Groundedness in Retrieval-augmented Long-form Generation: An Empirical Study

Alessandro Stolfo[*]
ETH Zürich
stolfoa@ethz.ch

Abstract

We present an empirical study of *groundedness* in long-form question answering (LFQA) by retrieval-augmented large language models (LLMs). In particular, we evaluate whether every generated sentence is grounded in the retrieved documents or the model's pre-training data. Across 3 datasets and 4 model families, our findings reveal that a significant fraction of generated sentences are consistently ungrounded, even when those sentences contain correct ground-truth answers. Additionally, we examine the impacts of factors such as model size, decoding strategy, and instruction tuning on groundedness. Our results show that while larger models tend to ground their outputs more effectively, a significant portion of correct answers remains compromised by hallucinations. This study provides novel insights into the groundedness challenges in LFQA and underscores the necessity for more robust mechanisms in LLMs to mitigate the generation of ungrounded content.

1 Introduction

One of the most significant challenges to the safe deployment of large language models (LLMs) is their propensity to generate hallucinated content (Bubeck et al., 2023; Alkaissi and McFarlane, 2023; Ji et al., 2023). The risk of hallucinating increases when LLMs are tasked with generating long content (i.e., more than a single sentence) (Shuster et al., 2021; Maynez et al., 2020). This is problematic because generating long-form text is a critical component of a number of important tasks, such as disambiguating complex topics, explicit problem decomposition and reasoning, question answering, and synthesis of information from multiple sources.

As a step towards mitigating hallucination, a number of studies have measured the *groundedness* of LLM generations (for a recent survey, see Li et al., 2023). In these studies, a sentence is considered to be grounded in a document if the text of the document supports the claim made in the sentence. However, many of these efforts focus on short LLM generations (i.e., a single word or phrase) rather than long generations (Bohnet et al., 2022). Others rely on Google searches for exact string match as a heuristic for evaluating or improving groundedness (Agrawal et al., 2023; Athaluri et al., 2023; Gao et al., 2023a).

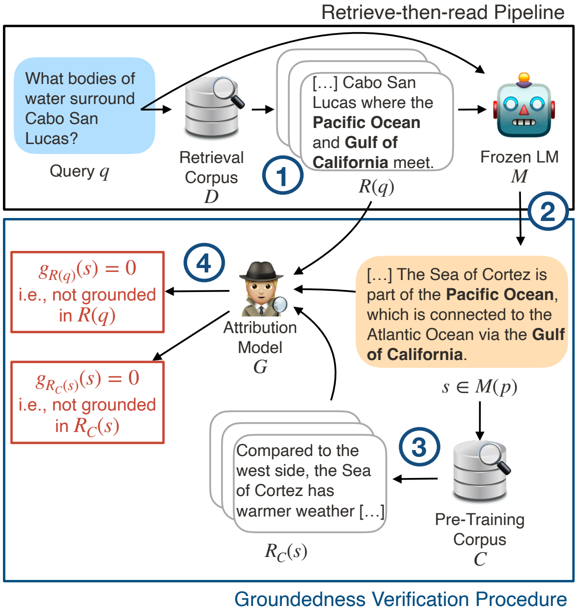
Retrieve-then-read Pipeline
What bodies of water surround Cabo San Lucas?
Query q
Retrieval Corpus D
1
[…] Cabo San Lucas where the **Pacific Ocean** and **Gulf of California** meet.
$R(q)$
Frozen LM M
2
$g_{R(q)}(s) = 0$ i.e., not grounded in $R(q)$
4
Attribution Model G
[…] The Sea of Cortez is part of the **Pacific Ocean**, which is connected to the Atlantic Ocean via the **Gulf of California**.
$s \in M(p)$
$g_{R_C(s)}(s) = 0$ i.e., not grounded in $R_C(s)$
3
Compared to the west side, the Sea of Cortez has warmer weather […]
Pre-Training Corpus C
$R_C(s)$
Groundedness Verification Procedure

Figure 1: **Our experimental setup.** Using a set of retrieved documents (1), an LLM generates an answer in an LFQA setting (2). Then, the model's pre-training corpus is searched for documents related to the generation (3). Finally, a grounding model verifies whether the model's response is supported by any of the considered documents (4).

In this work, we focus on long-form question answering (LFQA) due to its generality and relevance. Our research addresses two central questions: 1. how frequently do LLMs generate ungrounded sentences in LFQA? and, 2. how do model size, family, pre-training recipe, and decod-

[*] Work partially carried out while at Oracle Labs.

Findings of the Association for Computational Linguistics: NAACL 2024, pages 1537–1552
June 16-21, 2024 ©2024 Association for Computational Linguistics

ing style affect this rate? We address these questions by studying the provenance of the information contained in the model's generations.

LFQA is typically aided by retrieval augmentation (Karpukhin et al., 2020; Izacard et al., 2022, *inter alia*). By harnessing external data sources, these models incorporate information pertinent to the query, reducing the likelihood of hallucinations and incorrect outputs (Shuster et al., 2021). However, there is no guarantee that models consistently utilize the retrieved information in their outputs (Krishna et al., 2021). This setting presents additional challenges in measuring groundedness: the generated text is likely influenced by both the inference-time context (i.e., the retrieved information) and the extensive pre-training of the model. While groundedness relative to the retrieved information is well-studied, we also attempt to ground model-generated text in specific pre-training documents.

In particular, we measure whether the text generated in a retrieval-augmented fashion contains information that is *grounded* in the retrieved documents or in the model's pre-training corpus (our procedure is illustrated in Figure 1). We employ a groundedness-verification model (Honovich et al., 2022) that determines whether a portion of the model's output can be attributed to a given text passage. We analyze four different families of pre-trained language models on three datasets. We discover that, even when containing ground-truth answers, a significant portion of the generated sentences are not grounded in the retrieved or pre-training documents and may include fabricated claims. This trend persists across the range of models and datasets examined.

Additionally, we study the impact of model size, decoding strategy, and instruction tuning on the rate of correct and hallucinated content. We find that the larger models are generally more adept at grounding their outputs in the given sources. However, even for the largest models analyzed (Falcon 180B; Penedo et al., 2023), approximately 25% of the outputs that contain ground-truth answers are not grounded. Interestingly, we observe that instruction tuning and beam search decoding strategies contribute to a reduction in the generation of ungrounded content. These methods appear to help models better utilize training and inference-time documents, thereby mitigating the tendency to produce fabricated information.

2 Background

Hallucination & Factuality. Before describing our experimental setup, we more precisely define hallucination. Following previous work, we define a *hallucination* as text that is not grounded in the data provided to the model at either training or inference time (Ji et al., 2023; Agrawal et al., 2023). Such hallucinations are sometimes characterized as *open-domain* hallucinations in order to distinguish them from semantic deviations of the generated output in, e.g., machine translation—referred to as *closed-domain* hallucination (Ji et al., 2023). It is important to distinguish *factuality* from hallucination. Specifically, factuality, or the factual correctness of (generated) text, refers to the quality of being based on a fact, i.e., world knowledge (Maynez et al., 2020). Note that a model might output text that is grounded in its pre-training or inference-time data, yet is factually incorrect. While the number of grounded, factual errors may be reduced by improving the factuality of the data, preventing a model from generating text that is neither grounded nor factually accurate is a challenging problem with no known solution.

Setting. We consider the task of open-domain LFQA in a few-show setting. We adopt the *retrieve-then-read* paradigm, in which a language model performs question answering using passages retrieved from a corpus at inference time (Lewis et al., 2020; Izacard and Grave, 2021). This approach, although simple, was shown to improve the few-shot performance of pre-trained LLMs on multiple QA benchmarks (Si et al., 2022; Mallen et al., 2023).

3 Experimental Procedure

In this section, we detail our experimental setup, including how correctness and groundedness are measured, as well as the datasets used. We begin by defining notation.

3.1 Notation

Let $\mathcal{Q}$ be a collection of questions and $\mathcal{D}$ be a corpus of documents. Consider a question $q \in \mathcal{Q}$, which is annotated with a set $\mathcal{Y}$ of ground-truth string answers. A retrieve-then-read system proceeds in 3 steps. First, a retriever, $R : \mathcal{Q} \to \mathcal{D}$, returns a set of k documents, $R(q) = \{d^{(1)}, \dots, d^{(k)}\}$. Second, the question and documents are combined to form a prompt, p. Finally, the question-answering model, M, consumes the prompt and produces an answer, $M(p) = \langle s_1, s_2, \dots \rangle$, which is comprised of sentences s_i's. We denote by $\mathcal{S}$ the set of all generated sentences. In a few-shot scenario, the prompt p additionally contains a set of question-documents-answer triples, which include manually annotated answers from a held-out dataset.

3.2 Measuring Correctness

Like previous work (Gao et al., 2023b), we adopt a definition of correctness based on exact match (EM). Specifically, for a question-answer pair $(q, \mathcal{Y})$, The accuracy of a model output, $M(p)$, is computed as the fraction of elements from $\mathcal{Y}$ that are substrings of $M(p)$, i.e.,

$$\mathrm{EM}(M(p), \mathcal{Y}) = \frac{|\{y \in \mathcal{Y} : \texttt{substr}(y, M(p))\}|}{m},$$

where $\texttt{substr}(y, M(p)) := \mathbb{1}\{\exists\ s \in M(p) : y \in s\}$ indicates whether y is a substring of the model output, and $m = |\mathcal{Y}|$. Concretely, in the example illustrated in Figure 1, the set of ground-truth answers is $\mathcal{Y} = \{$ `Pacific Ocean`, `Gulf of California`, `Sea of Cortez` $\}$. Since the model output includes the strings `Pacific Ocean` and `Gulf of California`, the accuracy of the model on this example is $\mathrm{EM}(M(p), \mathcal{Y}) = \frac{2}{3}$. As we are interested in separately analyzing the groundedness of long-form outputs that contain correct answers and those that contain no correct answers, we refer to the set of model outputs with an exact match of 0 as EM^0, and all other model outputs as belonging to EM^+.

3.3 Measuring Groundedness

In our work, we assume that the question-answering model, M, is pre-trained on a corpus, $\mathcal{C}$. We measure the extent to which each model output is grounded in the retrieved documents, $R(q)$, as well as the training corpus, $\mathcal{C}$. Since we focus on LFQA, we follow previous work and measure the groundedness of each sentence of each model output independently (Gao et al., 2023a).

Groundedness in the retrieved documents. Formally, let S be the set of all sentences and $G : \mathcal{D} \times \mathcal{S} \to \{0, 1\}$ be a grounding model, which takes a document and a sentence and outputs 1 if the sentence is grounded in the document. Then, a model-generated sentence, s, is grounded in a collection of documents, $\mathcal{Z}$, if there exists a document in $\mathcal{Z}$ that grounds s. For example, for the retrieved documents, $R(q)$,

$$g_{R(q)}(s) = \begin{cases} 1 & \exists\ d \in R(q) : G(d, s) = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

In words, the function $g_{R(q)}(s)$ returns 1 if at least one of the documents in $R(q)$ grounds s, and 0 otherwise.

Groundedness in the pre-training data. While retrieval-augmented text-generation models have been shown to draw on the retrieved documents, they also produce sentences that are not grounded in the text provided to them during inference. Though some of these sentences may not be grounded in any of the data ever provided to the model, others may be grounded in the model's pre-training data. Ideally, we would compute $g_{\mathcal{C}}(s)$, i.e., whether the model-generated sentence, s, is grounded in *any* pre-training document. Since this computation is prohibitively expensive, we approximate $g_{\mathcal{C}}(s)$ by performing post-generation retrieval from the model's pre-training corpus, $\mathcal{C}$, and then testing whether s can be grounded in the retrieved documents. In practice, we compute the dense representation of s and of each document in the corpus using the MiniLM-v2 (Wang et al., 2020b) sentence-Transformer (Reimers and Gurevych, 2019). We perform an exact search for the top 5 pre-training documents $R_{\mathcal{C}}(s)$ closest to s in terms of cosine similarity using the FAISS library (Johnson et al., 2019). The grounding $g_{R_{\mathcal{C}}(s)}(s)$ of s is checked against each retrieved document, as in Eq. 1. We report additional experimental details in Appendix B.

Groundedness scores. To quantify the groundedness of statements generated by the model, we calculate the fraction of sentences in the set $\mathcal{S}$ that are grounded in the retrieved or pre-training documents. Specifically, we compute the following expression:

$$\frac{1}{|\mathcal{S}|}\left|\{s \in \mathcal{S} : \mathrm{condition}(s)\}\right|$$

where condition denotes the grounding condition applied to each statement s. Based on this formulation, we compute groundedness scores for:

- the **retrieved documents only**, considering s for which $g_{R(q)}(s) = 1$ and $g_{R_{\mathcal{C}}(s)}(s) = 0$,

- the **pre-training corpus only**, s meeting the condition $g_{R(q)}(s) = 0$ and $g_{R_{\mathcal{C}}(s)}(s) = 1$,
- **both** ($g_{R(q)}(s) = 1$ and $g_{R_{\mathcal{C}}(s)}(s) = 1$),
- **none** ($g_{R(q)}(s) = 0$ and $g_{R_{\mathcal{C}}(s)}(s) = 0$).

3.4 Experimental Setup

Datasets. We perform experiments on three datasets:

1. **ASQA** (Stelmakh et al., 2022) - ambiguous factual questions that have multiple correct answers. The desired model behavior includes providing a long-form generation that discusses all the correct answers.
2. **HotpotQA** (Yang et al., 2018) - multi-hop question answering that requires reasoning over multiple entities in Wikipedia. The desired model behavior includes explicitly providing its reasoning in addition to the correct answer.
3. **StrategyQA** (Geva et al., 2021) - multi-hop question answering, similar to HotpotQA. The correct answers to questions in this dataset are either True or False.

On ASQA, for each question, we perform dense retrieval (using GTR; Ni et al., 2022) from Wikipedia and include $k = 3$ retrieved paragraphs in the model's prompt (example provided in Appendix A). For HotpotQA and StrategyQA, instead of performing retrieval, we supply the model with the documents from which the correct answer can be determined (i.e., R is an oracle function that retrieves necessary and sufficient information). We do this to mitigate the effects of poor retrieval, since the reasoning over multiple documents required by these datasets makes correct retrieval necessary for reasonable performance.

Models. We experiment with four different families of Transformer-based pre-trained language models: Pythia (Biderman et al., 2023), Falcon (Penedo et al., 2023), MPT (Team, 2023), and Silo (Min et al., 2023). Post-generation retrieval is carried out on the whole training corpora for Pythia (the Pile; Gao et al., 2020) and Silo (Open-license Corpus; Min et al., 2023). For the MPT and Falcon models, we retrieve from the C4 dataset (Raffel et al., 2020), which represents ∼60% of the training data used for MPT (in terms of # of tokens), and was created in a similar way to the Falcon's training corpus (Penedo et al., 2023).

Grounding. Similar to prior work (Bohnet et al., 2022; Gao et al., 2023b), we assess groundedness using a natural language inference-based approach, which was shown to have a strong correlation with human judgment (Rashkin et al., 2023; Gao et al., 2023b; Chen et al., 2023). In particular, we use TRUE (Honovich et al., 2022), a T5-11B (Raffel et al., 2020) model trained on a set of natural language inference datasets to automatically determine whether a generated statement is supported by a given text passage. We carry out a manual validation of the TRUE model, in which we provide a small set of annotators with 100 instances of $(q, s, R(q), R_{\mathcal{C}}(s), g_{R(q)}(s), g_{R_{\mathcal{C}}(s)}(s))$. The annotators are asked to judge whether the grounding model's predictions $g_{R(q)}(s)$ and $g_{R_{\mathcal{C}}(s)}(s)$ are correct (i.e., whether the generated statement is correctly determined to be grounded or ungrounded with respect to the considered sources). We observe that the annotators agree with the model in 82% of the cases overall and in 98% of the correct but ungrounded cases. We provide additional details about the groundedness verification procedure and its manual validation in Appendix C.

4 How Frequently are Generations Grounded?

We begin our analysis by measuring the rate at which models of various sizes and families generate sentences that are grounded in the retrieved documents as well as the pre-training data. As an example, consider Figure 2, which provides a visualization of the sentences generated by the Pythia 12B model on the ASQA dataset divided into 8 sectors. Each sector represents one group in the cross-product of the following categories:

- whether a sentence could be grounded in the retrieved documents, the pre-training data, both, or neither;
- and whether the sentence was part of a long-form generation that was deemed incorrect (EM^0), or not (EM^+), according to exact match.

We expect that most sentences that are part of EM^+ will be grounded in either the retrieved documents or the pre-training corpus (or both), since those documents represent the source of the model's correctness. For sentences that are part of EM^0, we make no assumptions on the frequency of grounding. That is because incorrect answers could arise

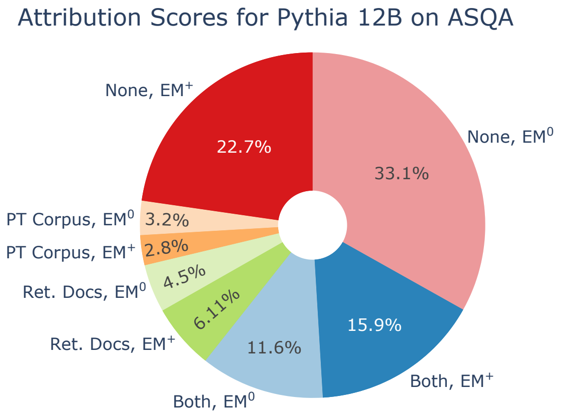

Figure 2: **Groundedness & correctness.** Each of the 8 sectors in the chart corresponds to a specific combination of groundedness (in the retrieved documents, pre-training data, both, or neither) and EM correctness (either belonging to EM^0 or EM$^+$). The area of a sector corresponds to the fraction of all model-generated sentences over all ASQA test examples that exhibit that groundedness-correctness combination.

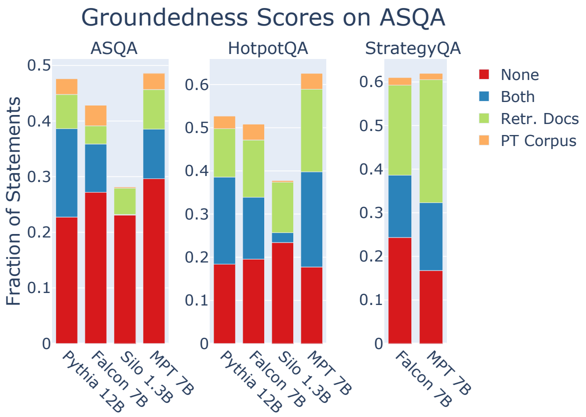

Figure 3: **Groundedness across datasets.** The height of each bar represents the fraction of generated sentences that belong to partially correct generations. A significant fraction of these sentences are not grounded in either the retrieved or pre-training documents.

from the model emitting ungrounded sentences, or grounded sentences that are off-topic or otherwise incorrect.

Figure 2 reveals that overall, for Pythia 12B, ∼44% of the sentences generated are grounded in at least one of the two sources considered (i.e., blue, green, and orange sectors), and that ∼48% belong to EM$^+$ (i.e., the darker-color sectors). Interestingly, we observe that nearly half of the generated sentences belonging to EM$^+$ cannot be grounded in the retrieved documents or pre-training corpus. This is unexpected since, by virtue of containing at least some of the ground-truth answers, the model demonstrates that it may have access to relevant information. Upon inspection, we find that the high proportion of ungrounded sentences in these answers contain ground-truth named entities or text snippets that are presented in a nonsensical or factually incorrect manner. We provide examples of such outputs in Section 6. We note that sentences that cannot be grounded via our methods could be the result of sub-optimal retrieval in the pre-training corpus, or errors from the TRUE (grounding) model. We elaborate more on these concerns in Appendices B and C.

Additionally, Figure 2 also shows that roughly one-fourth of the generations can be grounded in both the pre-training and retrieved document. This is likely due to overlap between the pre-training and retrieval corpora, or the appearance of common knowledge present in both corpora.

Does ungrounded content appear consistently? For the remainder of this analysis, we focus on outputs that contain ground-truth answers (i.e., EM$^+$), as they might represent subtler and more interesting cases of undetected hallucination. Figure 3 visualizes of frequency of grounded sentences in correct and partially correct answers on ASQA, HotPotQA, and StrategyQA for four models from the four families considered: MPT 7B, Falcon 7B, Silo 1.3B, and Pythia 12B. Note that the height of each bar represents the fraction of generated sentences that belong to long-form generations containing at least 1 ground-truth answer.

On all three datasets and all models considered, we observe that a substantial fraction of the outputs in EM$^+$ are not grounded in the retrieved documents or in the pre-training data.[1] This consistency across different models and datasets indicates a prevalent pattern where models are able to generate correct answers that are found in sentences that are not directly supported by the retrieved documents or pre-training data.

5 What Factors Affect Groundedness?

In this section, we study the interplay between the tendency of models to generate grounded content and three factors: the size of the model (Section 5.1), the decoding strategy (Section 5.2), and instruction-tuning (Section 5.3).

[1] Results for Pythia and Silo are omitted for StrategyQA due to their accuracy being marginally better than a random chance, precluding meaningful analysis.

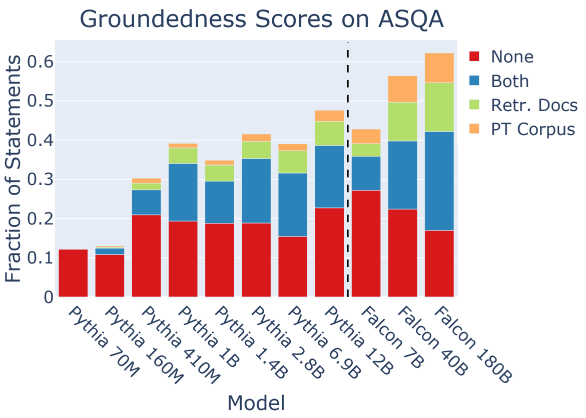

Figure 4: **Groundedness by size.** As before, the height of each bar represents the fraction of generated sentences that belong to partially correct generations. Increased model size correlates with an increase in the number of sentences in EM$^+$, but also an increase in groundedness.

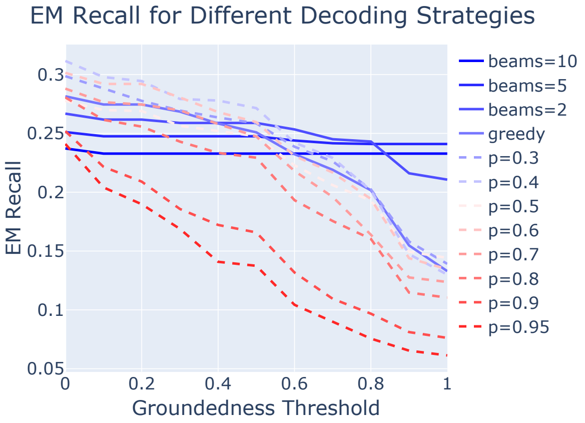

Figure 5: **Groundedness by decoding.** Exact Match (EM) scores against the minimum fraction of sentences required for a model generation to be considered valid (*groundedness threshold*). As the grounding threshold tightens, the EM scores for random sampling quickly degrade. The scores for beam search, however, remain roughly unaltered, indicating a higher level of groundedness. Results obtained with Pythia 12B on ASQA.

5.1 Model Size

For smaller models, particularly Pythia 70M and Pythia 410M, the majority of generated sentences cannot be grounded in either the retrieved documents or the pre-training corpus (Figure 4). Interestingly, while they do not have a clear grounding to either the documents in the context or the pre-training corpus, the responses generated by these models occasionally match tokens from the ground truth answers. A possible interpretation of this result is that these models may rely more on internal heuristics or pattern-matching capabilities rather than effectively using external information or learned knowledge (Elazar et al., 2022; McCoy et al., 2019).

Pythia models in the range 1-12B generate an increased fraction of grounded output compared to their smaller counterparts. However, no stark trend is observed within this size range. Conversely, for significantly larger models (Falcon 40B and 180B), there is a clear increase in the proportion of content that can be grounded to the provided context or the pre-training corpus. This indicates that larger models are better equipped to integrate and utilize external information from the provided context and their extensive pre-training. However, it is important to note that even with the largest models, there remains a non-negligible fraction of generated sentences that are part of EM$^+$ but cannot be grounded in either the retrieved documents or the pre-training corpus.

5.2 Decoding Strategy

We measure the impact of the decoding algorithm on the frequency of grounded sentences as well as correctness. In particular, we test readily available decoding strategies: greedy decoding, nucleus sampling (Holtzman et al., 2019), and beam search. For nucleus sampling we vary the `top_p` parameter; for beam search, we test beam widths 2, 5, and 10.

Figure 5 illustrates the impact of employing various decoding methods with the Pythia 12B model on the ASQA dataset. In the Figure, the x-axis represents a *groundedness threshold*, i.e., the minimum fraction of sentences in a model generation that must be grounded (in either retrieved or pre-training documents) for that generation to be considered valid. That is, at $x = 1.0$, all sentences in a generation must be grounded for the generation to be valid. For any groundedness threshold x, the corresponding y-value represents the average Exact Match (EM) score across generations, where invalid generations automatically get a score of 0.

Intuitively, as the groundedness threshold becomes more stringent (i.e., as we require a higher fraction of sentences to be grounded), the EM scores should decrease. Indeed, this trend is observed for greedy decoding and nucleus sampling. However, an interesting deviation from this trend is observed in the case of beam search decoding. Unlike the other strategies, the EM scores for beam search do not exhibit a significant decline as the groundedness threshold increases. In particular,

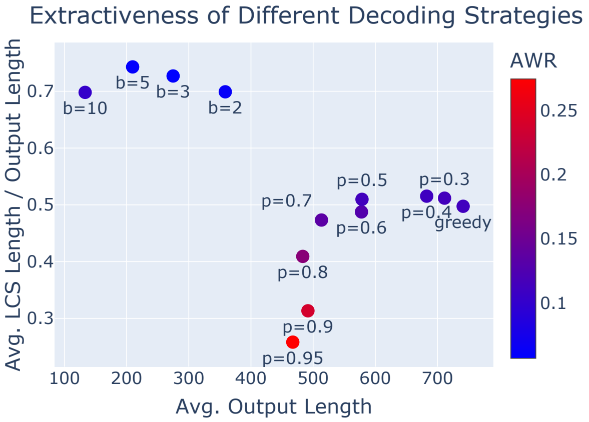

Figure 6: **Extractiveness by decoding.** Average ratio between the length of the LCS and the length of the model output (y-axis) against the average length of the output (x-axis). Length is measured as the number of characters. The color scale illustrates the average ratio of abstracted words (AWR) for the different decoding strategies: beam search (b), greedy, and nucleus sampling (p). Results obtained on ASQA with Pythia 12B.

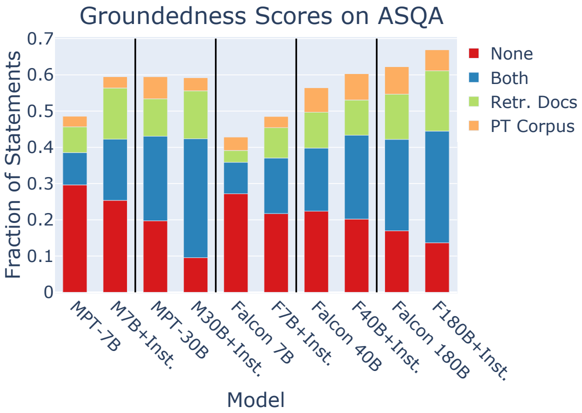

Figure 7: **Effect of instruction-tuning.** As before, the height of each bar represents the fraction of generated sentences that belong to EM^+. Compared to the corresponding base models, instruction-tuned models tend to exhibit greater correctness as well as a larger fraction of grounded sentences.

the decline is less steep as the number of beams increases. This result shows that while beam search may initially show lower EM scores compared to nucleus sampling without considering groundedness, its effectiveness emerges when groundedness is taken into account. A possible explanation for this phenomenon can be identified in the tendency of beam search to give a higher likelihood to sequences that previously appeared in the model input (Holtzman et al., 2019). Since in our setting the retrieved documents are provided as an input sequence to the model, greedy and nucleus sampling might assign a higher probability to grounded sentences.

To clarify this aspect, we carry out analyses of the models' *extractiveness* (i.e., the tendency of a model to replicate portions of text verbatim from the retrieved documents). The results are reported in Figure 6. By measuring the longest common substring (LCS) between the model outputs and the retrieved documents, and comparing this to the overall length of the model outputs, we find that a larger number of beams generally result in shorter outputs, but with proportionately longer common substrings. Additionally, we compute the proportion of abstracted words (i.e., words that do not appear in any of the documents included in the prompt) present in the model output and notice that it decreases as the decoding becomes less random (as the parameter p in nucleus sampling decreases, as one would expect), but also that it becomes substantially smaller with beam search. These findings clarify how beam search affects groundedness and suggest a trade-off between extractiveness and groundedness in retrieval-augmented generation.

5.3 Instruction Tuning

Figure 7 illustrates the impact of instruction tuning on the groundedness and correctness of sentences generated by various models on the ASQA dataset. For instruction-tuned models, we observe a marked improvement in both the overall correctness and the fraction of grounded sentences. This trend holds across various model families and sizes, suggesting that instruction tuning enhances the model's ability to generate content that not only contains ground-truth answers but is also more grounded in the provided context or pre-training data.

6 Examples of Model Output

In Table 1, we present a handful of representative sentences generated by the Pythia 12B model on the ASQA dataset. Additionally, we supply relevant portions of the retrieved documents, $R(q)$, (from Wikipedia) as well as the documents retrieved from the pre-training corpus, after the model generates an answer. Overall, the selected sentences demonstrate the model's ability to generate content that, while sometimes correctly includes ground-truth answers, showcases varying degrees of groundedness.

Consider the example in the first row of Table 1. In this instance, the model includes the correct

Query q	Sentence s	$d \in R(q)$	$d \in R_C(s)$	$g_{R(q)}(s)$	$g_{R_C(s)}(s)$
What bodies of water surround Cabo San Lucas?	…The **Sea of Cortez** is part of the **Pacific Ocean**, which is connected to the Atlantic Ocean via the **Gulf of California**.	…El Arco, a natural stone arch over ocean in Cabo San Lucas where the Pacific Ocean and Gulf of California meet …	…compared to the west side of the Baja Peninsula, the Sea of Cortez has warmer weather and a climate more in line with most people's…	0	0
When did the Brisbane Broncos last win a premiership?	The Brisbane Broncos last won a premiership in **2006**.	…having endured twelve completed seasons without winning the premiership since their most recent success in 2006.	…The Broncos did not start the 2006 season well, suffering a 36–4 defeat by the previous year's runners-up, the North Queensland…	1	0
When will they finish building the Sagrada Familia?	…The construction of this building began in 1882 and it will be finished in **2026**.	…is large unfinished Roman Catholic church in Barcelona, designed by Catalan architect Antoni Gaudí (1852–1926). Gaudí's …	…'s still being worked into today a goal of completion in 2026 a century after Gaudi's death…	0	1
Names of the rivers in the garden of Eden?	The rivers in Eden were the **Pishon**, the **Gihon**, and the **Tigris**, and **Euphrates**.	…divided into four branches: the Pishon, where the gold is; the Gihon; the Tigris; and the Euphrates. …	…leaving the garden branching out into four streams, the Euphrates, Gihon, Pison, and the Tigris.	1	1

Table 1: **Examples of generations** by Pythia 12B on ASQA, along with relevant portions of the retrieved documents and the predictions from the groundedness-verification model ($g_{R(q)}(s)$ and $g_{R_C(s)}(s)$). The sentences were selected among the generations containing one or more ground-truth answers (reported in bold), i.e., they belong to EM$^+$.

named entities within its generated text. However, it also produces an ungrounded (and incorrect) claim that the Pacific Ocean is "*connected to the Atlantic Ocean via the Gulf of California.*" This statement is factually incorrect and represents a clear case of the model generating a plausible-sounding but erroneous connection between entities.

Inspecting the model-generated answers, we observed a common trend in which models blend fabricated content with factually correct statements. This trend is still present in larger models, whose generations are overall more grounded than small models but might feature a melding of fact and hallucination that can be more subtle. In Table 2, we provide some examples obtained with Falcon 180B. These instances illustrate the models' ability to accurately retrieve and use specific terms from the training data or the retrieved documents while also highlighting the challenge of ensuring that the relationships and contexts it generates are factual.

7 Related Work

Long-form Question Answering. LFQA (also referred to as generative QA) (Fan et al., 2019) is a question-answering task in which the goal is to generate—rather than extract—the correct answer to a question, usually by drawing from provided passages. The majority of the work in LFQA involves a human evaluation process aimed at measuring the factual correctness of generated answers (Su et al., 2022; Nakano et al., 2021; Krishna et al., 2021). In the absence of human evaluation, the quality of an answer is typically evaluated using automatic metrics such as ROUGE (Lin, 2004). However, these metrics require human-annotated answers, and, more importantly, they fail to pick up hallucinated content (Xu et al., 2023; Ji et al., 2023; Krishna et al., 2021). Other approaches, such as factual consistency (Durmus et al., 2020; Wang et al., 2020a), are used to measure the faithfulness of a generated summary to a corresponding set of source documents. Similar strategies were subsequently applied to evaluate dialogue systems (Honovich et al., 2021) and to edit and improve the output of retrieval-augmented language models (Gao et al., 2023a).

Groundedness & Attribution. In light of the importance of mitigating hallucination, work on

Query q	**Generated Sentence** s
How many times did Brazil win the FIFA World Cup?	Brazil has won the World Cup 5 times. They won in 1958, 1962, 1970, 1994, and 2002. **The only other country that has won as many times as Brazil is Germany** (also known as West Germany).
Who did the artwork for Pink Floyd's wall?	The artwork for Pink Floyd's Wall was created by Gerald Scarfe. The artwork on the cover of The Wall features a uniform row of white bricks **with a hammer in the foreground**.
The director of the romantic comedy "Big Stone Gap" is based in what New York city?	Adriana Trigiani is the writer of the romantic comedy Big Stone Gap. **She was born and raised in New York City**. She works in Greenwich Village, New York City.

Table 2: **Examples of generations** by Falcon 180B on ASQA (first two rows) and HotpotQA (last row). The generated content features a mixture of correct and fabricated content (the latter highlighted in **red**): Germany won the World Cup 4 times, not 5, *The Wall*'s album cover does not feature a hammer in the foreground, and Adriana Trigiani was actually born in Virginia.

evaluating groundedness has enjoyed significant attention (Li et al., 2023). Initial work develops the *attributable to identified sources* (AIS) score, which represents a human evaluation quantifying the degree to which generated text adheres to its cited sources (Rashkin et al., 2023). Later work demonstrates that AIS can be well-approximated using a model trained for predicting entailment, as in natural language inference (Honovich et al., 2022). However, in these studies, groundedness is often computed against provided passages (Bohnet et al., 2022; Gao et al., 2023b; Yue et al., 2023). In contrast to previous studies that treat the accuracy of correct entities and output groundedness as distinct aspects (Gao et al., 2023b), our research delves into exploring their interconnection.

More similar to our work are those that measure groundedness against documents retrieved by a web search API (Liu et al., 2023; Gao et al., 2023a; Chen et al., 2023). In some cases, such as checking the existence of a generated reference, this is an appropriate strategy (Agrawal et al., 2023). But in the general case, we argue that such an approach can be problematic because of the varying quality, factuality, and relevance of internet search results.

Another line of work explores the relationship between a model's generated text and its pre-training data. For example, one study measures how often models repeat content verbatim from their pre-training corpora (McCoy et al., 2023). Similarly, other works study the provenance of the model-generated content within pre-training corpora but rely on gradient-based methods (Han and Tsvetkov, 2022) or metrics based on n-gram overlap (Weller et al., 2024). Others analyze model generations with the intent to characterize the extent to which they can be attributed to a model's parametric memory vs. additional information provided at inference. However, this is measured by either constructing prompts that contain sentences that conflict with information in the pre-training data (Longpre et al., 2021), or by drawing on correlations with respect to the rarity of the entities produced in model generations (Mallen et al., 2023). Unlike these works, we verify whether the model output can be supported by passages retrieved from the pre-training corpus, as well as the context supplied during inference.

8 Conclusion

This study analyzes the rate at which the long-form output produced by retrieval-augmented LLMs is grounded in retrieved documents and pre-training data. Through empirical analysis across various models and datasets, we highlight the propensity of LLMs to blend correct information with hallucinated content. Our findings indicate that this tendency is prevalent across different model sizes and persists even in the largest models available. Our analyses reveal that while larger models generally produce more grounded content, they are not immune to generating ungrounded information. We observed that instruction tuning and beam search decoding reduce ungrounded sentence generation. Aligned with the results of concurrent research (Choi et al., 2023), our findings point to specialized decoding algorithms being good candidates for significantly reducing hallucination.

Limitations

We identify a handful of limitations of our work below.

Imperfect Retrieval from Pre-training Corpus. Given the size of pre-training corpora, it is possible for our approach to exhibit false negatives.

That is, when attempting to retrieve passages in the pre-training corpus that ground a model-generated sentence, we may incorrectly conclude that the generated sentence is not grounded in the pre-training corpus. This is a result of retrieval being imperfect. Despite this, we suspect that the rate of these false negatives is low given: a) manual inspection of the ungrounded sentences and b) the relatively high true positive rate, i.e., that rate at which we successfully ground generated sentences in pre-training documents.

Scattered Correct Information. When attempting to ground a model-generated sentence, our approach considers each source document independently. However, this is limited when a generated sentence amalgamates information from multiple sources, in a way no single source fully supports. An enhanced method, potentially examining the concatenation of multiple source documents, could address this issue, and we propose this as an area for future research. A related—and more general—limitation is our reliance on a grounding model, which is also imperfect.

Dependence on Pre-training Data Availability. Our methodology relies on accessing the pre-training corpus or a dataset containing most of the documents contained therein. This dependence is a significant limitation, especially for models where the pre-training data is not readily available or is incomplete. Indeed, this requirement significantly limits the family of models we use in our experiments.

Acknowledgments

Alessandro would like to thank Ari Kobren for providing an incredibly generous amount of feedback, guidance, and practical support throughout the project. He would also like to thank the whole Machine Learning Research Group at Oracle Labs, most notably Adam Pocock, Qinlan Shen, and Michael Wick for providing feedback and helping with manual annotation, Jason Peck and Jeffrey Alexander for the support with the computing infrastructure, and Stephen Green for practical support. Alessandro is supported by armasuisse Science and Technology through a CYD Doctoral Fellowship.

References

Ayush Agrawal, Lester Mackey, and Adam Tauman Kalai. 2023. Do language models know when they're hallucinating references? *arXiv preprint arXiv:2305.18248*.

Hussam Alkaissi and Samy I McFarlane. 2023. Artificial hallucinations in ChatGPT: Implications in scientific writing. *Cureus*, 15(2).

Sai Anirudh Athaluri, Sandeep Varma Manthena, VSR Krishna Manoj Kesapragada, Vineel Yarlagadda, Tirth Dave, and Rama Tulasi Siri Duddumpudi. 2023. Exploring the boundaries of reality: Investigating the phenomenon of artificial intelligence hallucination in scientific writing through ChatGPT references. *Cureus*, 15(4).

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

Bernd Bohnet, Vinh Q Tran, Pat Verga, Roee Aharoni, Daniel Andor, Livio Baldini Soares, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, Kai Hui, et al. 2022. Attributed question answering: Evaluation and modeling for attributed large language models. *arXiv preprint arXiv:2212.08037*.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Hung-Ting Chen, Fangyuan Xu, Shane A Arora, and Eunsol Choi. 2023. Understanding retrieval augmentation for long-form question answering. *arXiv preprint arXiv:2310.12150*.

Sehyun Choi, Tianqing Fang, Zhaowei Wang, and Yangqiu Song. 2023. KCTS: Knowledge-Constrained tree search decoding with token-level hallucination detection. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14035–14053, Singapore. Association for Computational Linguistics.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Amir Feder, Abhilasha Ravichander, Marius Mosbach, Yonatan Belinkov, Hinrich Schütze, and Yoav Goldberg. 2022. Measuring causal effects of data statistics on language model'sfactual'predictions. *arXiv preprint arXiv:2207.14251*.

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.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 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, Toronto, Canada. Association for Computational Linguistics.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023b. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Xiaochuang Han and Yulia Tsvetkov. 2022. Orca: Interpreting prompted language models via locating supporting data evidence in the ocean of pretraining data. *arXiv preprint arXiv:2205.12600*.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

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.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. 2021. q^2: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. 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*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

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.

Kalpesh Krishna, Aurko Roy, and Mohit Iyyer. 2021. Hurdles to progress in long-form question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4940–4957, Online. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Dongfang Li, Zetian Sun, Xinshuo Hu, Zhenyu Liu, Ziyang Chen, Baotian Hu, Aiguo Wu, and Min Zhang. 2023. A survey of large language models attribution. *arXiv preprint arXiv:2311.03731*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Nelson F Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating verifiability in generative search engines. *arXiv preprint arXiv:2304.09848*.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. 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.

R Thomas McCoy, Paul Smolensky, Tal Linzen, Jianfeng Gao, and Asli Celikyilmaz. 2023. How much do language models copy from their training data? evaluating linguistic novelty in text generation using raven. *Transactions of the Association for Computational Linguistics*, 11:652–670.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Sewon Min, Suchin Gururangan, Eric Wallace, Hannaneh Hajishirzi, Noah A Smith, and Luke Zettlemoyer. 2023. Silo language models: Isolating legal risk in a nonparametric datastore. *arXiv preprint arXiv:2308.04430*.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.

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.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

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.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. 2023. Measuring attribution in natural language generation models. *Computational Linguistics*, pages 1–66.

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.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Lee Boyd-Graber, and Lijuan Wang. 2022. Prompting GPT-3 to be reliable. In *The Eleventh International Conference on Learning Representations*.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Dan Su, Xiaoguang Li, Jindi Zhang, Lifeng Shang, Xin Jiang, Qun Liu, and Pascale Fung. 2022. Read before generate! faithful long form question answering with machine reading. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 744–756, Dublin, Ireland. Association for Computational Linguistics.

MosaicML NLP Team. 2023. Introducing MPT-7B: A new standard for open-source, commercially usable LLMs.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020a. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020b. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788.

Orion Weller, Marc Marone, Nathaniel Weir, Dawn Lawrie, Daniel Khashabi, and Benjamin Van Durme. 2024. “according to . . . ”: Prompting language models improves quoting from pre-training data. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2288–2301, St. Julian’s, Malta. Association for Computational Linguistics.

Fangyuan Xu, Yixiao Song, Mohit Iyyer, and Eunsol Choi. 2023. A critical evaluation of evaluations for long-form question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3225–3245, Toronto, Canada. Association for Computational Linguistics.

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.

Xiang Yue, Boshi Wang, Kai Zhang, Ziru Chen, Yu Su, and Huan Sun. 2023. Automatic evaluation of attribution by large language models. *arXiv preprint arXiv:2305.06311*.

A Prompting Details

To elicit models to generate long-form answers leveraging the retrieved documents, we construct a few-shot prompt with exemplars of question-documents-answer triples. In particular, we use 3-shot prompts on HotpotQA and StrategyQA and 2-shot prompts on ASQA (due to the longer documents used for this dataset). In Table 4, we provide an example of a prompt used on ASQA.

B Pre-training Corpus Retrieval

B.1 Retrieval Details

The retrieval procedure from the pre-training corpora was carried out by first dividing the corpus into passages of 768 contiguous characters. Then, each passage was embedded using the MiniLM-v2 (Wang et al., 2020b) sentence-Transformer (Reimers and Gurevych, 2019). This procedure was carried out in parallel by 12 64-CPU computing nodes in parallel and took ∼24 hours for each corpus. Finally, given a sentence generated by a model, a search for the 5 most relevant passages in the corpus was performed using the FAISS library (Johnson et al., 2019). The search was carried out in parallel by 20 computing nodes on different subsets of the corpus and took ∼12 hours for all sentences generated by a model on a dataset.

B.2 Validation of the Retrieval Procedure

Given the size of pre-training corpora, it is possible for our retrieval approach to produce false negatives. However, we believe that the rate of these false negatives is low for two main reasons. First, the rate of positives (i.e., the rate at which we successfully ground generated sentences in pre-training documents) is relatively high (e.g., 34% and 48% of the overall statements for Pythia 12B and Falcon 180B, respectively). Second, when manually inspecting ungrounded sentences, we notice that a large number of them are nonsensical or contain fabricated information.

Moreover, we carry out an additional validation study. Recognizing the impracticality of conducting a comprehensive search across the entire pre-training corpus to definitively show the absence of supporting text for a given claim, we opt for a focused approach. We analyze a random subset of the generated statements that are judged ungrounded and manually determine whether each sentence is factually correct. We found that 40 out of 50 instances inspected contain factually incorrect information. We report some examples of generations by Pythia 12B in Table 3.

While the factual incorrectness of a sentence does not definitively rule out support from pre-training documents, it strongly suggests their absence. We therefore believe that our method of retrieving from the pre-training corpus represents a reasonable approximation for verifying groundedness in this context.

C Groundedness Verification Method

We performed inference with the TRUE model (Honovich et al., 2022) following previous work that employed the model for attribution verification (Gao et al., 2023a; Bohnet et al., 2022; Gao et al., 2023b; Chen et al., 2023): the model was prompted with a concatenation of a potential supporting document (preceded by the string "`premise:`") and an LLM-generated statement (preceded by the string "`hypothesis:`"). If the model's output is "`1`", then the generated statement is considered grounded in the supporting document, otherwise not.

C.1 Evaluation of the Verification Method

The validation of the model was carried out by a team of 5 annotators (consisting of the author and collaborators), each of whom was assigned a set of 20 instances of $(q, s, R(q), R_{\mathcal{C}}(s), g_{R(q)}(s), g_{R_{\mathcal{C}}(s)}(s))$, where

- q is a question,
- s is a statement belonging to the model-generated answer to q,
- $R(q)$ are the documents retrieved to augment the LLM generation,
- $R_{\mathcal{C}}(s)$ are the pre-training documents retrieved post-generation,
- $g_{R(q)}(s)$ and $g_{R_{\mathcal{C}}(s)}(s)$ are the groundedness predictions of the TRUE model with respect to each of the potential supporting document.

The 20 instances are sampled at random, making sure that an equal amount of instances comes from each of the categories: ungrounded, grounded in the pre-training documents only, grounded in the pre-generation retrieved documents only, and grounded in both types of documents. For each instance, an annotator determines whether the predictions of the groundedness model are correct or

Query q	**Generated Sentence** s	**Comment**
Who's the highest paid NBA player in 2017?	Kobe Bryant became the third player to achieve the milestone of earning $40-Million in the 2016–17 season, surpassing the record of 29-Million he had earned the previous season.	Kobe Bryant retired in 2016.
How many medals did Australia win in the 2000 Olympics?	The host nation had never entered in the Summer Olympics before, the Sydney Olympics being Australia's first appearance.	Australia's first appearance in the Olympics was in 1896.
What is the oldest company in the Dow Jones index?	In the early 2000s, the Dow Jones Industrial Average was composed of the following nine companies: The components of the DJIA have changed 52 times since its beginning on May 26, 1896. General Electric had the longest continuous presence on the index, beginning in 1907 and ending in 2018.	The last sentence is correct and grounded in a retrieved document, but the statement "In the early 2000s, the Dow Jones Industrial Average was composed of the following nine companies" is hallucinated.

Table 3: **Examples of ungrounded generations.** The generated content features hallucinated details that render the statements factually incorrect. The examples are obtained with Pythia 12B on ASQA.

not. A score of 1 was assigned by the annotator if the groundedness model accurately identified the supporting document subset (which could be none, in the case of ungrounded content) for the given LLM-generated statement. Conversely, a score of 0 was given if the model failed to correctly identify the supporting documents.

D Additional Experimental Details

Computing infrastructure. All experiments with models in the size range 70M-12B were carried out using a single 40GB Nvidia A100. Generations with MPT 30B and Falcon 40B were obtained using four 40GB Nvidia A100s, and with Falcon 180B using eight 40GB Nvidia A100s. The runtime for each model on each dataset was ≤12 hours.

Licenses. For our analyses we three QA datasets (ASQA, HotpotQA, and StrategyQA) and three pretraining corpora (the Pile, C4, and OLC). ASQA is available under the Apache 2.0 license, StrategyQA, the Pile, and OLC are available under the MIT license, HotpotQA is available under CC-BY, and C4 is released under the terms of ODC-BY.

E Additional Results

In Figure 8, we report the groundedness scores computed for the Pythia and Falcon models with different sizes on HotpotQA. We observe similar trends to the ASQA setting. Figure 9 illustrates the groundedness scores obtained with different Falcon and MPT models on StrategyQA.

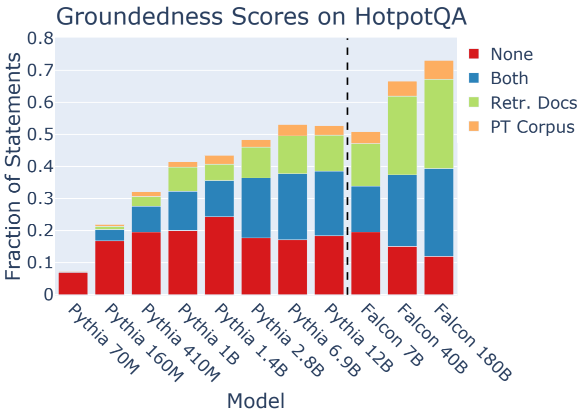

Figure 8: **Groundedness by size.** The results are consistent with the one obtained on ASQA.

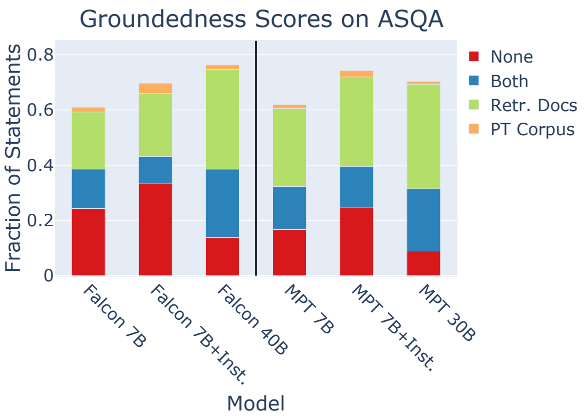

Figure 9: **Groundedness on StrategyQA.** As in previous figures, the size of each bar represents the fraction of generated sentences that belong to partially correct generations.

Instruction: Write an accurate, engaging, and concise answer for the given question, possibly using the provided search results (some of which might be irrelevant).

Question: Who played galen in planet of the apes?

Document [1](Title: Planet of the Apes): installment. Jacobs died on June 27, 1973, bringing an end to the APJAC Productions era of the "Planet of the Apes" franchise. Former Fox executive Stan Hough took over as producer for the television project, titled "Planet of the Apes". CBS picked up the series for its 1974 autumn lineup. Ron Harper and James Naughton played Alan Virdon and Peter Burke, two 20th-century American astronauts who pass through a time warp to a future where apes subjugate humans (unlike the original film, the humans can speak). Roddy McDowall returned to the franchise as Galen, a chimpanzee who joins the astronauts.
Document [2](Title: Planet of the Apes (1968 film)): chimpanzees: animal psychologist Zira (Kim Hunter) and surgeon Galen (Wright King). While unable to speak as his throat wound is healing, called "Bright Eyes" by Zira and placed with one of the captive primitive humans he later names "Nova", Taylor observes the enhanced society of talking apes and in a strict caste system: the gorillas being the military police, hunters and workers; the orangutans overseeing the affairs of government, science, and religion; and intellectual chimpanzees being mostly scientists. While their society is a theocracy similar to the beginnings of the human Industrial Era, the apes consider the primitive humans as
Document [3](Title: Planet of the Apes (1968 film)): Planet of the Apes (1968 film) Planet of the Apes is a 1968 American science fiction film directed by Franklin J. Schaffner. It stars Charlton Heston, Roddy McDowall, Kim Hunter, Maurice Evans, James Whitmore, James Daly and Linda Harrison. The screenplay by Michael Wilson and Rod Serling was loosely based on the 1963 French novel "La Planète des Singes" by Pierre Boulle. Jerry Goldsmith composed the groundbreaking avant-garde score. It was the first in a series of five films made between 1968 and 1973, all produced by Arthur P. Jacobs and released by 20th Century Fox. The film tells the

Answer: In the 1968 film Planet of the Apes, Galen was played by Wright King. And in the tv series Planet of the Apes, Galen was played by Roddy McDowall.

Instruction: Write an accurate, engaging, and concise answer for the given question, possibly using the provided search results (some of which might be irrelevant).

Question: Who has the highest goals in world football?

Document [1](Title: Argentina–Brazil football rivalry): "Football Player of the Century", by IFFHS International Federation of Football History and Statistics, 1999, "South America Football Player of the Century", by IFFHS International Federation of Football History and Statistics. Pelé's 1281 goals are recognized by FIFA as the highest total achieved by a professional footballer, although the Soccer Statistic Foundation (rssf) recognizes only 767 goals in official mode, occupying the third place after Josef Bican (805) and Romario (772). For his part, Maradona has been named the best soccer player in World Cup history both by The Times and FourFourTwo, publication that also rewarded him as the "Best
Document [2](Title: Godfrey Chitalu): have beaten Gerd Müller's record of 85 goals in a year, the Football Association of Zambia claimed that the world record actually pertained to Godfrey Chitalu who had scored 116 goals (possibly 117) during the 1972 calendar year and 107 during the 1972 season. The difference of goals is due to first 9 goals being scored before the season officially started. The Football Association of Zambia presented the evidence to FIFA but a spokesperson responded that they would ratify neither Lionel Messi's nor Chitalu's records as they do not keep statistical track of domestic competitions. Nonetheless, it could constitute the
Document [3](Title: Godfrey Chitalu): highest official tally claimed by a national football association. Chitalu made his international debut on 29 June 1968 in a friendly match against Uganda in Lusaka which Zambia won 2–1. He scored his first goal in a 2–2 draw against the same team five days later. Chitalu played a prominent role during the World Cup qualification matches against Sudan with Zambia being eliminated on a strange rule which was peculiar to Africa and favoured the team that won the second leg. Despite the aggregate score being tied at 6–6 after Zambia won the first leg 4–2 and lost the return

Answer:

Table 4: **Example of prompt** with one exemplar of question-documents-answer triple, used for ASQA.