answer A to question Q, the objective is to identify supporting evidence sentences (attributions) $e_i \in E$ for each answer part $a_i \in A = a_1, ..., a_m$. Each a_i may have multiple evidence sentences associated with it. An answer part is defined as a single complete sentence from an answer.

3.2 Answer Decomposition

To link answer components to a given document, we decompose the answer parts into smaller units known as information units.

3.2.1 Definition

Answers to a question often extend over considerable lengths, encapsulating diverse facts and information. Within these responses, the veracity of facts can vary, presenting a mix of true and false statements. When tasked with extracting evidence from a document in response to a question, the determination of factual accuracy relies on whether a given fact is substantiated by at least one supporting evidence from the source document.

The granularity at which we deconstruct an answer can vary, spanning from fine-grained to coarse-grained representations. While previous studies have investigated the use of a sentence from an answer as an information unit, it is important

to note that some sentences may be intricate, containing multiple conjunctions of information. This complexity underscores the need for a nuanced approach to information unit selection in order to capture what is to be attributed from an answer accurately. Identifying which information from the answer requires grounding is an important task to prevent grounding unnecessary sentences.

For answer part $a_i \in A$, we can decompose a_i as $iu_1, ..., iu_n$ such that information$(iu_i) \subseteq$ information(a_i).

3.2.2 Revisiting Fine Grained Decomposition

FActScore (Min et al., 2023) defines an atomic fact as a short sentence that conveys a single piece of information. This method assesses the factual precision of long texts through fine-grained decomposition but does not consider what specifically requires attribution and decomposition. By decomposing one sentence at a time, it often overlooks contextual semantics and the context of the question. To address these limitations, we propose Coarse Grained Decomposition (CoG).

3.2.3 Coarse Grained Decomposition (CoG)

Determining the appropriate granularity for decomposing an answer is subjective. For tasks requiring evidence extraction from a source document, it is impractical to set a strict upper limit on the number of facts per information unit. To effectively guide this process, we propose a question-contextualized decomposition approach. By developing a question-aware prompt template, we direct the LLM to generate information units that align closely with the specific context and requirements of the question.

We design the prompt template for coarse

grained decomposition by giving instructions on how an answer should be broken down and how not to be broken down. LLMs are known to perform poor on negation tasks (Truong et al., 2023). So instead of prompting the model to not generate a poor information unit, we provide a template to generate good and bad information units enabling us to parse out the required good information units. We provide the question and entire answer to the decomposer to obtain a more contextualized decomposition. See Appendix A for complete prompt.

Instruction Schema. The instruction schema is a generalizable schema that follows (Wang et al., 2022). The **Task Definition** defines a given task and lists the input and output formats for the question, answer, and decomposition. The **Instruction Set** guides the language model to generate useful decompositions. We explain what good and bad instruction units are. Good information units are relevant, meaningful, and are directly associated with the question. They are broken down at logical conjunctions and, when reassembled, convey the same meaning as the original answer. Conversely, bad information units include redundant content, non-statements, or irrelevant information. **Positive Samples** illustrate the expected decomposition for a given question, aligning with the provided instructions. We include examples covering all instructions in the set. **Negative Samples** demonstrate negation instructions to the LLM, which typically struggles with negation tasks. These samples align with negation instructions.

3.3 Classifier

The classifier aids the attribution system's efficiency by deciding whether a sentence requires decomposition. This benefits in cost and latency by identifying and excluding simplistic sentences from decomposition. For instance, the sentence "Alex is an engineer." is already a simple sentence containing a single fact and therefore does not require decomposition.

To systematically identify such cases, we have implemented a rule-based classifier that evaluates sentences based on their linguistic characteristics, specifically their part-of-speech (POS) tags, as described by Toutanova et al. (Toutanova et al., 2003). Let S be the set of POS tags obtained from a sentence, with N, P, V, and A representing Noun, Pronoun, Verb, and Article respectively. A sentence is deemed to not require decomposition if it

satisfies the following constraint (C_l):

$$C_i = (|S \cap \{N \cup P \cup A\}| = |S| - 1)$$
$$C_j = (S - \{N \cup P \cup A\} \in V)$$
$$C_k = (S \subseteq \{N \cup P \cup A\})$$
$$C_l = (C_i \wedge C_j) \vee C_k$$

This constraint implies that if a sentence consists solely of any combination of nouns, pronouns, or articles with at most one verb, it is classified as a simple sentence and is not decomposed[1]. This classification rule acts as a high-precision filter, targeting sentences that are single independent clauses. These clauses are straightforward and typically do not benefit from further decomposition.

The design of this rule was based on linguistic rules and empirical observations to ensure high precision. While this rule is optimized for high precision, it is possible that simple sentences are not identified as simple sentences. In such instances, the integrity of the decomposition process is not compromised since the output remains effectively unchanged when processed by the decomposer.

3.4 Attributors

We evaluate two types of attributors for extracting evidence from documents: retrievers and LLMs. Retrievers rank evidence sentences from the document, while LLMs select the most appropriate evidences for each information unit.

3.4.1 Retrievers

We investigate various retrieval methods that serve as attributors by ranking evidence sentences within a document. Each information unit is treated as a query to retrieve evidence. For answer parts containing multiple information units, our goal is to select the most relevant evidence set. To optimize this selection, we use a greedy merging strategy.

Alogrithm 1 shows a greedy merging algorithm on how the evidences are chosen for an answer part. Let IU be the list of information units for an answer part, E the list of evidences from the document and L the final ordered list of evidences. score(iu, e) refers to the score obtained from the retriever when iu is the query and e is a single evidence from the document. We rank the list of evidences based on score(iu, e). If two information units have a high score for the same evidence

---

[1]This constraint does not hold true if punctuation other than full stop (.) or quotes (''') exist.

sentence, we include the evidence only once. The intuition to using a greedy algorithm is to is to obtain only top evidences for an answer part. If an information unit has a low score with the evidences, it means that information is not attributable to the document.

Algorithm 1 Merging of Evidences for Answer Part

$IU = [iu_1, iu_2, \ldots, iu_n]$
$E = [[e_{11}, \ldots, e_{1m}], \ldots, [e_{n1}, \ldots, e_{nm}]]$
Initialize an empty list L
for each iu in IU **do**
    Initialize a variable max_score $\leftarrow -\infty$
    Initialize a variable best_evidence $\leftarrow$ null
    for each e in $E[iu]$ **do**
        if score(iu, e) > max_score and e not in L **then**
            max_score $\leftarrow$ score(iu, e)
            best_evidence $\leftarrow e$
        end if
    end for
    if best_evidence $\neq$ null **then**
        $L \leftarrow$ (max_score, best_evidence)
    end if
end for
Sort L in descending order based on the score

3.4.2 Large Language Models

Prior works have shown the ability of LLMs to generate text with citations, which reduces the hallucinations (Gao et al., 2023b). We explore LLMs as attributors when the task is broken down into finding evidence for smaller sentences. We ask the LLM to find evidences that support information units for a given question.

Instruction Schema. To guide the LLM in finding attribution for a question and its corresponding information unit, the instruction schema for using LLM as an attributor comprises two main components. The **Task Definition** defines the task of attributing an information unit from a list of retrieved sentences. The **Instruction Set** instructs the LLM to identify and select a valid list of evidence and sort such that the most relevant evidence appears first. For complete prompt, see Appendix B.

4 Datasets

We reformulate the Citation Verifiability dataset (Liu et al., 2023) and QASPER dataset (Dasigi et al., 2021) for our task. The statistics of the dataset are present in Table 6 (Appendix C).

4.1 Citation Verifiability Dataset

The Citation Verifiability dataset (Liu et al., 2023) consists of questions from NaturalQues-

tions (Kwiatkowski et al., 2019) and ELI5 (Fan et al., 2019). Answers are generated from search engines such as Bing Chat and NeevaAI. The answers contain inline citations pointing to web pages. Human annotators judged citations as *fully*, *partially*, or *not* supporting sentences. For fully supported sentences, annotators provided supporting sentences from cited web pages. We consider these supporting sentences as attributions for an answer part. This dataset provides evidences on a sentence level and provides citations only for the informative parts of the answer.

4.2 QASPER Dataset

QASPER (Dasigi et al., 2021) is a dataset for information-seeking questions and answers grounded in research papers. Each question is written by an NLP practitioner who only read the title and abstract of the corresponding paper, and the question seeks information present in the full text. The questions are then answered by a separate set of NLP practitioners who also provide supporting evidence for their answers. The answers in the dataset can be unanswerable, extractive (spans in the paper serving as the answer), or free-form. We take the evidences marked as supporting evidence (evidences with high agreement amongst annotators) as the ground truth attribution. This dataset does not provide the evidence on a sentence level, but on an answer level.

5 Experiments and Evaluation

5.1 Baselines and Evaluation Strategy

We evaluate the performance of three retrieval-based systems and four large language models (LLMs) as attributors. The retrieval methods include: BM25, a sparse model; GTR, a dense model (Ni et al., 2022); and MonoT5 (Pradeep et al., 2021). We explore the capabilities of four LLMs: GPT-4 (`gpt-4`), GPT-3.5 (`gpt-3.5-turbo`), LLaMa 2 70B (`llama-2-70b-chat-hf`), and LLaMa 2 13B (`llama-2-13b-chat-hf`) (Touvron et al., 2023). We use GPT-4 (`gpt-4`) for obtaining decomposition. We use NVIDIA A100 GPU to run inference for LLaMa 2 models. We do not provide retrieve-and-read-based baselines (Guu et al., 2020; Borgeaud et al., 2022; Izacard et al., 2022) as they generate answers along with attributions, whereas our task assumes answer as an input. Table 1 and Table 2 tabulates the results for retrieval based and LLM based attributors re-

Decomposer+Attributor	Verifiability			QASPER		
	Top 1 P/R/F1	Top 2 P/R/F1	Top 4 P/R/F1	Top 1 P/R/F1	Top 2 P/R/F1	Top 4 P/R/F1
NIL+BM25	0.66/**0.53**/0.59	0.44/**0.65**/**0.53**	0.27/**0.72**/0.39	**0.42**/**0.18**/**0.26**	**0.32**/**0.25**/**0.28**	**0.22**/**0.31**/**0.25**
FActScore+BM25	0.31/0.24/0.27	0.24/0.34/0.28	0.18/0.47/0.26	0.24/0.09/0.13	0.18/0.13/0.15	0.11/0.16/0.13
(CoG - neg.)+BM25	0.67/0.51/0.59	0.45/0.61/0.52	0.28/0.69/0.40	0.35/0.17/0.22	0.30/0.22/0.25	0.21/0.29/0.23
CoG+BM25	**0.69**/0.52/**0.60**	**0.46**/0.62/**0.53**	**0.29**/0.70/**0.41**	0.37/0.17/0.23	0.30/0.23/0.26	**0.22**/0.30/0.24
NIL+GTR	0.66/**0.51**/0.57	0.43/**0.62**/**0.51**	0.27/**0.72**/0.39	**0.41**/**0.18**/**0.25**	**0.31**/0.24/**0.27**	**0.22**/**0.31**/**0.26**
FActScore+GTR	0.57/0.44/0.50	0.38/0.56/0.45	0.24/0.66/0.35	0.28/0.10/0.15	0.21/0.15/0.18	0.15/0.22/0.18
(CoG-neg.)+GTR	0.67/0.50/0.57	0.44/0.58/0.50	0.27/0.70/0.40	0.37/0.16/0.22	0.29/0.21/0.25	0.21/0.30/0.25
CoG+GTR	**0.69**/**0.51**/**0.59**	**0.45**/0.59/**0.51**	**0.29**/0.71/**0.42**	0.39/0.17/0.24	0.30/0.23/0.26	**0.22**/**0.31**/**0.26**
NIL+MonoT5	0.70/**0.54**/0.61	0.47/0.68/0.55	0.28/**0.76**/0.41	**0.47**/**0.21**/**0.29**	0.34/**0.28**/**0.31**	0.24/0.35/0.28
FActScore+MonoT5	0.62/0.50/0.55	0.41/0.61/0.49	0.24/0.66/0.36	0.39/0.16/0.23	0.26/0.21/0.24	0.18/0.26/0.21
(CoG – neg.)+MonoT5	0.70/0.53/0.60	0.46/0.67/0.54	0.28/0.74/0.41	0.45/0.20/0.27	0.33/0.27/0.30	0.24/0.34/0.28
CoG+MonoT5	**0.72**/0.54/**0.62**	**0.49**/0.66/**0.56**	**0.30**/0.75/**0.43**	0.46/0.20/0.28	**0.35**/**0.28**/**0.31**	**0.25**/**0.35**/**0.29**

Table 1: Retrieval based attributor results

Decomposer+Attributor	Verifiability	QASPER
	P/R/F1	P/R/F1
NIL+GPT 4	0.26/0.76/0.39	0.18/0.15/0.16
FActScore+GPT 4	0.18/0.69/0.29	0.17/0.27/0.21
(CoG-neg.)+GPT 4	0.26/0.76/0.39	0.19/0.29/0.23
CoG+GPT 4	**0.29**/**0.79**/**0.42**	**0.22**/**0.32**/**0.26**
NIL+GPT 3.5	0.25/0.75/0.37	0.16/0.15/0.16
FActScore+GPT 3.5	0.17/0.69/0.27	0.15/0.27/0.20
(CoG – neg.) + GPT 3.5	0.27/0.75/0.40	0.18/0.27/0.22
CoG+GPT 3.5	**0.28**/**0.77**/**0.41**	**0.22**/**0.31**/**0.26**
NIL+LLaMa 2 (70 B)	0.20/0.73/0.32	0.13/0.10/0.11
FActScore+LLaMa 2 (70 B)	0.15/0.66/0.25	0.12/0.25/0.16
(CoG-neg.) + LLaMa 2 (70 B)	0.22/0.75/0.34	0.17/0.26/0.21
CoG+LLaMa 2 (70 B)	**0.23**/**0.77**/**0.36**	**0.20**/**0.28**/**0.23**
NIL+LLaMa 2 (13 B)	0.19/0.72/0.30	0.11/0.09/0.10
FActScore+LLaMa 2 (13 B)	0.13/0.63/0.22	0.10/0.24/0.14
(CoG-neg.) + LLaMa 2 (13 B)	0.21/0.73/0.33	0.16/0.25/0.20
CoG+LLaMa 2 (13 B)	**0.22**/**0.74**/**0.34**	**0.18**/**0.25**/**0.21**

Table 2: LLM based attribution results

spectively. NIL refers to using an answer sentence as an information unit, FActScore refers to using fine grained decomposition as information units and CoG refers to question contextualised coarse grain decomposition as information units. For all the LLM based attributions, we take the top 100 sentences retrieved from BM25 to fit within the context limit. We keep the retrieved evidences same across all settings for a fair evaluation.

5.2 Evaluation Measures

For the retriever based attributors, since we get a score for a query and evidence, we report precision (P), recall (R), and F1 scores of top 1, 2, and 4 predicted attributions per sentence of an answer. For the LLM based attributors, we ask the LLM to output only highly relevant evidences. Since there is no score between the query (information unit) and evidence, we collect all the shortlisted evidences and report the precision, recall and F1 scores. To ensure a reliable evaluation, we do not consider the samples where answer sentences are an exact match from from the documents. For Verifiability dataset, we report attributions on sentence level for each sentence in the answer and for QASPER we report attributions on sentence level for the whole answer.

6 Results and Discussion

In Table 1, while using CoG as the decomposer for retrieval based methods, we observe increased precision and overall F1 score on the Citation Verifiability dataset. The observed increase in precision on the Verifiability dataset is because CoG decomposer does not provide information units for the sentences that do not require attribution.

We notice comparable results over the QASPER dataset. This dataset majorly contains extractive and short sentences. The average length of extractive answers in the dataset is 14.4 and that of abstractive answers in 15.6 words. Due to the short and highly extractive nature of the dataset, it is likely that using answer sentences as information unit suffices.

Interestingly, we observe that on QASPER, GTR and MonoT5 have comparable performance while using CoG and single answer sentence as an information unit. Whereas, in BM25 there is a slight reduction in scores. This is likely due to the way the answer is decomposed or *augmented* when the question is taken into context. This impacts retrieval methods like BM25 where the frequency of words are taken into account to calculate the score. This motivates the use of embedding based retrievers as attributors over term frequency based retrievers.

In Table 2, when utilizing CoG as the decomposer, we observe improvements across all LLMs.

Question	"paint cast iron"				
Answer	"To paint cast iron, you should first coat it with oil-based primer to create a smooth surface and help the paint adhere. You can find cast iron paint on Amazon."				
GT Attribution	CoG Decompositions	NIL-MonoT5	CoG-MonoT5	NIL-GPT4	CoG-GPT4
["Coat the cast iron with oil-based primer.', "Priming the metal creates a smooth surface and will help the paint adhere."]	["To paint cast iron, you should first coat it with oil-based primer.", "The oil-based primer helps create a smooth surface and help the paint adhere."]	["If you're working with a smaller piece of cast iron, you can wipe it down with a damp rag, instead.", "Apply oil-based paint to the cast iron."]	["Coat the cast iron with oil-based primer.", "If you're working with a smaller piece of cast iron, you can wipe it down with a damp rag, instead."]	["Priming the metal creates a smooth surface and will help the paint adhere.", "Apply oil-based paint to the cast iron.", "Coat the cast iron with oil-based primer."]	["Coat the cast iron with oil-based primer.", "Priming the metal creates a smooth surface and will help the paint adhere."]
[]	["You can find cast iron paint on Amazon."]	["Read on for our complete guide to painting cast iron easily at home."]	["Read on for our complete guide to painting cast iron easily at home."]	[]	[]

Table 3: Qualitative example of how decomposition affects retrieval based attributor and LLM based attributor. GT refers to ground truth. Each row depicts an answer part and respective decompositions and attributions for each method.

(Asher et al., 2023) demonstrated that LLMs operate without formal guarantees for tasks requiring entailments and in-depth language comprehension. Prompting techniques such as Chain-of-Thought (CoT) (Wei et al., 2023) are well-known to enhance the performance of LLMs. The improved performance with the use of CoG proves that providing decomposed answers in the form of information units likely simplifies the task of finding evidence that entails the specified information unit.

FActScore based decomposition performs poorly across all settings. The fine grain decompositions do not capture the information that is required to be attributed. Table 4 shows the large number of decompositions obtained for an answer. Intuitively, having a large number of decompositions should result in higher recall and lower precision. Yet, this trend is not observed. This indicates that the information that the answer part conveys gets diluted when broken down into finer parts. This impacts the retriever and LLM based attributors to perform well. We also observe that CoG without negative sampling performs slightly poor than with negative sampling in all cases.

We observe that identifying relevant information helps MonoT5 make better attributions compared to using answer sentences as queries. Retrieval-based attributors rely on query and evidence scores, which can lead to irrelevant matches, especially since finding an appropriate threshold is challenging and dataset-specific. Unlike retrieval methods, LLMs consider question context, resulting in more accurate attributions. However, it is interesting to observe that when the information unit is given as a complete sentence, LLM struggles to find the precise set of evidences. Examples for these observations are present in Table 7 (Appendix D).

6.1 Ablation Study

6.1.1 Analysis of Decomposers

In the Citation Verifiability dataset, ground truth citations are available only for answer parts that have information relevant to the question. We assess sentence attributions using FActScore, CoG-Question (coarse grain decomposition without the question as context), and CoG as decomposers. In Verifiablility datset, **573** sentences **do not** require attributions. Using FActScore, we attribute **509** of them compared to **491** using CoG-Question and **473** using CoG decomposition. Table 4 provides a qualitative example decomposition obtained. We observe that fine-grained attribution may lead to information duplication across multiple units, resulting in a higher number of decomposition per sentence, as shown in Figure 3.

Human evaluation. We conduct a human survey to validate the quality of decomposition. The objective is to understand what specific facts within an answer are essential and should be credited as factual references in a document. We ask 3 annotators of similar backgrounds (Indian origin, above undergraduate studies, fluent in English). They are provided with 120 examples each, along with question, answer, FActScore and CoG decomposition. We provide the instruction set that is given to the LLM, so humans can validate whether the decomposition adhere to the requirements.

In terms of alignment of answer decomposition for the task of attribution, our outputs are marked better than the baseline in 80% cases. The inter-

Question	paint cast iron
Answer	*Are you looking for information on how to paint cast iron? If so, I found a helpful article on wikiHow that provides a step-by-step guide on how to paint cast iron.* The process involves prepping the cast iron by cleaning it and removing any rust or old paint. Then, you can prime and paint the cast iron using an oil-based primer and paint. *Would you like more information on this topic?*
FActScore	['The person is looking for information.', 'The person is looking for information on how to paint cast iron.'], ['There is a helpful article on wikiHow.','The article provides a step-by-step guide.', 'The article is about how to paint cast iron.'], ['The process involves prepping cast iron.', 'Prepping cast iron involves cleaning it.', 'Prepping cast iron involves removing rust.', 'Prepping cast iron involves removing old paint.'], ['You can prime with an oil-based primer.', 'You can paint with an oil-based paint.', 'You can prime and paint a cast iron.'], ['This is a question.', 'The topic referenced is unclear.]
CoG - Question	[], ['There is a helpful article on wikiHow that provides a step-by-step guide on how to paint cast iron.'], ['The process involves prepping the cast iron by cleaning it', 'The process involves prepping the cast iron by removing any rust or old paint.'], ['Prime the cast iron using an oil-based primer.', 'Paint the cast iron using an oil-based paint'], []
CoG	[], [], ['The process involves prepping the cast iron by cleaning it', 'The process involves prepping the cast iron by removing any rust or old paint.'], ['Prime the cast iron using an oil-based primer.', 'Paint the cast iron using an oil-based paint'], []

Table 4: Example from Citation Verifiability dataset: In the answer, portions highlighted in red do not need attributions. Lists show the decomposition outputs for each answer part. CoG - Question denotes coarse-grain decomposition without the question in context.

Evidence	NIL+MonoT5	CoG+MonoT5
"WNUT16: WNUT16 was a shared task on Named Entity Recognition over Twitter BIBREF10."	-0.043	-0.021

Table 5: Example of retriever score getting affected while using answer part as *iu* vs using decomposed *iu*.

Figure 3: Average number of decomposition per sentence using each method.

annotator agreement (Krippendorff, 1970) agreement for is 0.68, indicating a strong agreement among annotators. The details of the survey are provided in Appendix F.

Classifier. We check the number of answer sentences that are returned without decomposition on the dev set of QASPER and Verifiability dataset. The classifier correctly identifies 25 out of 158 answer sentences (17%) on QASPER and 22 out of 184 answer sentences in Verifiability (12%) as sentences that do not require any decomposition. The classifier does not classify any answer sentence that undergoes decomposition as *a simple sentence*.

6.2 Effect of Decomposition on Retrievers

Complex answer sentences in retrieval systems can impact the scores obtained for evidence sentences. For instance, in Table 5, the answer comprising multiple facts mapped to different evidence may result in lower scores for the correct evidence compared to using decomposed information units as search queries.

6.2.1 Effect of Decomposition on LLMs

Table 2 shows improved performance when CoG decomposition are used to guide the LLM. We observe that while attributions obtained for a complex

answer (without decomposition) aligns with the answer on a high level, it does not provide concrete support to the information present in the answer. Whereas, in CoG+GPT4, we notice a more nuanced evidence retrieval because the task of searching information is broken down into smaller pieces. Examples for these observation are present in Table 10 (Appendix E).

7 Conclusion

In this paper, we raise research questions regarding the consideration of post-hoc attributions as a grounding task rather than a naive mapping task. On this basis, we propose and explore a novel approach to the factual decomposition of generated answers for attribution, employing template-based instructional tuning. We empirically establish the fact that our proposed granular-level Coarse Grained Decomposition (CoG) helps identify the spans of answers that need decomposition by following the semantic context inferred from the question asked. We also qualitatively and empirically establish that using Language Models (LLMs) as attributors provides the breathing space to consider attribution as contextual semantic grounding rather than performing as a retrieval and mapping task. Through various ablation studies, we establish that on an extractive dataset, retriever-based algorithms can perform better by incorporating our Coarse

Grained Decomposition as input.

8 Limitations

We present promising results in enhancing post-hoc attributions for comprehending lengthy documents. Adopting a post-hoc perspective provides research opportunities when dealing with outputs from opaque models. Currently, our focus is solely on post-hoc attributions through a unimodal lens, involving textual input and textual output. The transition to multimodal attributions would present additional challenges. An intriguing avenue for research involves exploring attributions from tables, charts, and images and performing reasoning over them.

In our post-hoc approach, we identify only those parts with supporting evidence but do not address how to mitigate unsupported claims. This creates opportunities to explore the incorporation of feedback loops for unattributable answer parts, aiming to generate more reliable answers.

References

Marco Ancona, Enea Ceolini, A. Cengiz Öztireli, and Markus H. Gross. 2017. A unified view of gradient-based attribution methods for deep neural networks. *CoRR*, abs/1711.06104.

Nicholas Asher, Swarnadeep Bhar, Akshay Chaturvedi, Julie Hunter, and Soumya Paul. 2023. Limits for learning with language models.

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

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. 2022. Improving language models by retrieving from trillions of tokens.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. 2021. A dataset of information-seeking questions and answers anchored in research papers.

Nouha Dziri, Hannah Rashkin, Tal Linzen, and David Reitter. 2022. Evaluating attribution in dialogue systems: The BEGIN benchmark. *Transactions of the Association for Computational Linguistics*, 10:1066–1083.

Owain Evans, Owen Cotton-Barratt, Lukas Finnveden, Adam Bales, Avital Balwit, Peter Wills, Luca Righetti, and William Saunders. 2021. Truthful ai: Developing and governing ai that does not lie.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

Martin Funkquist, Ilia Kuznetsov, Yufang Hou, and Iryna Gurevych. 2023. Citebench: A benchmark for scientific citation text generation.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, et al. 2023a. Rarr: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023b. Enabling large language models to generate text with citations.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training.

Andreas Holzinger, Bernd Malle, Anna Saranti, and Bastian Pfeifer. 2021. Towards multi-modal causability with graph neural networks enabling information fusion for explainable AI. *Inf. Fusion*, 71:28–37.

Jie Huang and Kevin Chang. 2024. Citation: A key to building responsible and accountable large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 464–473, Mexico City, Mexico. Association for Computational Linguistics.

Jie Huang and Kevin Chen-Chuan Chang. 2023. Citation: A key to building responsible and accountable large language models.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

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

Klaus Krippendorff. 1970. Estimating the reliability, systematic error and random error of interval data. *Educational and Psychological Measurement*, 30(1):61–70.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Minghan Li, Xueguang Ma, and Jimmy Lin. 2022. An encoder attribution analysis for dense passage retriever in open-domain question answering. In *Proceedings of the 2nd Workshop on Trustworthy Natural Language Processing (TrustNLP 2022)*, pages 1–11, Seattle, U.S.A. Association for Computational Linguistics.

Robert Litschko, Max Müller-Eberstein, Rob van der Goot, Leon Weber, and Barbara Plank. 2023. Establishing trustworthiness: Rethinking tasks and model evaluation.

Nelson F. Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating verifiability in generative search engines.

Huan Ma, Changqing Zhang, Yatao Bian, Lemao Liu, Zhirui Zhang, Peilin Zhao, Shu Zhang, Huazhu Fu, Qinghua Hu, and Bingzhe Wu. 2023. Fairness-guided few-shot prompting for large language models. *CoRR*, abs/2303.13217.

Alex Mei, Sharon Levy, and William Yang Wang. 2023. Foveate, attribute, and rationalize: Towards physically safe and trustworthy ai.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, and Yinfei Yang. 2022. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Fabio Petroni, Samuel Broscheit, Aleksandra Piktus, Patrick Lewis, Gautier Izacard, Lucas Hosseini, Jane Dwivedi-Yu, Maria Lomeli, Timo Schick, Pierre-Emmanuel Mazaré, Armand Joulin, Edouard Grave, and Sebastian Riedel. 2022. Improving wikipedia verifiability with ai.

Anirudh Phukan, Shwetha Somasundaram, Apoorv Saxena, Koustava Goswami, and Balaji Vasan Srinivasan. 2024. Peering into the mind of language models: An approach for attribution in contextual question answering. *arXiv preprint arXiv:2405.17980*.

Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. 2021. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. 2022. Measuring attribution in natural language generation models.

Abhilasha Sancheti, Koustava Goswami, and Balaji Srinivasan. 2024. Post-hoc answer attribution for grounded and trustworthy long document comprehension: Task, insights, and challenges. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 49–57, Mexico City, Mexico. Association for Computational Linguistics.

Gabriele Sarti, Grzegorz Chrupała, Malvina Nissim, and Arianna Bisazza. 2023a. Quantifying the plausibility of context reliance in neural machine translation.

Gabriele Sarti, Nils Feldhus, Ludwig Sickert, and Oskar van der Wal. 2023b. Inseq: An interpretability toolkit for sequence generation models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 421–435, Toronto, Canada. Association for Computational Linguistics.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 252–259.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Thinh Hung Truong, Timothy Baldwin, Karin Verspoor, and Trevor Cohn. 2023. Language models are not naysayers: an analysis of language models on negation benchmarks. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 101–114, Toronto, Canada. Association for Computational Linguistics.

Pranav Narayanan Venkit, Tatiana Chakravorti, Vipul Gupta, Heidi Biggs, Mukund Srinath, Koustava Goswami, Sarah Rajtmajer, and Shomir Wilson. 2024. " confidently nonsensical?": A critical survey on the perspectives and challenges of 'hallucinations' in nlp. *arXiv preprint arXiv:2404.07461*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models.

Ruochen Zhao, Hailin Chen, Weishi Wang, Fangkai Jiao, Do Xuan Long, Chengwei Qin, Bosheng Ding, Xiaobao Guo, Minzhi Li, Xingxuan Li, and Shafiq Joty. 2023. Retrieving multimodal information for augmented generation: A survey. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 4736–4756. Association for Computational Linguistics.

Appendix

A   Instruction Schema for CoG

You are a helpful assistant. You will be given a question and corresponding answer that is grounded to document. You need to break down the answer for a given question into information units. The answer is already split into sentences. Map each sentence from the answer to the corresponding information unit/ units. Give only those information units that are attributable to the grounded document.

Instruction on what good information units are:

1. Give information units that are relevant to the sentence.

2. Information units should be meaningful.

3. Break down information units at conjunctions.

4. Information units should be co-referenced with respect to question.

5. When the information units are put back together, it should convey the same information as the answer.

Instruction on what bad information units are:

1. Information units that convey duplicate information.

2. Information units that are non statements.

3. Information units that are not meaningful to the question.

4. Information units that repeat facts present in the answer for introduction, conclusion or summary of an answer.

Examples:
  QUESTION:

   "Where was 'For You' by Rita Ora filmed?"

  ANSWER:

   [1:{"The music video for 'For You' by Liam Payne and Rita Ora was filmed at Oheka Castle on Long Island, off the coast of the eastern United States."}]

  GOOD ATOMIC FACTS:
  {"The song 'For You' is performed by Liam Payne." : 1, "The song 'For You' is performed by Rita Ora." : 1, "The music video for 'For You' was filmed at Oheka Castle." : 1, "Oheka Castle is located on Long Island." : 1, "Long Island is off the coast of the eastern United States." : 1}
  BAD ATOMIC FACTS:
  {"The song is 'For You'." : 1, "Liam Payne and Rita Ora were filmed." : 1 }

B Instruction Schema for LLM Attributor

Given a question, information units relevant to the question and retrieved evidences, retrieve sentences from the evidence which support the information units. The sentences which support the information unit will be considered attributions to the information unit. The sentence should provide a reasoning to the information unit, question and answer.

Output a list of retrieved sentences. Output only a valid list and no other text. If no sentence is supported, return empty list []. Be precise in identifying sentences that support the information units by returning only highly relevant sentences. Return a list of length 0, 1 or 2. Do not return more. DO NOT PARAPHRASE THE SENTENCES FROM THE RETRIEVED EVIDENCES. OUTPUT EXACT SENTENCES IN THE LIST. Sort the returned list based on the relevance to the information unit. The highly relevant evidence should appear as the first element.

OUTPUT FORMAT: ["sentence 23", "sentence 34", "sentence 40"]

QUESTION: {question}
ANSWER: {answer}
INFORMATION UNITS: {iu}
EVIDENCES: {evidences}
Output a valid python list from now on
OUTPUT LIST:

C Dataset Statistics

Table 6 tabulates the statistics of the dataset used for evaluation.

Dataset	Verifiability	QASPER
No. of Questions	136	599
Avg. No. of source sentences	130.03	517.49
Avg. No. of sentences per answer	3.57	1.67
Avg. No. of words per answer	72.96	14.8
Avg. No. of attributions per answer sentence	1	NIL
Avg. No. of attributions per answer	2.13	3.65
Avg. No. of answers per question	2.75	1.47

Table 6: Dataset statistics reported on test sets.

D Qualitative Example to compare retriever and LLM

Table 7 shows an example of how question in context of attribution enables LLMs to perform better attributions.

E Qualitative Examples for Ablation Studies

Table 8 sets the ground by setting the context of the example for the qualitative ablation stuides in Table 9 and Table 10.

F Human Survey

Figure 4 displays the survey format and the instructions provided to human annotators. Each step in the template clearly delineates guidelines for identifying good and bad atomic facts. The form included a radio button for evaluators to select which decomposition they believe best matches between the FActScore methodology and our approach. Notably, our definition of good atomic facts considers the relevance of the decomposition to the posed question.

During this process, we encountered instances of human error in evaluation. Figure 5 illustrates a case where human annotators preferred the FActScore-based decomposition over the CoG decomposition. The most common reason for choosing FActScore over CoG was confusion among annotators about which information units were relevant to the question. For instance, Figure 5 relates to a question about the actor who starred in "O Brother, Where Art Thou." The answer includes introductory elements unrelated to the question, which ideally should not be attributed. The FActScore system fails to discriminate and attempts to generate facts for all sentences. In contrast, our system decomposes only those parts that are pertinent to both the question and the answer. This nuance was not captured during the human evaluation, leading to a preference for FActScore over our methodology, which is misleading. These instances significantly contributed to the error cases during evaluation.

Question	"paint cast iron"				
Answer	"To paint cast iron, you should first coat it with oil-based primer to create a smooth surface and help the paint adhere. You can find cast iron paint on Amazon."				
GT Attribution	CoG Decompositions	NIL-MonoT5	CoG-MonoT5	NIL-GPT4	CoG-GPT4
["Coat the cast iron with oil-based primer.', "Priming the metal creates a smooth surface and will help the paint adhere."]	["To paint cast iron, you should first coat it with oil-based primer.", "The oil-based primer helps create a smooth surface and help the paint adhere."]	["If you're working with a smaller piece of cast iron, you can wipe it down with a damp rag, instead.", "Apply oil-based paint to the cast iron."]	["Coat the cast iron with oil-based primer.", "If you're working with a smaller piece of cast iron, you can wipe it down with a damp rag, instead."]	["Priming the metal creates a smooth surface and will help the paint adhere.", "Apply oil-based paint to the cast iron.", "Coat the cast iron with oil-based primer."]	["Coat the cast iron with oil-based primer.", "Priming the metal creates a smooth surface and will help the paint adhere."]
[]	["You can find cast iron paint on Amazon."]	["Read on for our complete guide to painting cast iron easily at home."]	["Read on for our complete guide to painting cast iron easily at home."]	[]	[]

Table 7: Qualitative example of how decomposition affects retrieval based attributor and LLM based attributor. GT refers to ground truth. Each row depicts an answer part and respective decompositions and attributions for each method.

Question	"Which downstream tasks are used for evaluation in this paper?"
Answer	"Various sequence tagging tasks: Argument detection, ACE entity and event detection, part-of-speech tagging, CoNLL chunking, CoNLL named entity recognition, GENIA bio-entity recognition, WNUT named entity recognition. They also evaluate on Stanford Sentiment Treebank, Penn TreeBank constituency parsing, and Stanford Natural Language Inference."
CoG IUs	["Argument detection is used for evaluation in this paper.", "ACE entity and event detection is used for evaluation in this paper.", "CoNLL named entity recognition is used for evaluation", "CoNLL chunking is used for evaluation.", "WNUT named entity recognition is used for evaluation.", "Part-of-speech tagging is used for evaluation in this paper.", "GENIA bio-entity recognition is used for evaluation."]
GT Attributions	["GENIA NER: The Bio-Entity Recognition Task at JNLPBA BIBREF9 annotated Medline abstracts with information on bio-entities (like protein or DNA-names).", "POS: We use the part-of-speech tags from Universal Dependencies v. 1.3 for English with the provided data splits.", "We use the CoNLL 2003 NER model, the Stanford Sentiment Treebank (SST-5) model, the constituency parsing model for the Penn TreeBank, and the Stanford Natural Language Inference Corpus (SNLI) model.", "ACE Entities/Events: ACE 2005 dataset BIBREF8 consists of 599 annotated documents from six different domains (newswire, broadcast news, broadcast conversations, blogs, forums, and speeches).", "We trained this architecture for the following datasets: Arguments: Argument component detection (major claim, claim, premise) in 402 persuasive essays BIBREF7.", "NER: CoNLL 2003 shared task on named entity recognition.", "Chunking: CoNLL 2000 shared task dataset on chunking."]

Table 8: Tabulation of question, answer, CoG decompositions and ground truth attributions for Table 9 and Table 10

Evidence	NIL+MonoT5	CoG+MonoT5
"GENIA NER: The Bio-Entity Recognition Task at JNLPBA BIBREF9 annotated Medline abstracts with information on bio-entities (like protein or DNA-names).",	-0.032	-0.013
"WNUT16: WNUT16 was a shared task on Named Entity Recognition over Twitter BIBREF10."	-0.043	-0.021

Table 9: Example of retriever score getting affected while using answer part as information unit vs decomposed information unit. The example provided is for the ground truth evidence from Table 8.

NIL+GPT4	CoG+GPT4
"For the models included in AllenNLP, we observed a training speed-up of 19-44%, while improving the test performance in 3 out of 5 datasets.", "'"The results for the second experiment, that uses AllenNLP and ELMo embeddings in combination with other input representations, are presented in the lower part of Table 1.", "Only for the GENIA dataset achieved the learned weighted average a significantly better performance than using the output of the second layer."	"We trained this architecture for the following datasets: Arguments: Argument component detection (major claim, claim, premise) in 402 persuasive essays BIBREF7 .", "ACE Entities/Events: ACE 2005 dataset BIBREF8 consists of 599 annotated documents from six different domains (newswire, broadcast news, broadcast conversations, blogs, forums, and speeches).", "POS: We use the part-of-speech tags from Universal Dependencies v. 1.3 for English with the provided data splits.", "Chunking: CoNLL 2000 shared task dataset on chunking.", "GENIA NER: The Bio-Entity Recognition Task at JNLPBA BIBREF9 annotated Medline abstracts with information on bio-entities (like protein or DNA-names).", "WNUT16: WNUT16 was a shared task on Named Entity Recognition over Twitter BIBREF10 .", "We use the CoNLL 2003 NER model, the Stanford Sentiment Treebank (SST-5) model, the constituency parsing model for the Penn TreeBank, and the Stanford Natural Language Inference Corpus (SNLI) model."

Table 10: Example of LLM attributions getting affected while using answer part as information unit vs decomposed information unit.

Attributable Information Units

Chat-based question-answer systems often include additional text, such as pleasantries, and provide answers that are abstractive in nature (complex sentences that incorporate multiple facts). This survey aims to understand your opinion on what specific facts within an answer are essential and should be credited as factual references in a document.

Instructions on Identifying Good and Bad Information Units

What are Good Information Units?
1. Information units should be relevant to the sentence.
2. Each information unit should be meaningful.
3. Information units should be related to the question.
4. *When the information units are combined, they should convey the same information as the original answer (including reasoning).*

What are Bad Information Units?
1. Information units that convey duplicate information.
2. Information units that are not complete statements.
3. Information units that are not meaningful in relation to the question.
4. Information units that repeat facts used in the introduction, conclusion, or summary of the answer.

You will be provided with a question and answer. Select the option that has the best set of information units for the answer.

1. **Question**: What is a second mortgage when you are buying a home?

   Answer: second mortgage is a loan taken out against a property that already has a home loan on it, secured by the equity in the home. Common examples include a home equity loan and a home equity line of credit (HELOC). To qualify for a second mortgage, you will need to meet a few financial requirements, such as a credit score of 620, a debt-to-income ratio of 43%, and enough equity in your first home. If you default on your loan, the original mortgage will receive all proceeds from the property's liquidation until it is all paid off.

   ○ ['A second mortgage is a loan.', 'A second mortgage is taken out against a property.', 'The property already has a home loan on it.', 'The second mortgage is secured by the equity in the home.', 'Home equity loan is an example.', 'Home equity line of credit (HELOC) is an example.', 'To qualify for a second mortgage, certain financial requirements must be met.', 'A credit score of 620 is required.', 'A debt-to-income ratio of 43% is required.', 'Enough equity in the first home is required.', 'Defaulting on a loan can result in the original mortgage receiving proceeds from the liquidation of the property.', 'The original mortgage will receive all proceeds until it is paid off.']

   ○ ['second mortgage is a loan taken out against a property that already has a home loan on it, secured by the equity in the home.', 'Common examples include a home equity loan and a home equity line of credit (HELOC).', 'To qualify for a second mortgage, you will need to meet a few financial requirements, such as a credit score of 620, a debt-to-income ratio of 43%, and enough equity in your first home.', "If you default on your loan, the original mortgage will receive all proceeds from the property's liquidation until it is all paid off."]

2. **Question**: Why don't nations unite to fight ISIS?

   Answer: There are many reasons why nations don't unite to fight ISIS. Some countries may not want to repatriate their citizens who have joined ISIS. Others may have different priorities or interests. However, many leaders and experts agree that it is important for the world to come together to combat ISIS.

   ○ ['Nations are not uniting to fight ISIS.', "There are many reasons why nations don't unite to fight ISIS.", 'Some countries may not want to repatriate their citizens.', 'Some citizens have joined ISIS.', 'Others may have different priorities.', 'Others may have different interests.', 'Many leaders agree.', 'Many experts agree.', 'It is important for the world to come together.', 'It is important for the world to come together to combat ISIS.']

Figure 4: Screenshot of Microsoft Forms used for survey.

7. **Question**: who plays in o brother where art thou

Answer: O Brother, Where Art Thou? is a 2000 comedy-drama film written, produced, co-edited and directed by Joel and Ethan Coen. The movie stars George Clooney as Everett, John Turturro as Pete Hogwallop and Tim Blake Nelson as Delmar O'Donnell. Other actors include Chris Thomas King as Tommy Johnson, John Goodman as Big Dan Teague and Holly Hunter as Penny. Is there anyone specific you would like to know more about?

○ ['"O Brother, Where Art Thou?" is a 2000 film.', '"O Brother, Where Art Thou?" is a comedy-drama.', '"O Brother, Where Art Thou?" was written by Joel and Ethan Coen.', '"O Brother, Where Art Thou?" was produced by Joel and Ethan Coen.', '"O Brother, Where Art Thou?" was co-edited by Joel and Ethan Coen.', '"O Brother, Where Art Thou?" was directed by Joel and Ethan Coen.', 'George Clooney is in the movie.', 'George Clooney stars as Everett.', 'John Turturro is in the movie.', 'John Turturro stars as Pete Hogwallop.', 'Tim Blake Nelson is in the movie.', "Tim Blake Nelson stars as Delmar O'Donnell.", 'Chris Thomas King is an actor.', 'Chris Thomas King played the role of Tommy Johnson.', 'John Goodman is an actor.', 'John Goodman played the role of Big Dan Teague.', 'Holly Hunter is an actor.', 'Holly Hunter played the role of Penny.', 'There is someone specific.', 'It is desired to know more about that person.']

○ ['The movie stars George Clooney as Everett', 'John Turturro as Pete Hogwallop', "Tim Blake Nelson as Delmar O'Donnell", 'Chris Thomas King as Tommy Johnson', 'John Goodman as Big Dan Teague', 'Holly Hunter as Penny']

(a) Example question and answer decomposition. First option shows FActScore decomposition and second option shows CoG decomposition.

7. **Question**: who plays in o brother where art thou

Answer: O Brother, Where Art Thou? is a 2000 comedy-drama film written, produced, co-edited and directed by Joel and Ethan Coen. The movie stars George Clooney as Everett, John Turturro as Pete Hogwallop and Tim Blake Nelson as Delmar O'Donnell. Other actors include Chris Thomas King as Tommy Johnson, John Goodman as Big Dan Teague and Holly Hunter as Penny. Is there anyone specific you would like to know more about?

More Details

● ['"O Brother, Where Art Thou?" i...   2
● ['The movie stars George Cloon...   1

(b) Case where human annotators preferred FActScore-based decomposition over CoG decomposition.

Figure 5: Human Annotation Error