“According to . . . ”: Prompting Language Models Improves Quoting from Pre-Training Data

Orion Weller*, **Marc Marone***, **Nathaniel Weir,**
Dawn Lawrie, Daniel Khashabi, Benjamin Van Durme
Johns Hopkins University
`{oweller2,mmarone1}@jhu.edu`

Abstract

Large Language Models (LLMs) may hallucinate and generate fake information, despite pre-training on factual data. Inspired by the journalistic device of *“according to sources”*, we propose *according-to* prompting: directing LLMs to ground responses against previously observed text. To quantify this grounding, we propose a novel evaluation metric (QUIP-Score) that measures the extent to which model-produced answers are directly found in underlying text corpora. We illustrate with experiments on three corpora (Wikipedia, PubMed, and the U.S. legal tax code) that these prompts improve grounding under our metrics, with the additional benefit of often improving end-task performance. Furthermore, prompts that ask the model to decrease grounding (or to ground to other corpora) indeed decrease QUIP-Score, indicating the ability of LLMs to increase or decrease grounded generations on request.[1]

1 Introduction

As the deployment of Large Language Models (LLMs) in real-world applications continues to grow, their tendency to generate false content (Ji et al., 2022) poses significant risks to downstream users. Recent work has attempted to address this issue by augmenting them with retrieval (Shuster et al., 2021; Sun et al., 2023; Borgeaud et al., 2022); however, these models still struggle with hallucination problems in practice (Liu et al., 2023).

This work explores the intriguing possibility of *steering LLMs by prompting* them to quote more of the curated sources of information they have memorized during pre-training, thereby reducing their tendency to generate false information. As illustrated in Figure 1, we explore whether adding phrases such as “According to Wikipedia” can guide LLMs to quote from Wikipedia, which is

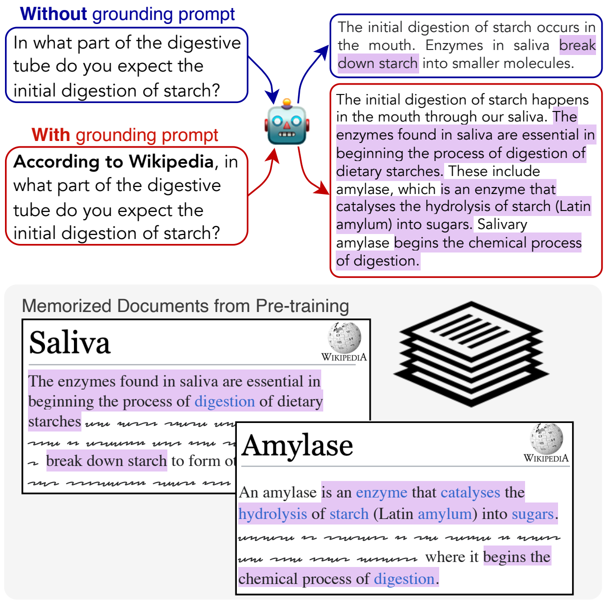

Figure 1: Prompting LLMs to respond with quotes directly from pre-training data (shown in purple). Prompting increases the proportion of quoted information.

presumably observed in the pre-training corpus. We find empirical evidence that this is attainable using current LLMs (both open and closed source).

Our study is inspired by two recent research areas. First, larger LLMs can be more effectively guided using natural language prompts (Ouyang et al., 2022; Wan et al., 2023; Ganguli et al., 2023). Second, as LLMs grow in size, their ability to remember facts and statements from pre-training improves (Kandpal et al., 2022; Tirumala et al., 2022; Carlini et al., 2023, 2020). Thus, we seek to steer LLMs to use their memorization for a positive purpose: producing more grounded outputs.

A key step in this study is quickly determining whether generated outputs overlap significantly with pre-training data; i.e., efficiently performing membership testing via a DATA PORTRAIT (Marone and Van Durme, 2023). We design a new metric called QUIP-Score, short for **Qu**oted **I**nformation **P**recision, which builds on DATA POR-

[1]We publicly release all code at `https://github.com/orionw/according-to`
* Authors contributed equally

Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 2288–2301
March 17-22, 2024

TRAITS and takes advantage of its speed and efficiency. QUIP-Score then calculates n-gram overlap, quantifying how much of a passage is formed of spans that are exactly contained in the corpus.

To illustrate *according-to* prompting, we perform experiments based on the task of open-domain question answering (ODQA), for which provenance-grounded answers are of particular importance. We collect human-authored prompts designed to steer generations toward information grounded in our target corpora (Wikipedia, PubMed, and the U.S. legal tax code). We observe that across all human-authored prompts, we can increase the amount of overlap with the chosen corpora by 5-105% while maintaining or even improving the downstream performance. We show results across numerous datasets and models, including both open- and closed-sourced LLMs.

Interestingly, we also observe the opposite phenomenon – it is possible to *discourage* LLMs from grounding via prompts that either discourage grounding or encourage grounding to other corpora. For example, we find this can decrease overlap with Wikipedia while lowering performance on downstream tasks that rely on Wikipedia content.

We conduct scaling experiments on different model sizes, which indicate that as size increases, so does the effectiveness of our proposed approach. This suggests that hallucinations may diminish with further scaling of instruction following LLMs.

In summary, we present *according-to* prompting, a simple and effective approach to improving an LLMs' ability to generate more factual information. Additionally, we introduce QUIP-Score, an efficient metric for measuring groundedness of LLM generations against their pre-training corpus. We experiment with various prompting strategies across models, datasets, and scaling trends, and we find that *according-to* methods consistently improve groundedness under our introduced metric.

2 Related Work

Memorization in LLMs. Large language models have been observed to memorize their training data (Carlini et al., 2020; Chang et al., 2023, among others). This is problematic when web-scraped training data contains sensitive personal data or low-quality information sources (Dodge et al., 2021; Luccioni and Viviano, 2021). However, it can be beneficial for models to memorize content from *carefully curated and trusted corpora*, where careful de-duplication (Lee et al., 2022a) and curation strategies (Feng et al., 2022) can improve language model quality (Gao et al., 2020). Work on analyzing memorization has proposed measuring n-gram overlap against the first page of Google Search results as a proxy for memorization, using exact matches (Carlini et al., 2020) and BLEU (Levy et al., 2021).

We measure quoting (and thus, memorization in closed-book generation settings) building off of Marone and Van Durme (2023) who propose using membership testing tools that they label DATA PORTRAITS. As one implementation, they use a Bloom Filter (Bloom, 1970) for storing n-grams. We use this method for checking membership in a corpus as it allows us to build a fast, lightweight, and scalable metric for measuring quotation against large amounts of data (see Section 3.1 for details).

Hallucination and grounding. Numerous studies (De Cao et al., 2021; Li et al., 2022; Weller et al., 2023) have demonstrated that LLMs struggle with both hallucination and factuality, leading to frequent inaccuracies and outright falsehoods. Previous research has attempted to alleviate this problem in various ways, including retrieving grounded documents before generation (Sun et al., 2023; Borgeaud et al., 2022; Mallen et al., 2023; Weller et al., 2022), applying new decoding approaches (He et al., 2022), post hoc tuning of LLMs (Menick et al., 2022; Lee et al., 2022b), and analyzing the model's output training data (Han and Tsvetkov, 2022; Park et al., 2023). Crucially, these works have a common thread: showing that grounding LLM generations results in fewer hallucinations (Lazaridou et al., 2022; Andriopoulos and Pouwelse, 2023). Our work focuses on a subset of grounding, *quoting*, and is driven by the simple premise that *anything quoted is grounded and not hallucinated.* Our work therefore builds off the established research and is complementary to it, as we investigate a novel yet straightforward approach to steer LLMs towards more factual responses.

Attribution. A related line of work is attribution of generated text to their sources (Rashkin et al., 2021; Bohnet et al., 2022). Our work is related to this literature in that, our approach allows provable attribution to macro-level sources of information, such as Wikipedia or medical articles. However, we do not focus on offering any fine-grained attribution to the originating source documents. Given these distinctions our focus here is different from –and complementary to– the attribution literature.

LLM Steerability via prompting. The larger LMs become, the easier they are to steer with natural language prompts (Kandpal et al., 2022; Carlini et al., 2023; Mishra et al., 2022a; Srivastava et al., 2023). Several works (Mishra et al., 2022b; Chung et al., 2022; Wang et al., 2022b; Wan et al., 2023) have shown that larger instruction-tuned models are more easily steered than smaller and non-instruction-tuned models. This is desirable in our setting, as we seek to use these capabilities of LLMs for a novel application of steerability: *quoting more from a given corpus.*

Improving LLMs through prompting. Much recent work has focused on improving LLM performance on various benchmarks by improving the prompt given to the model. A sub-genre of these works includes those that ask the model to produce text before generating the answer, such as Chain-of-Thought (Wei et al., 2022) or Recitation-based Generation (Sun et al., 2022). We differ from these works by generating the answer first, then the explanation, indicating that our performance gains are not due to the same phenomena. Furthermore, our paper's focus is on improving LLM's ability to quote, rather than improving end-task performance.

3 Methodology

Defining Grounding There are many definitions of *grounding* in the community (Bohnet et al., 2022; Mallen et al., 2023). While acknowledging the broad scope of the term, we adopt a narrow definition: we call generated text *grounded* with respect to a corpus if it is an exact quotation from the corpus. This is more stringent than some definitions because it does not count semantic grounding, e.g. when lexical forms do not match; however, quotation is one form of grounding that is intuitive and simple to measure.[2] Hence, we use *quoting* and *grounded* interchangeably.

[2]We leave it to future work to expand our metric to the semantic grounding case, as semantic grounding (e.g. finding paraphrases) while matching the generations over an entire corpus is non-trivial; using retrieval systems biases the model towards lexical match (even for dense retrieval, c.f. MacAvaney et al. (2022)) and existing work in attribution/grounding does not scale to allow grounding to numerous (2+) passages.

3.1 QUIP-Score: Measuring Grounding to Pre-Training Data

In order to understand grounding and quoting from model pre-training data, we need a metric to measure quoting. An intuitive approach is to use an n-gram measure, which can compare n-grams found in an LLM's generation to those in a corpus. Such a quotation metric must be efficient to scale to large reference corpora.

Problems with existing N-gram metrics Existing n-gram metrics like BLEU or ROUGE store counts of n-grams from the references. However, storing counts requires the use of data structures like a conventional hashtable, which is computationally difficult for a large corpus like Wikipedia. We estimate naively scaling `sacrebleu` (Post, 2018) to use Wikipedia as a reference would consume ~1.5 TB of RAM (Appendix C).

QUIP-Score To enable efficient measurement of quoting from pre-training data, we start with a Bloom filter-based DATA PORTRAIT (Marone and Van Durme, 2023), which allows for both faster and more memory efficient boolean membership queries than allowed by methods that use a hashtable to store counts. The Bloom filter approach enables one-time indexing of a large corpus with constant time lookups.

We define our new metric, QUIP-Score, as the character n-gram precision of overlap between generated output and the pre-training corpus.[3] More formally, for generation Y and text corpus C:

$$\text{QUIP}(Y;C) = \frac{\sum_{\text{gram}_n \in Y} \mathbb{1}_C(\text{gram}_n)}{|\text{gram}_n \in Y|},$$

where $\mathbb{1}(.)$ is an indicator function implemented with the DATA PORTRAIT: 1 if $\text{gram}_n \in C$ else 0. Thus, a score of 0.5 would indicate that 50% of the generated text n-grams are found in the pre-training corpus. We macro-average this quantity over a set of generations to obtain a single performance number for a given test dataset.

QUIP-Score Implementation We build the DATA PORTRAIT on the version of Wikipedia included in the Pile,[4] as it allows for us to exactly test the pre-training data included in many mod-

[3]QUIP scores are not comparable across datasets, as they are specific to a given corpus. This is acceptable for our experiments that compare generations against one corpus.

[4]wikipedia/20200301.en

els like GPT-J[5] (See §6 for experiments applying QUIP-Score to other corpora). We use character-based n-grams as opposed to token-based, as different models have different tokenization schemes. Furthermore, character-based n-gram metrics have widespread usage in machine translation with metrics like chrF/chrF++ (Popović, 2015, 2017). We chose 25 character grams for the sketch[6] (approximately 5 words) as we found it empirically gave meaningful results (neither too small nor too large an n-gram). Note that because the DATA PORTRAIT checks for exact matches it is sensitive to orthographic variation (e.g. case, whitespace), We view QUIP-Score as a lower bound on actual quoting performance.

3.2 Validity of QUIP-Score

As QUIP-Score is an n-gram metric, it inherits many of the same qualities of established metrics like BLEU and ROUGE. Further, many previous works have established the connection between higher amounts of grounding and fewer hallucinations (§2). Building upon these previous studies, we establish that QUIP-Score (1) accurately measures quoting like other n-gram metrics and (2) is correlated with fewer hallucinations.

We first conduct a straightforward experiment: what is the QUIP-Score when measuring entirely quoted documents (e.g. exact Wikipedia pages) vs documents that are not necessarily quotes (e.g. from the Pile)? We randomly sample 1000 documents from each. We find that the average QUIP-Score for Wikipedia documents is 99.9%[7] with a standard deviation of 0.1% while on the Pile it is 17.0% ± 0.8%. Thus we can see that QUIP-Score correctly measures full quotations and that random text has approximately 17% QUIP-Score.

Next, we consider partial, contextual quotations as found in LLM generations from NQ. We bin generations by QUIP-Score ranges, sampling 50 from each bin. We then conduct two manual analyses: (1) how much of the generations are a quotation (none, some, majority, or all/nearly all) and (2) whether the generation is a hallucination (using gold provenances and answers, plus Google Search when unsure). Table 1 shows that as QUIP-Score increases, the amount of quotations increases and the amount of hallucinations decreases.

QUIP-Score	None	Some	Major.	All	Halluc.
0.0 – 0.25	12%	76%	12%	0%	20%
0.25 – 0.5	0%	16%	84%	0%	22%
0.5 – 0.75	0%	0%	80%	20%	12%
0.75 – 1.0	0%	0%	48%	52%	6%

Table 1: Random sampled generations from NQ, binned by QUIP-Score. **As QUIP-Score increases, quoting increases and hallucinations decrease.** *Major.* stands for Majority, while *Halluc.* stands for Hallucination %.

We do not expect these results to be surprising, as they have been demonstrated by a large amount of literature on n-gram metrics (Belz and Reiter, 2006; Reiter and Belz, 2009; Popović, 2015), and by the grounding and hallucination literature (Lazaridou et al., 2022; Borgeaud et al., 2022; Andriopoulos and Pouwelse, 2023). However, this analysis empirically demonstrates that using quoting for grounding and QUIP-Score as the n-gram metric retains these desired properties.

4 Grounding via *according-to* Prompting

The previous results show 1) that we can efficiently measure quotation rate and 2) that more quotations correlate with fewer hallucinations. Next, we seek to improve knowledge grounding by causing LLMs to quote directly from trusted resources seen during training.[8] We hope to access *helpful* memorized content: strings copied from high-quality or trusted documents. We induce this behavior by taking a normal task prompt (e.g. an ODQA question) and appending an instructional phrase that encourages grounding such as "Respond by using information from Wikipedia in your response".[9] We call this strategy *according-to* prompting. Our experiments measure the change in QUIP-Score of generations from a *according-to* prompt vs one without the extra instruction (i.e. a null prompt).

To verify that prompts can both increase and decrease grounding, we also include prompts that are anti-grounding (e.g. "Respond by using information from [another source] in your response" or "Respond without using any information from Wikipedia.") This allows us to test the hypothesis that models can ground (or not ground) to a

[5]Note, for several models evaluated here (e.g. OpenAI models) the exact Wikipedia version trained on is unknown.

[6]Not having multiple n-gram sizes like BLEU typically does allows us to significantly reduce memory consumption and had similar results to averaging across sizes.

[7]QUIP-Score is 99.9 due to a single very short sampled document, where length < n-gram size

[8]Since we want to know what the LLM recalls on its own, we specifically do not use any retrieval models.

[9]We tried appending, prepending, and their combinations in early experiments and found that appending the grounding/anti-grounding prompts performed the best.

given corpus when asked because of the semantic meaning of the prompt, rather than the length of the prompt. As prompting is notoriously brittle (e.g. changing the phrasing can affect the results) we provide a number of grounding and anti-grounding prompts to test whether these prompts provide consistent gains or are merely prompting artifacts (see Table 2 for the list of prompts used).

4.1 Datasets

We use a variety of datasets to test if LLMs are consistent and to check whether grounding affects the end-task performance of a given dataset. To best measure the grounding of the output however, the model generations must be long enough to have many n-grams that can be measured. Thus, we test on long-form question answering (QA), and for datasets that do not lend themselves well to long-form output (e.g. short-form QA) we ask the models to generate both the answer and a corresponding explanation whose n-grams can be measured.

Note that our purpose is not to improve state-of-the-art performance on these tasks, as our main research question is to analyze the grounding of model outputs. However, we note that *according-to* prompting often achieves competitive or improved performance compared to other prompting baselines, as it naturally correlates with the ability to answer questions from the grounded material.

We use the following datasets, each of which targets factual knowledge in Wikipedia: **ELI5** (Fan et al., 2019) (the KILT Petroni et al. (2021b) version), **Natural Questions** (Kwiatkowski et al., 2019), **TriviaQA (TQA)** (Joshi et al., 2017), and **HotpotQA** (Yang et al., 2018). These datasets comprise a mixture of short- and long-form plus single- and multi-hop QA. §A provides further details.

4.2 Models and Prompting

We test a wide array of models in our experiments including most OpenAI models (Wang et al., 2023), T5-based models (T5 adapted to language modeling, Raffel et al. 2020; Lester et al. 2021 and FLAN-T5 Chung et al. 2022), GPT-J instruction tuned[10] (Wang and Komatsuzaki, 2021), and Koala (Geng et al., 2023) (a Llama variant, Touvron et al. 2023). By doing so, we provide (1) results on both open and closed-source models, (2) results for models using many variations of instruction-tuning data, and (3) models ranging from 220 million parameters to 175B models. Note that our experiments consist solely of providing prompts to the models and do not include fine-tuning (as the goal is to see what these models can do zero-shot).

For short-form QA datasets, we prompt models to produce an answer plus an explanation, then measure QUIP-Score of the latter. We found smaller models (e.g. < 15B parameters) were not able to follow instructions to provide both answer and explanation in a parseable format from just one prompt. Thus, we do two-step prompting with them, first for the answer, then for the explanation (and appending the grounding prompt, if used). §B.2 provides prompting details and full text of the prompts used.

5 Results

We first analyze a wide range of *according-to* prompts on ChatGPT. We then test the null prompt and the best performing *according-to* prompt on a variety of other models for further analysis. Table 2 shows results for different prompts using ChatGPT. There is a clear trend under which all *according-to* prompts perform similarly or improve upon QUIP-Score compared to the null. QUIP-Scores for the anti-grounding prompts are the same or worse than the null prompt (i.e. no additional text) and significantly worse than the *according-to* prompts.

Surprisingly, we find that *according-to* prompts also perform similarly, and sometimes even better than, the null prompt on end task performance (e.g. up to a 6% improvement on NQ, 2.5% on HotpotQA). This is not the case for ROUGE-L on ELI5, as that metric measures lexical similarity to Reddit, rather than similarity to Wikipedia.

We use these results on ChatGPT to inform our next experiments, using the null prompt and the best grounding prompt ("Respond to this question using only information that can be attributed to Wikipedia") in our future experiments due to cost.

5.1 Results from Other Models

We show the relative difference of the grounding prompt over the null prompt for more models in Table 3, which further confirms our findings (for the absolute instead of relative numbers, see Appendix B.2). For example, using the grounding prompt with Text-Davinci-003 improves over the null prompt by around 15% QUIP-Score and 5-20% for the specific task. For all models evaluated, the grounding prompt improves in both end-task performance and QUIP-Score by 5-105%.

[10] https://huggingface.co/nlpcloud/instruct-gpt-j-fp16

	Prompt (appended after the question)	TQA QUIP	TQA EM	NQ QUIP	NQ EM	Hotpot QUIP	Hotpot F1	ELI5 QUIP	ELI5 R-L
	∅ (no additional prompt)	31.6	77.8	32.8	32.9	28.3	35.7	24.1	22.7
grounding prompts	"Based on evidence from Wikipedia:"	31.1	77.3	32.8	34.0	28.1	35.9	26.3	22.3
	"As an expert editor for Wikipedia, I am confident in the following answer."	31.7	73.2	33.0	30.2	28.7	35.3	25.5	22.7
	"I found some results for that on Wikipedia. Here's a direct quote:"	31.7	70.1	33.8	27.6	28.1	33.1	27.2	21.0
	"Reference Wikipedia when answering the following question."	32.8	75.9	34.6	34.4	28.9	35.9	25.7	22.0
	"Answer according to Wikipedia."	33.6	78.8	34.3	34.8	29.2	36.6	26.5	21.7
	"Go to https://www.wikipedia.org and find direct quotes to answer the question. Response: ""	34.5	72.7	32.9	31.7	30.4	35.5	25.8	20.4
	"Respond by using information from Wikipedia in your response."	34.9	76.3	35.3	32.9	29.9	36.1	26.3	21.9
	"Respond to this question using only information that can be attributed to Wikipedia."	35.7	76.6	37.0	33.9	30.4	36.2	28.0	21.5
anti-grounding	"Respond by using information from Reddit in your response."	26.1	75.8	26.5	31.6	22.4	35.0	21.9	22.2
	"Respond by using information from Github in your response."	26.7	74.3	28.2	32.4	23.2	33.7	24.3	22.0
	"Respond without using any information from Wikipedia in your response."	30.4	76.9	32.0	32.0	26.8	32.9	24.7	22.1
	Zero-Shot No-Retrieval SOTA	-	*68.2*	-	*24.9*	-	*44.6*	-	*22.7*
	Retreival-Augmented SOTA	-	*89.4*	-	*60.4*	-	*51.4*	-	*26.5*

Table 2: Impact of various prompts on the grounding (QUIP-Score) and performance scores, using ChatGPT (§5). The top row is the null prompt (no additional prompt other than the question), the middle section includes grounding prompts, and the last section includes anti-grounding prompts. We find that **grounding prompts generally improve the QUIP-Score while anti-grounding prompts generally reduce QUIP-Score**. Colored cells indicate changes (gains, losses, or the same) relative to the null row. ELI5 ROUGE-L (R-L) is based on similarity to Reddit rather than Wikipedia. See §B.1 for sources of SOTA results.

Model	TQA QUIP	TQA EM	NQ QUIP	NQ EM	Hotpot QUIP	Hotpot F1	ELI5 QUIP	ELI5 R-L
Text-Davinci-003	+14.7%	+5.3%	+14.7%	+20.6%	+14.4%	+7.2%	+16.5%	-3.8%
GPT-4	-	-	-	-	-	-	+17.6%	-2.3%
GPT-J Instruct	+12.1%	-	+15.2%	-	+13.9%	-	+18.1%	-2.5%
Koala 7B	+5.1%	-	+6.3%	-	+5.0%	-	+35.5%	+14.6%
FLAN-T5 XXL	+43.3%	-	+41.5%	-	+20.7%	-	+105.2%	+48.4%

Table 3: Percent improvement of *according-to* over null prompt. **The *according-to* prompt improves performance in nearly every dataset and metric by 5-15%**. We omit EM/F1 scores of smaller models for which our prompting method yield the same answer for grounding and null (§4.2). Due to cost, we only evaluate GPT-4 on ELI5.

Thus, our findings hold for a wide variety of models and model sizes – even when prompts are not tuned for the specific model being prompted, indicating the generality of our approach.

5.2 Impact of Model Size

Does model size impact their ability to quote from their pre-training data? We answer this question using QUIP-Score in Figure 3, which shows that smaller models perform the same (for FLAN-T5 models) or worse (for OpenAI models) with a grounding prompt as opposed to the null prompt. However, larger models perform significantly better with the grounding prompt as opposed to the null prompt, for both OpenAI models and FLAN-T5 models. We can conclude that a **model's ability to quote from its pre-training data improves with size**.

5.3 Impact of Entity Popularity

Another potential factor influencing generation of memorized content is the popularity of the entities mentioned in a question (Kandpal et al., 2022; Carlini et al., 2023). Previous work has shown that entity co-occurrence (as measured by the number of times in the pre-training set that the entities in the question and in the answer co-occur in the same passage) is strongly correlated with task performance (Kandpal et al., 2022). We use their code and data (from the Pile) to explore whether QUIP-Score correlates with co-occurrence frequency.

Due to the imbalance between co-occurrence counts, we sample 400 instances (or as many as available) from each dataset and co-occurrence frequency bin.[11] We measure the QUIP-Score on these instances using the output generations from ChatGPT on both grounding and null prompts.

Figure 2 shows that QA entity popularity is positively correlated with QUIP-Score for both grounding and null prompts, more so for grounding. We find that the model better recalls information from Wikipedia when QA entities frequently co-occur.

[11] See Kandpal et al. (2022) for frequency bin design details.

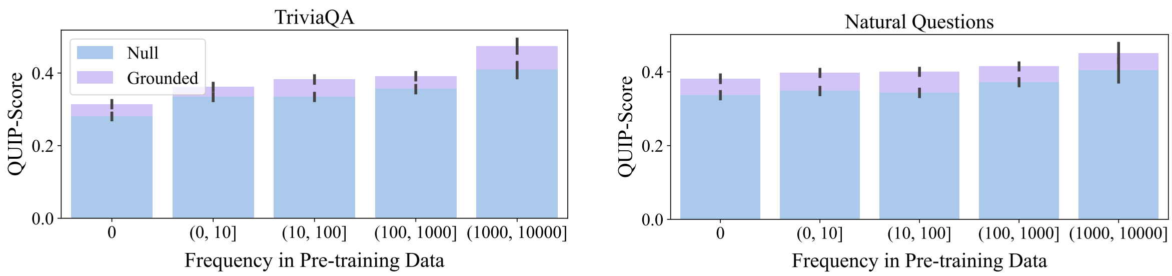

Figure 2: Impact of entity popularity on QUIP-Scores, showing that **models are better able to quote pre-training text about popular entities**. The x-axis shows how many times the given entity relationship was found co-occurring in pre-training data. Bars indicate 1 standard error. We use the ranges following (Kandpal et al., 2022).

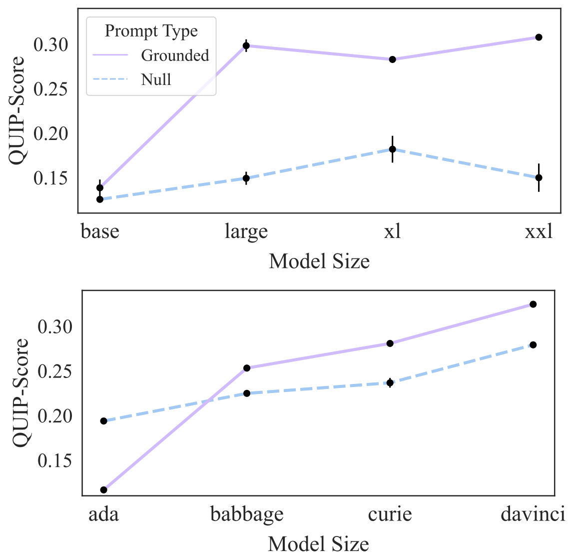

Figure 3: Model size vs QUIP-Score performance using FLAN-T5 (top) and OpenAI (bottom) models. **As model scale increases, so does performance**. At smaller model sizes, the grounding prompt is not more effective than the null prompt, but gains efficacy with model size. Error bars indicate 1 standard error.

5.4 Impact of Instruction Tuning

One potential reason for why these models can recall their pre-training data on request is a better capability to instruction-follow. We test this hypothesis in Figure 4 that compares T5-11B compared to FLAN-T5-11B.

We find that instruction-tuning does help, as the QUIP-Scores for T5-v1.1-Adapt are similar between grounding and null prompts, while the FLAN-T5 model has a large difference between the null and grounding prompt (roughly 2x better).

5.5 Qualitative Examples

Figure 5 shows example outputs from a variety of models. When queried with *according-to* prompts, the model generates much larger chunks of texts that occur in Wikipedia (shown in purple).

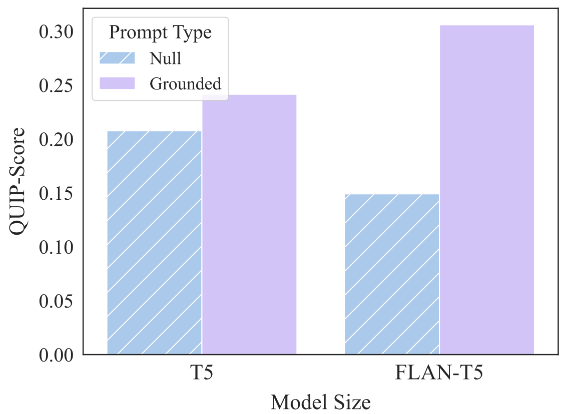

Figure 4: Comparing instructed-tuned FLAN-T5 XXL to non-instruction tuned T5-v1.1-Adapt XXL. Note that **grounding has a larger impact on instruction-tuned models as compared to non-instruction tuned**.

It is important to note that although the text may be *grounded*, it does not mean the output generation is *correct* with respect to the question. For example, the TriviaQA example shows that both models predicted the incorrect quote for Smokey the Bear, although the *according-to* explanation was better grounded in Wikipedia.

6 Grounding to Other Corpora

In the previous experiments we've shown that LLMs can ground to Wikipedia upon request. However, are *according-to* prompts limited to the Wikipedia domain only? To answer this question we build two more DATA PORTRAITS, one on PubMed articles and one with the U.S. legal tax code applying to tax year 2022. We then evaluate on datasets where having such background knowledge would be helpful and evaluate both the downstream task performance and QUIP-Score, similar to the previous Wikipedia experiments.

Datasets. We use datasets in the MultiMedQA benchmark suite (Singhal et al., 2022) to test

Prompt	Generated Text
1. Question (ELI5)	**Why is it that every time we put something down our throat toothbrush, finger etc. we start to gag but when we but food and water down our throat we don't gag at all?,**
Null Prompt (ChatGPT)	This is because our gag reflex is triggered by objects that touch the back of our throat and activate sensory receptors called the pharyngeal receptors. When we eat or drink, the muscles in our throat automatically contract and move food and liquid down to the stomach, bypassing the pharyngeal receptors and avoiding the gag reflex. However, when we put non-food objects like a toothbrush or finger in our throat, they can activate these receptors and trigger the gag reflex.
Grounded Prompt (ChatGPT)	The gag reflex is a natural protective mechanism of the body that can be triggered by stimulation of the soft palate, the back of the tongue or the back of the throat. It is a reflex contraction of the muscles of the back of the throat that can cause vomiting or choking. The reflex can be more easily triggered by foreign objects, such as toothbrushes or fingers, that are not natural to the body, compared to food and water which are familiar substances that we regularly ingest without triggering the reflex. Additionally, the shape and texture of the foreign object may also contribute to triggering the gag reflex.
2. Question (NQ)	**In what part of the digestive tube do you expect the initial digestion of starch?**
Null Prompt (ChatGPT)	The initial digestion of starch occurs in the mouth; enzymes in saliva break down starch into smaller molecules.
Grounded Prompt (ChatGPT)	The initial digestion of starch happens in the mouth; Salivary amylase, an enzyme found in the mouth, is responsible for breaking down starch into simpler sugars.
3. Question (ELI5)	**Why do towels that feel so rough on the skin dry you so much better, but towels that feel soft don't dry well at all?**
Null Prompt (Davinci)	Towels that feel rough on the skin are usually made of a coarser fabric, such as terry cloth, which is made of loops of cotton yarn. This fabric is more absorbent than softer fabrics, such as velour or microfiber, which are made of finer fibers. The loops of terry cloth fabric create more surface area for the towel to absorb water, which is why it is more effective at drying than softer towels.
Grounded Prompt (Davinci)	Towels that feel rough on the skin are usually made of a material with a higher absorbency, such as cotton or linen. These materials are able to absorb more water than softer materials, such as microfiber or polyester. The absorbency of the material allows it to draw moisture away from the skin more effectively, resulting in better drying. Soft towels, on the other hand, are usually made of materials with lower absorbency, which means they are not able to absorb as much water and therefore do not dry as well.
4. Question (TriviaQA)	**According to Smokey The Bear, Remember... only you can what?**
Null Prompt (GPT-J-Instruct)	"Only you can see me"; Because fire requires a fuel source to burn, and smoke is the byproduct of a fire's burning process. Therefore, since smoke is created whenever a fire burns, the presence of smoke indicates that a fire is burning somewhere.
Grounded Prompt (GPT-J-Instruct)	"Only you can remember"; According to Smokey The Bear, memory is an internal mental process that is unique to each person. It is the ability to remember past experiences, along with the ability to recognize and interpret the information that is stored in our memories. This explains why, despite the fact that memory techniques vary between individuals, memory is a universal quality that all humans share. Therefore, the answer to the question is "yes".
5. Question (HotpotQA)	**Which magazine is based out of more countries, Adventist World or Baseball Digest?**
Null Prompt (FLAN-T5 XXL)	adventist world; Adventist World is a magazine that is based out of many countries.
Grounded Prompt (FLAN-T5 XXL)	adventist world; Adventist World is a monthly magazine published by the Seventh-day Adventist Church. It is based in the United States, Canada, and many other countries.

Figure 5: Example generations from various considered models. Purple text was found in Wikipedia. Note that for non-ELI5 datasets, models were prompted to generate the answer, a semicolon, and then the explanation (see §4.2). Note that better grounding to Wikipedia does not always imply correct answers (see Question 4).

Dataset	Null QUIP	Null EM	Grounded QUIP	Grounded EM
According to PubMed...				
PubMedQA	54.1	48.2	59.6↑ (**+5.5**)	49.6↑ (**+1.4**)
MedQA	45.1	53.3	45.9↑ (**+0.8**)	54.0↑ (**+0.7**)
MedicationQA	36.7	N/A	39.6↑ (**+2.9**)	N/A
According to the U.S. Tax Code...				
SARA	4.4	52.0	13.3↑ (**+8.9**)	55.0↑ (**+3.0**)

Table 4: Results with ChatGPT using *according-to* prompts for PubMed (top) and the U.S. legal tax code (bottom). *according-to* prompts consistently improve quoting on the non-Wikipedia domains while maintaining task performance. MedicationQA does not have an automated evaluation metric, so only QUIP is reported.

grounding to PubMed: PubMedQA (Jin et al., 2019) a reading comprehension task over PubMed abstracts, MedQA (Jin et al., 2020) consisting of multiple-choice questions from the US Medical Licensing Exam, and MedicationQA (Abacha et al., 2019) which asks open-domain questions about patient medications. Although these last two are not directly sourced from PubMed, they contain information that is likely to be found in it. Note that we do not give the model the abstract as typically done in PubMedQA, but instead evaluate closed-book in order to measure quotes from model parameters.

In the legal domain, we use the SARA dataset (Holzenberger et al., 2020) consisting of tax cases to be evaluated using natural language inference.[12]

Results. The results in Table 4 with ChatGPT show that *according-to* prompts improve end-task performance and QUIP-Scores. On SARA, QUIP-Scores more than triple, while also minorly increas-

[12] As these datasets have different formats, e.g. NLI and multiple choice, we change the prompt slightly to accommodate them (Appendix D). We use the test set for all datasets.

ing performance. In the medical domain, grounding to PubMed improves performance slightly as well, and improves QUIP scores on all datasets.

7 Discussion and Future Implications

Our results strongly suggest that LLMs can be steered via prompting to increase the amount by which they quote human-authored sources in their training data. This finding has strong implications not just for our considered tasks, but also for a wide array of other task spaces in which provenance grounding is important.

We note that our *according-to* prompting strategy is orthogonal to other directions in LLM grounding, including using retrieval augmentation, and as *according-to* prompting is simple and generally increases both grounding and task performance we would encourage future research to try our approach in tandem.

8 Conclusion

Large language models struggle with hallucination, or generating incorrect information, despite the large amount of factual pre-training data they were trained on. To help alleviate this problem, we proposed *according-to* prompts, asking language models to ground their output to their pre-training corpus. To quantify the extent to which models achieve this goal, we introduced a new metric, QUIP-Score, that efficiently and quickly measures the percent of the model's generation that exists as exact quotes in the pre-training corpus. We showed that prompting models with *according-to* prompts greatly improves the QUIP-Score while anti-grounding prompts reduces the QUIP-Score, across a variety of domains and corpora. Our analysis also shows that QUIP-Score increases with the popularity of the entity in the question and the model size. We hope that this work brings more attention to the positive aspects of LLM memorization and encourages more work into understanding LLM grounding to their pre-training data.

9 Limitations

Our proposed metric only accounts for exact lexical match and will miss other types of grounded statements - thus we view QUIP-Score as a lower bound on grounding where grounding is defined only by quoting from source material. QUIP-Score is also DATA PORTRAIT specific, as the amount of n-grams in the portrait affect the scores. We leave it to future work to generalize this metric, as our work focuses on using it to compare two prompts with the same Portrait.

We also recognize the possibility of a discrepancy between the pre-training data of private models like ChatGPT and the Wikipedia version we use for analysis, due to limited information on their pre-training. However, this might not be a significant concern, as although Wikipedia is not completely static, a substantial part of the information in this knowledge source remains consistent over a short span of years. Furthermore, our results with ChatGPT are similar compared with models for which we do have the exact pre-training data (like GPT-J).

Acknowledgements

This work has been supported in part by the U.S. National Science Foundation under grant No. 2204926. OW and NW are also supported by the National Science Foundation Graduate Research Fellowship Program.

References

Asma Ben Abacha, Yassine Mrabet, Mark Sharp, Travis R Goodwin, Sonya E Shooshan, and Dina Demner-Fushman. 2019. Bridging the gap between consumers' medication questions and trusted answers. In *MedInfo*, pages 25–29.

Konstantinos Andriopoulos and Johan A. Pouwelse. 2023. Augmenting llms with knowledge: A survey on hallucination prevention.

Anja Belz and Ehud Reiter. 2006. Comparing automatic and human evaluation of nlg systems. In *11th conference of the european chapter of the association for computational linguistics*, pages 313–320.

Burton H. Bloom. 1970. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426.

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*.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving language models by retrieving from trillions of tokens. In *International Conference on Machine Learning* (ICML).

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2023. Quantifying memorization across neural language models. In *International Conference on Learning Representations* (ICLR).

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Xiaodong Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2020. Extracting training data from large language models. In *USENIX Security Symposium* (USENIX).

Kent K Chang, Mackenzie Cramer, Sandeep Soni, and David Bamman. 2023. Speak, memory: An archaeology of books known to chatgpt/gpt-4. *arXiv preprint arXiv:2305.00118*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jesse Dodge, Maarten Sap, Ana Marasovic, William Agnew, Gabriel Ilharco, Dirk Groeneveld, and Matt Gardner. 2021. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In *Conference on Empirical Methods in Natural Language Processing* (EMNLP).

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long Form Question Answering. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

Yukun Feng, Patrick Xia, Benjamin Van Durme, and João Sedoc. 2022. Automatic document selection for efficient encoder pretraining. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9522–9530, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Deep Ganguli, Amanda Askell, Nicholas Schiefer, Thomas Liao, Kamilė Lukošiūtė, Anna Chen, Anna Goldie, Azalia Mirhoseini, Catherine Olsson, Danny Hernandez, et al. 2023. The capacity for moral self-correction in large language models. *arXiv preprint arXiv:2302.07459*.

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

Xinyang Geng, Arnav Gudibande, Hao Liu, Eric Wallace, Pieter Abbeel, Sergey Levine, and Dawn Song. 2023. Koala: A dialogue model for academic research. Blog post.

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*.

Hangfeng He, Hongming Zhang, and Dan Roth. 2022. Rethinking with retrieval: Faithful large language model inference. *arXiv preprint arXiv:2301.00303*.

Nils Holzenberger, Andrew Blair-Stanek, and Benjamin Van Durme. 2020. A dataset for statutory reasoning in tax law entailment and question answering. *arXiv preprint arXiv:2005.05257*.

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. *arXiv preprint arXiv*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2022. Survey of hallucination in natural language generation. *ACM Computing Surveys*.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2020. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *arXiv preprint arXiv:2009.13081*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. Pubmedqa: A dataset for biomedical research question answering. In *Conference on Empirical Methods in Natural Language Processing* (EMNLP).

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2022. Large language models struggle to learn long-tail knowledge. *arXiv preprint arXiv:2211.08411*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural Questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics* (TACL), 7:453–466.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022a. Deduplicating training data makes language models better. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

Nayeon Lee, Wei Ping, Peng Xu, Mostofa Patwary, Pascale N Fung, Mohammad Shoeybi, and Bryan Catanzaro. 2022b. Factuality enhanced language models for open-ended text generation. *Advances in Neural Information Processing Systems* (NeurIPS).

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Conference on Empirical Methods in Natural Language Processing* (EMNLP).

Sharon Levy, Michael Saxon, and William Yang Wang. 2021. Investigating memorization of conspiracy theories in text generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4718–4729, Online. Association for Computational Linguistics.

Shaobo Li, Xiaoguang Li, Lifeng Shang, Zhenhua Dong, Chengjie Sun, Bingquan Liu, Zhenzhou Ji, Xin Jiang, and Qun Liu. 2022. How pre-trained language models capture factual knowledge? a causal-inspired analysis. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

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

Alexandra Sasha Luccioni and Joseph D Viviano. 2021. What's in the box? a preliminary analysis of undesirable content in the common crawl corpus. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

Sean MacAvaney, Sergey Feldman, Nazli Goharian, Doug Downey, and Arman Cohan. 2022. Abnirml: analyzing the behavior of neural ir models. *Transactions of the Association for Computational Linguistics*, 10:224–239.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. 2023. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

Marc Marone and Benjamin Van Durme. 2023. Data portraits: Recording foundation model training data. *arXiv preprint arXiv:2303.03919*.

Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, et al. 2022. Teaching language models to support answers with verified quotes. *arXiv preprint arXiv:2203.11147*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2022a. Reframing instructional prompts to gptk's language. In *Annual Meeting of the Association for Computational Linguistics* (ACL) - *Findings*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022b. Cross-Task Generalization via Natural Language Crowdsourcing Instructions. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training Language Models to Follow Instructions with Human Feedback. In *Advances in Neural Information Processing Systems* (NeurIPS).

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. 2023. TRAK: Attributing Model Behavior at Scale. *arXiv preprint arXiv:2303.14186*.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021a. KILT: A benchmark for knowledge intensive language tasks. In *Conference of the North American Chapter of the Association for Computational Linguistics* (NAACL).

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021b. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram f-score for automatic mt evaluation. In *Conference on Machine Translation* (WMT).

Maja Popović. 2017. chrF++: words helping character n-grams. In *Conference on Machine Translation* (WMT).

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

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. *Journal of Machine Learning Research* (JMLR).

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. 2021. Measuring attribution in natural language generation models. *arXiv preprint arXiv:2112.12870*.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings-Conference on Empirical Methods in Natural Language Processing* (EMNLP).

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2022. Large language models encode clinical knowledge. *arXiv preprint arXiv:2212.13138*.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research* (TMLR).

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 - Annual Meeting of the Association for Computational Linguistics* (ACL).

Weiwei Sun, Zhengliang Shi, Shen Gao, Pengjie Ren, Maarten de Rijke, and Zhaochun Ren. 2023. Contrastive learning reduces hallucination in conversations. In *Conference on Artificial Intelligence* (AAAI).

Zhiqing Sun, Xuezhi Wang, Yi Tay, Yiming Yang, and Denny Zhou. 2022. Recitation-augmented language models. *arXiv preprint arXiv:2210.01296*.

Kushal Tirumala, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. *Advances in Neural Information Processing Systems*, 35:38274–38290.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Alexander Wan, Eric Wallace, Sheng Shen, and Dan Klein. 2023. Poisoning language models during instruction tuning. *arXiv preprint arXiv:2305.00944*.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. `https://github.com/kingoflolz/mesh-transformer-jax`.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. 2022a. Rationale-augmented ensembles in language models. *arXiv preprint arXiv:2207.00747*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-Instruct: Aligning Language Model with Self Generated Instructions. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Maitreya Patel, Kuntal Kumar Pal, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Shailaja Keyur Sampat, Savan Doshi, Siddhartha Mishra, Sujan Reddy, Sumanta Patro, Tanay Dixit, Xudong Shen, Chitta Baral, Yejin Choi, Noah A. Smith, Hannaneh Hajishirzi, and Daniel Khashabi. 2022b. Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ Tasks. In *Conference on Empirical Methods in Natural Language Processing* (EMNLP).

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Orion Weller, Aleem Khan, Nathaniel Weir, Dawn Lawrie, and Benjamin Van Durme. 2022. Defending against poisoning attacks in open-domain question answering. *arXiv e-prints*, pages arXiv–2212.

Orion Weller, Kyle Lo, David Wadden, Dawn Lawrie, Benjamin Van Durme, Arman Cohan, and Luca Soldaini. 2023. When do generative query and document expansions fail? a comprehensive study across methods, retrievers, and datasets. *arXiv preprint arXiv:2309.08541*.

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

A Dataset Details

ELI5, or "Explain Like I'm 5" (Fan et al., 2019) is a long-form QA dataset composed of user questions and answers from the subreddit r/ELI5. We use the KILT version (Petroni et al., 2021a) dev set of ELI5 since it is a "grounded" subset of the original (with the non-grounded questions filtered out), allowing a more suitable evaluation of our research question.

Natural Questions (NQ) (Kwiatkowski et al., 2019) is a short-form (< 5 word answer) QA dataset gathered from real-world Google searches. To compare with previous work in prompting on NQ, we evaluate on the full development set.

TriviaQA (TQA) (Joshi et al., 2017) was collected by scraping question and answer pairs from trivia websites, and then matching the answers (short-form) to Wikipedia passages. Following previous work, we use the filtered dev set (7k instances).

HotpotQA (Yang et al., 2018) is a multi-step short-form question-answering dataset that requires two-step reasoning to come to the correct answer. It was gathered from crowdsourcing questions and answers from Amazon Mechanical Turk using two-hop links on Wikipedia. We use the full dev set.

B All Results

B.1 Sources for SOTA Performance in Table 3

SOTA zero-shot results are from LLaMA 33B and 65B (Touvron et al., 2023), PaLM 540B (Wang et al., 2022a), and BART (Su et al., 2022) respectively. For retrieval-augmented SOTA, we show Izacard et al. (2022) for NQ, TriviaQA and HotpotQA, and Su et al. (2022) for ELI5.

B.2 Additional Models for *According-To* vs Null

We show all results for the models in Table 3 that did not fit due to space.

Prompts for Short-Form QA. For short-form QA datasets, to help the models generate both the answer and the explanation in a parseable format, we append the following prompt before the question:

> You are a highly intelligent & complex question-answer generative model. You take a question as an input and answer it by imitating the way a human gives short answers with a corresponding explanation. You answer should be short - only a few words.\n\nYour output format should be the answer, then a semicolon, then the explanation.\n

For models that don't respond well to the above prompt (or similar prompts aimed at generating both answer and explanation from one generation), we use the following prompts in a two step manner:

> Output the answer only. {Insert Question}\nAnswer string only:

> Question: {Insert Question}\nAnswer: {Insert Previous Output}\n\n\nGive a detailed explanation for why this is true. {Insert Grounding Prompt Here} \nExplanation:

Prompt for ELI5 for smaller models. For T5-v1.1-Adapt and GPT-J-Instruct evaluated on ELI5, we append "Answer:" following the end of both the normal null and grounding prompts because otherwise the model outputs very short (< 5 words) or nonsensical responses. With this addition the model produces normal fluent text.

C Conventional N-Gram Metrics

Toolkits such as `sacrebleu` implement multiple n-gram metrics (Post, 2018). However, these tend to use conventional data structures such as python sets and dictionaries. These are not suitable for measuring n-gram metrics against very large references (i.e. the entirety of Wikipedia). In Table 6 we compare the sizes of several datastructures on a sample of $\sim$ 100M n-grams (approximately 0.07% of the 25 char-grams in Wikipedia). The typical CPython set or dictionary implementation uses a hashtable of pointers to data elements (i.e. character n-grams or strings). It requires substantial memory to store both the hashtable backing array and the string data elements (11,107 MiB). This could be optimized by storing only the table and not the data elements, introducing false positives for hash collisions (note that this is similar to a Bloom filter with $k = 1$ hash functions). One could also store only pointers (references) into the original text rather than copies of the string. These options are still larger than an optimal Bloom filter which uses around 14 bits per element for our chosen parameters. On the sampled data, this consumes only 163 MiB of memory. Extrapolating these storage costs indicates that using a naive, un-optimized python set or dictionary would consume around 1.5TB of memory to store all n-grams.

Note that these measurements are only for a single n-gram width. If comparing QUIP-Score to a

Model	Prompt	TQA		NQ		Hotpot		ELI5	
		QUIP	EM	QUIP	EM	QUIP	F1	QUIP	R-L
Text-Davinci-003	Null	35.9	68.2	38.7	24.3	34.6	29.2	27.7	23.7
Text-Davinci-003	Grounded	41.2	71.8	44.4	29.3	39.6	31.3	32.2	22.8
GPT-4	Null	-	-	-	-	-	-	21.0	21.5
GPT-4	Grounded	-	-	-	-	-	-	24.7	21.0
GPT-J-Instruct	Null	28.1	2.2	28.2	0.9	29.2	7.0	22.8	19.9
GPT-J-Instruct	Grounded	31.5	2.1	32.5	1.0	33.2	7.0	27.0	19.4
Koala	Null	34.0	17.2	36.1	6.3	33.9	13.2	24.1	19.9
Koala	Grounded	35.8	17.2	38.4	6.3	35.6	13.2	32.6	22.8
FLAN-T5 XXL	Null	18.6	31.5	23.5	13.3	25.8	23.6	14.9	12.4
FLAN-T5 XXL	Grounded	26.6	31.5	33.2	13.3	31.1	23.6	30.6	18.7

Table 5: Full results for other models. Note that the low EM scores for GPT-J and Koala are due to the model failing to output short answers zero shot (e.g. "the answer is..." instead of outputting the answer. Both used the same instruction tuning dataset.). GPT-4 was only run on ELI5 due to cost.

metric that stores n-grams of multiple widths, this could further increase memory usage.

Structure	Size (MiB)
set	11107
set (no elements)	4096
Bloom filter D.P	163

Table 6: Sizes of structures holding 100M n-grams.

D Prompts for Non-Wikipedia Datasets

We use the same style of prompt as in the Wikipedia sections, but modify them to adapt to the format of the tasks. For example, on the SARA dataset ChatGPT would predict entailment for every question unless additional wording was given to be more balanced in its prediction.

SARA.

> You are a highly intelligent & complex legal statutory entailment system that checks whether a particular judgement holds for a particular case in the U.S. legal tax code. You take a the ruling of a legal situation and respond with a reply of contradiction or entailment, along with a corresponding two paragraph explanation. You answer should be short - only contradiction or entailment. Be sure to verify that the entailment is 100% correct, otherwise choose contradiction.\n\nYour output format should be the answer, then a semicolon, then the verbose explanation.\n\nPremise:{Insert Text Background}\nHypothesis{Insert Question}\n\nFill in the following:\nANSWER HERE; EXPLANATION HERE

PubMed. We use the same prompt as usually specified for Short Form QA above, except using "`\n\nAccording to PubMed,`" as the grounding prompt.

MedQA. We use the same Short Form answer beginning prompt and the following changes to the grounding prompt, "`\n\nAccording to PubMed the multiple choice answer is:\n`" as without the multiple choice specifier it would fail to correctly predict the multiple choice answer.

MedicationQA. We only append a grounding prompt and no prompt before the question (as it is not short form). We use the same grounding prompt as in PubMedQA.

E Computational Resources

We use model APIs for experiemnts with OpenAI and use 1 A100 GPU for experiments with local models. Each experiment took less than an hour for each dataset approximately.