Large Language Models Struggle to Learn Long-Tail Knowledge

Nikhil Kandpal [1] **Haikang Deng** [1] **Adam Roberts** [2] **Eric Wallace** [3] **Colin Raffel** [1]

Abstract

The Internet contains a wealth of knowledge—from the birthdays of historical figures to tutorials on how to code—all of which may be learned by language models. However, while certain pieces of information are ubiquitous on the web, others appear extremely rarely. In this paper, we study the relationship between the knowledge memorized by large language models and the information in pre-training datasets scraped from the web. In particular, we show that a language model's ability to answer a fact-based question relates to how many documents associated with that question were seen during pre-training. We identify these relevant documents by entity linking pre-training datasets and counting documents that contain the same entities as a given question-answer pair. Our results demonstrate strong correlational and causal relationships between accuracy and relevant document count for numerous question answering datasets (e.g., TriviaQA), pre-training corpora (e.g., ROOTS), and model sizes (e.g., 176B parameters). Moreover, while larger models are better at learning long-tail knowledge, we estimate that today's models must be scaled by many orders of magnitude to reach competitive QA performance on questions with little support in the pre-training data. Finally, we show that retrieval-augmentation can reduce the dependence on relevant pre-training information, presenting a promising approach for capturing the long-tail.

1. Introduction

Large language models (LLMs) trained on text from the Internet capture many facts about the world, ranging from well-known factoids to esoteric domain-specific information. These models implicitly store knowledge in their parameters

[1]UNC Chapel Hill [2]Google Research [3]UC Berkeley. Correspondence to: Nikhil Kandpal <nkandpa2@cs.unc.edu>.
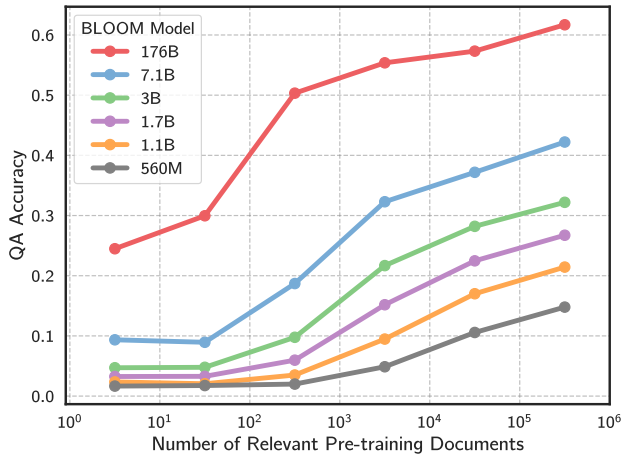
Figure 1. Language models struggle to capture the long-tail of information on the web. Above, we plot accuracy for the BLOOM model family on TriviaQA as a function of how many documents in the model's pre-training data are relevant to each question.

(Petroni et al., 2019; Roberts et al., 2020), and given the scale of today's pre-training datasets and LLMs, one would hope that they can learn a huge amount of information from web-sourced text. However, not all of the knowledge on the Internet appears equally often—there is a long-tail of information that appears rarely or only once.

In this work, we explore the relationship between the knowledge learned by an LLM and the information in its pre-training dataset. Specifically, we study how an LLM's ability to answer a question relates to how many documents associated with that question were seen during pre-training. We focus on factoid QA datasets (Joshi et al., 2017; Kwiatkowski et al., 2019), which lets us ground question-answer pairs into concrete subject-object co-occurrences. As an example, for the QA pair (*In what city was the poet Dante born?*, *Florence*), we consider documents where the entities Dante and Florence co-occur as highly relevant. To identify these entity co-occurrences we apply a highly-parallelized entity linking pipeline to trillions of tokens from datasets such as C4 (Raffel et al., 2020), The Pile (Gao et al., 2020), ROOTS (Laurençon et al., 2022), OpenWebText (Gokaslan & Cohen, 2019), and Wikipedia.

We observe a strong correlation between an LM's ability to answer a question and the number of pre-training documents
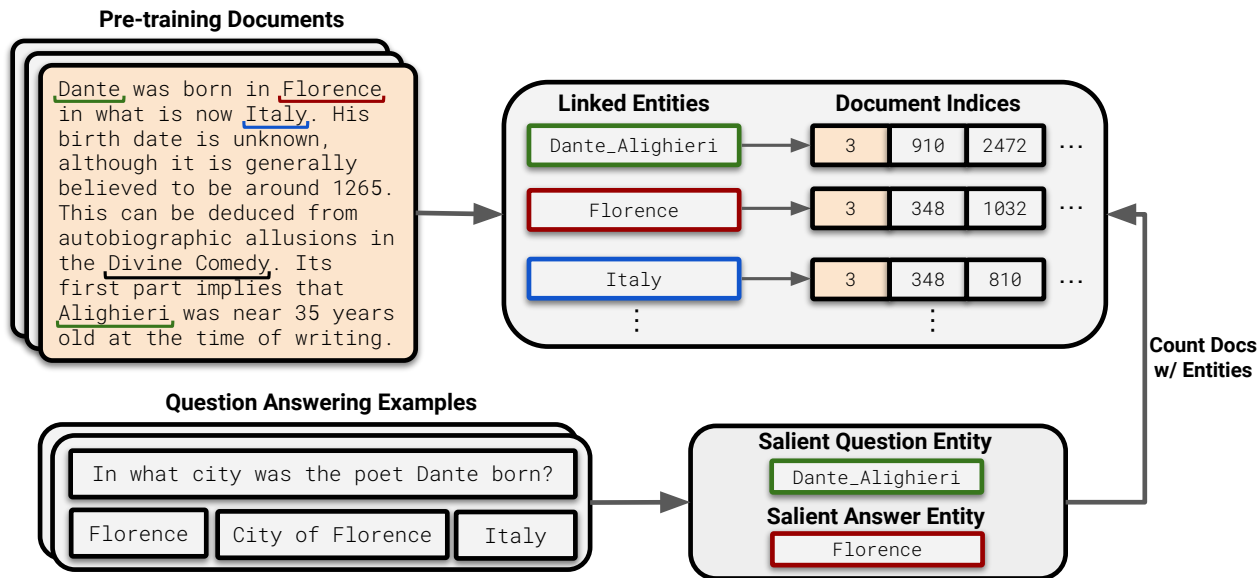
Figure 2. In our document counting pipeline, we first run entity linking on large pre-training datasets (*top left*) and store the set of the document indices in which each entity appears (*top right*). We then entity link downstream QA pairs and extract the salient question and answer entities (*bottom*). Finally, for each question we count the number of documents in which the question and answer entities co-occur.

relevant to that question for numerous QA datasets, pre-training datasets, and model sizes (e.g., Figure 1). For example, the accuracy of BLOOM-176B (Scao et al., 2022) jumps from 25% to above 55% when the number of relevant pre-training documents increases from 10^1 to 10^4.

We also conduct a counterfactual re-training experiment, where we train a 4.8B-parameter LM with and without certain documents. Model accuracy drops significantly on questions whose relevant documents were removed, which validates our entity linking pipeline and shows that the observed correlational trends are likely causal in nature.

Finally, we analyze ways to better capture knowledge that rarely appears in the pre-training data: model scaling and retrieval-augmentation. For model scaling, we find a strong log-linear relationship between parameter count and QA accuracy. These trends show that while scaling up LMs improves knowledge learning, models would need to be scaled dramatically (e.g., to one quadrillion parameters) to achieve competitive QA accuracy on long-tail questions. Retrieval-augmented systems are more promising—when a retriever succeeds in finding a relevant document, it reduces an LLM's need to have a large amount of relevant pre-training text. Nevertheless, retrieval systems themselves still exhibit a mild dependence on relevant document count.

Overall, our work is one of the first to study how LLM knowledge is influenced by pre-training data. To enable future research, we release our code as well as the entity data for ROOTS, The Pile, C4, OpenWebText, and Wikipedia at https://github.com/nkandpa2/long_tail_knowledge.

2. Identifying Relevant Pre-training Data

Background and Research Question   Numerous NLP tasks are *knowledge-intensive*: they require recalling and synthesizing facts from a knowledge source (e.g. Wikipedia or the web). Results on knowledge-intensive tasks have been dramatically improved using LLMs, as these models have been shown to leverage the vast amounts of knowledge they learn from their pre-training corpora (Roberts et al., 2020; Petroni et al., 2019; De Cao et al., 2021). However, it remains unclear as to *what kind* of knowledge LMs actually capture—for example, do they simply learn "easy" facts that frequently appear in their pre-training data?

We study this question using closed-book QA evaluations (Roberts et al., 2020) of LLMs in the few-shot setting (Brown et al., 2020). Models are prompted with in-context training examples (QA pairs) and a test question without any relevant background text. The goal of our work is to investigate the relationship between an LM's ability to answer a question and the number of times information relevant to that question appears in the pre-training data.

Our Approach   The key challenge is to efficiently identify all of the documents that are relevant to a particular QA pair in pre-training datasets that are hundreds of gigabytes in size. To tackle this, we begin by identifying the salient entities that are contained in a question and its set of ground-truth answer aliases. We then identify relevant pre-training documents by searching for instances where the salient question entity and the answer entity co-occur.

For example, consider the question *In what city was the poet Dante born?* with the valid answers *Florence*, *City of Florence*, and *Italy* (e.g., Figure 2). We extract the salient question and answer entities, Dante_Alighieri and Florence, and count the documents that contain both entities.

Our approach is motivated by Elsahar et al. (2018), who show that when only the subject and object of a subject-object-relation triple co-occur in text, the resulting triple is often also present. In addition, we conduct human studies that show our document counting pipeline selects relevant documents a majority of the time (Section 2.3). Moreover, we further validate our pipeline by training an LM without certain relevant documents and showing that this reduces accuracy on the associated questions (Section 3.2). Based on these findings, we refer to documents that contain the salient question and answer entities as **relevant documents**.

To apply the above method, we must entity link massive pre-training corpora, as well as downstream QA datasets. We accomplish this by building a parallelized pipeline for entity linking (Section 2.1), which we then customize for downstream QA datasets (Section 2.2).

2.1. Entity Linking Pre-training Data

We perform entity linking at scale using a massively distributed run of the DBpedia Spotlight Entity Linker (Mendes et al., 2011), which uses traditional entity linking methods to link entities to DBpedia or Wikidata IDs. We entity link the following pre-training datasets, which were chosen based on their use in the LLMs we consider:

- **The Pile:** an 825GB dataset that contains a mix of 22 different primarily English-language sources (Gao et al., 2020).

- **ROOTS (En):** the 490GB English subset of the ROOTS corpus (Laurençon et al., 2022). Note that we do not study if models trained on the non-English subsets of ROOTS are able to leverage cross-lingual factual knowledge.

- **C4:** a 305GB English corpus that was collected by filtering CommonCrawl (Raffel et al., 2020).

- **OpenWebText:** a 39GB English corpus that contains the text of web pages that were linked on the website Reddit (Gokaslan & Cohen, 2019).

- **Wikipedia:** a text dump of December 2018 Wikipedia articles from Lee et al. (2019), a standard corpus for evaluating open-domain QA systems (e.g. Karpukhin et al. 2020; Lewis et al. 2020; Guu et al. 2020).

For each document in these pre-training datasets, we record the linked entities in a data structure that enables quickly counting individual entity occurrences and entity co-occurrences. This pipeline took approximately 3 weeks to entity link 2.1TB of data on a 128-CPU-core machine.

2.2. Finding Entity Pairs in QA Data

We next entity link two standard open-domain QA datasets: Natural Questions (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017). To expand our sample sizes, we use both the training and validation data, except for a small set of examples used for few-shot learning prompts.

We first run the DBPedia entity linker on each example. Because there can be multiple annotated answers for a single example, we concatenate the question and all valid answers, as this enabled more accurate entity linking. We use the most common entity found in the set of ground truth answers as the salient answer entity. We then iterate over all entities found in the question and select the entity that co-occurs the most with the salient answer entity in the pre-training data. In cases where no entity is found in the question, answer, or both, we discard the example. If the resulting number of relevant documents is zero, we discard the example, as this is likely due to an entity linking error.

2.3. Human Evaluation of Document Counting Pipeline

Here, we conduct a human evaluation of our document identification pipeline. Note that a document can vary in the extent to which it is "relevant" to a particular QA pair. For instance, consider the QA pair (*William Van Allan designed which New York building—the tallest brick building in the world in 1930?*, *Chrysler Building*). The documents that we identify as relevant may (1) contain enough information to correctly answer the question, (2) contain information relevant to the question but *not enough* to correctly answer it, or (3) contain no relevant information. For example, a document that mentions that the Chrysler building was designed by William Van Allan, but not that it was the tallest brick building in 1930, would fall into the second category.

We randomly sample 300 QA pairs from TriviaQA and selected one of their relevant documents at random. We then manually labeled the documents into one of the three categories: 33% of documents contained enough information to answer the question and an additional 27% of contained some relevant information. Thus, our pipeline has ∼60% precision at identifying relevant documents for TriviaQA.

Our pipeline is imperfect as (1) the entity linker sometimes mis-identifies entities and (2) not all documents containing the salient question and answer entity are relevant. However, when applied at the scale of large-scale pre-training datasets, this pipeline is efficient and achieves enough precision and recall to observe correlational (Section 3.1) and causal (Section 3.2) relationships to QA performance.
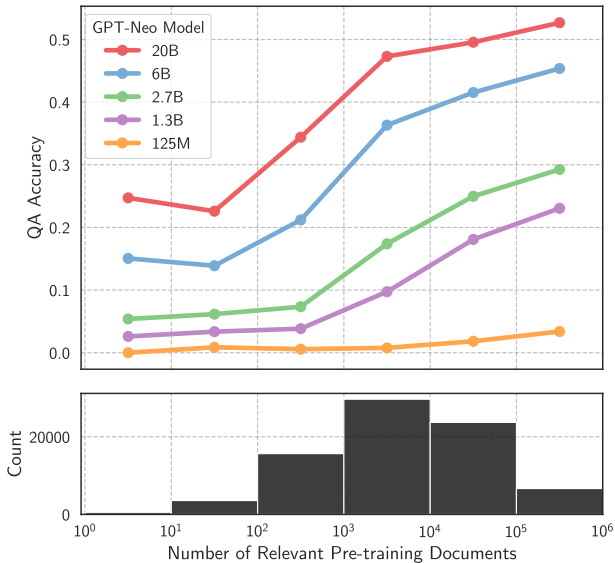
Figure 3. We plot accuracy on TriviaQA versus relevant document count for GPT-Neo. The trends match those seen for BLOOM (Figure 1). We also include a histogram that shows how many QA examples fall into each bucket; TriviaQA often asks about knowledge represented 10^2 to 10^5 times in the pre-training data.

3. LM Accuracy Depends on Relevant Document Count

In this section, we measure the relationship between an LLM's ability to answer a question and the number of relevant documents in the pre-training corpus. We use popular Transformer decoder-only LMs (Vaswani et al., 2017) that span three orders of magnitude in size:

- **GPT-Neo:** The GPT-Neo, GPT-NeoX, and GPT-J LMs trained by EleutherAI on the Pile (Gao et al., 2020) that range in size from 125M to 20B parameters (Black et al., 2021; Wang & Komatsuzaki, 2021; Black et al., 2022). We refer to these models collectively as GPT-Neo models.

- **BLOOM:** Models trained by the BigScience initiative on the ROOTS dataset (Scao et al., 2022). The BLOOM models are multi-lingual; we analyze their English performance only. The models range in size from 560M to 176B parameters.

- **GPT-3:** Models trained by OpenAI that range in size from $\approx$350M (Ada) to $\approx$175B parameters (Davinci). Since the pre-training data for these models is not public, we estimate relevant document counts by scaling up the counts from OpenWebText to simulate if the dataset was the same size as GPT-3's pre-training data. We recognize that there is uncertainty around these models' pre-training data, their exact sizes, and whether they have been fine-tuned. We therefore report these results in the Appendix for readers to interpret with these sources of error in mind.
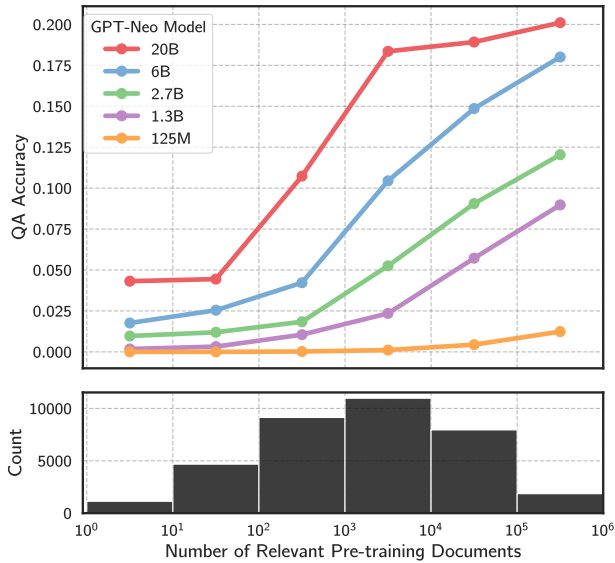

Figure 4. We plot accuracy on Natural Questions versus relevant document count for GPT-Neo. The trends match those in TriviaQA—model accuracy is highly dependent on fact count.

We use these LMs because (with the exception of GPT-3) they are the largest open-source models for which the pre-training data is publicly available. We focus on 4-shot evaluation, although we found that other amounts of in-context training examples produced similar trends. We use simple prompts consisting of templates of the form

> Q: [In-Context Question 1]
>
> A: [In-Context Answer 1]
>
> $\vdots$
>
> Q: [In-Context Question n]
>
> A: [In-Context Answer n]
>
> Q: [Test Question]

We generate answers by greedy decoding until the models generate a newline character, and we evaluate answers using the standard Exatch Match (EM) metric against the ground-truth answer set (Rajpurkar et al., 2016).

3.1. Correlational Analysis

We first evaluate the BLOOM and GPT-Neo model families on TriviaQA and plot their QA accuracies versus the number of relevant documents in Figures 1 and 3. For improved readability, we average the accuracy for QA pairs using log-spaced bins (e.g., the accuracy for all questions with 1 to 10 relevant documents, 10 to 100 relevant documents, etc.). Below each plot, we also include a histogram that shows how many QA examples fall into each bin. We trim the plots when the bins contain fewer than 500 QA examples to avoid reporting accuracies for small sample sizes.
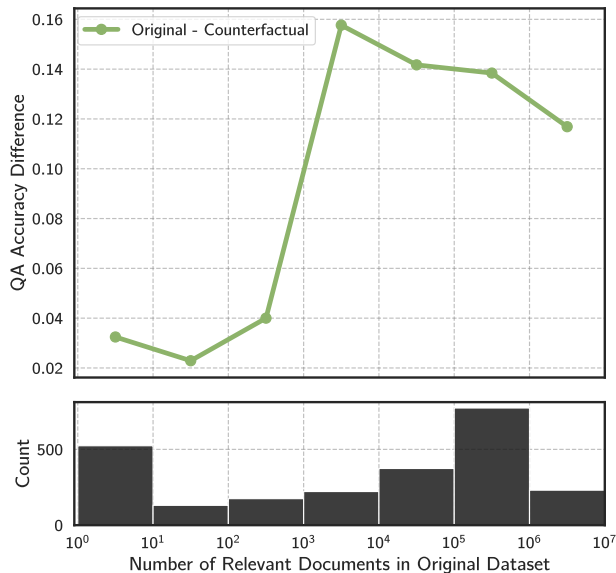
Figure 5. We run a counterfactual experiment, where we re-train an LM without certain documents. We take TriviaQA questions with different document counts and delete all of their relevant pre-training documents. The difference in accuracy between the original model and the re-trained LM (*counterfactual*) is high when the original number of relevant documents is large.

There is a strong correlation between question answering accuracy and relevant document count for all tested models. Correspondingly, when the number of relevant documents is low, models are quite inaccurate, e.g., the accuracy of BLOOM-176B jumps from 25% to above 55% when the number relevant documents increases from 10^1 to 10^4. Model size is also a major factor in knowledge learning: as the number of model parameters is increased, the QA performance substantially improves. For example, BLOOM-176B has over $4\times$ higher accuracy than BLOOM-560M on TriviaQA questions with more than 10^5 relevant documents.

We repeat this experiment using the Natural Questions QA dataset and find similar trends for all model families (see Figure 4 for GPT-Neo, and Figures 10 and 11 in the Appendix for BLOOM and GPT-3 results).

Simpler Methods for Identifying Relevant Documents Are Less Effective In the experiments above, we identify relevant documents by searching for co-occurrences of salient question and answer entities. To evaluate whether this process is necessary, we compare against two baseline document identification methods: counting documents that contain the salient question entity and counting documents that contain the salient answer entity (as done in Petroni et al. 2019).

We show in Figure 13 that all three document identification methods are correlated with QA accuracy. However, when

only considering QA examples where the question and answer entities co-occur few (< 5) times, the two baseline methods no longer correlate with QA accuracy. This indicates that counting documents with just the answer entity or question entity alone is insufficient for explaining why LMs are able to answer certain questions. This validates our definition of relevant documents as those that contain both the question entity and answer entity.

Humans Show Different Trends Than LMs An alternate explanation for our results is that questions with lower document counts are simply "harder", which causes the drop in model performance. We show that this is not the case by measuring human accuracy on Natural Questions. We use a leave-one-annotator-out metric, where we take questions that are labeled by 5 different human raters (all of whom can see the necessary background text), hold out one of the raters, and use the other four as the ground-truth answer set. We plot the human accuracy versus relevant document count in the top of Figure 7. Human accuracy is actually highest for the questions with *few* relevant documents, the opposite trend of models. We hypothesize that humans are better on questions with few relevant documents because (1) questions about rarer facts are more likely to be simple factoids compared to common entities, and (2) the Wikipedia documents are that are provided to the annotators are shorter for rarer entities, which makes reading comprehension easier and increases inner-annotator agreement.

3.2. Causal Analysis via Re-training

Our results thus far are correlational in nature: there may be unknown confounds that explain them away, i.e., the rarer questions are more difficult for LMs for other reasons. Here we establish a causal relationship by removing certain documents in the training data and re-training the LM.

We first train a baseline 4.8 billion parameter LM on C4, following the setup from Wang et al. (2022). We then measure the effect of deleting certain documents from the training set. For each log-scaled bin of relevant document count (e.g., 10^0 to 10^1 relevant documents, 10^1 to 10^2, ...) we sample 100 questions from Trivia QA and remove all relevant documents for those questions in C4. In total, this removes about 30% of C4. Finally, we train a "counterfactual" LM on this modified pre-training dataset and compare its performance to the baseline model. For both the baseline model and the counterfactual model, we train for a single epoch. Note that the counterfactual model was trained for 30% fewer steps, which makes it slightly worse in performance overall. To account for this, we only study the performance on questions whose relevant documents were removed.

We show the difference in performance between the two LMs on questions whose documents were removed in Fig-
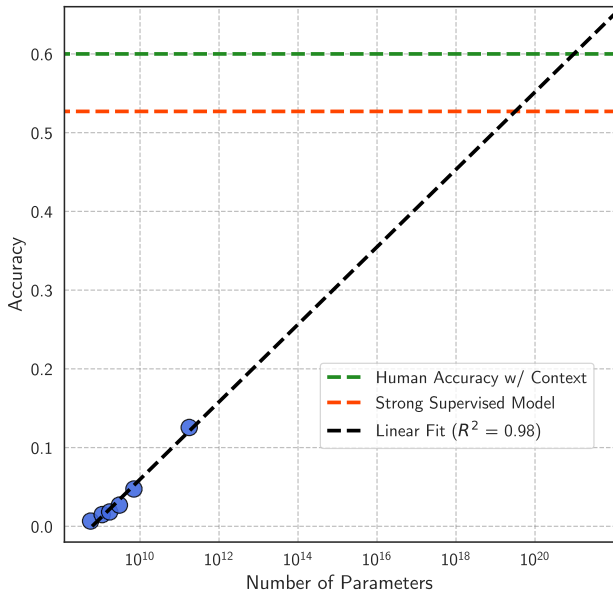
Figure 6. Scaling trends for fact learning. We plot BLOOM accuracy on rare instances from Natural Questions (< 100 relevant docs) as a function of the log of the model size. Extrapolating from the empirical line of best fit—which approximates the trend well at $R^2 = 0.98$—implies that immensely large models would be necessary to get high accuracy.

	ROOTS	Pile	C4	OWT	Wiki
ROOTS	-	0.97	0.97	0.94	0.87
Pile	-	-	0.95	0.96	0.87
C4	-	-	-	0.96	0.90
OWT	-	-	-	-	0.91
Wiki	-	-	-	-	-

Table 1. Spearman rank correlations of the relevant document counts for TriviaQA examples in The Pile, ROOTS, C4, OpenWeb-Text, and Wikipedia. Despite having different collection methodologies, these pre-training datasets are highly correlated in terms of how much information they contain related to different QA pairs.

dently, the amount of supporting information they provide for different TriviaQA examples is highly consistent as seen by the rank correlations between their relevant document counts in Table 1.

4.2. Can We Scale Up Models?

Using larger models consistently produces better QA performance. However, our results suggest that one would need immensely large LMs to achieve high accuracy on long-tail questions. In Figure 6 we plot a scaling trend line for rare fact learning, where we show BLOOM accuracy on rare instances from Natural Questions (< 100 relevant docs) as a function of the log of the model size. The empirical log-linear trend—which approximates the scaling extremely well ($R^2 = 0.98$)—shows that in order to match a strong supervised baseline (Izacard & Grave, 2021) or human performance, one would need a BLOOM model with over 10^{18} (one quintillion) parameters.[1] We see similar trends for other models and datasets (see Figure 12 in the Appendix).

ure 5. For questions with few relevant documents in the original C4 dataset, performance is poor for both the baseline and the counterfactual LM, i.e., their performance difference is small. However, for questions with many relevant documents, performance is significantly worse for the counterfactual LM. This suggests a *causal* link between the number of relevant documents and QA performance.

4. Methods to Improve Rare Fact Learning

Thus far, we showed that LLMs have a strong dependence on relevant document count. Here, we investigate methods to mitigate this dependence: increasing data scale, increasing model scale, and adding an auxiliary retrieval module.

4.1. Can We Scale Up Datasets?

Today's largest LLMs are pre-trained on hundreds of billions of tokens. One naïve approach for improving accuracy on questions about less-prevalent knowledge is to collect larger quantities of data. Our results suggest that this would not significantly improve accuracy as scaling datasets by moderate factors (e.g., 5x) usually results in small accuracy gains. An alternative idea would be to increase the diversity of the pre-training data. However, we also believe this would provide minimal benefit because many data sources are surprisingly correlated . Although each of the pre-training datasets considered were collected indepen-

Modifying the Training Objective Another option similar to scaling up models is to directly modify the training objective to encourage memorization. One simple method to accomplish this is to increase the number of training epochs. All of the LMs that we study do limited epochs, as it is generally seen as preferable to use large enough pre-training datasets so that the LM completes one epoch of training when the compute budget is exhausted (Raffel et al., 2020). However, in the context of QA, it may be preferable to increase epochs and reduce data size to ensure models memorize as much as possible. Alternatively, one could consider modifying the training loss to encourage the model to focus on salient facts (Guu et al., 2020) or designing a curriculum to minimize forgetting (Jagielski et al., 2023).

---

[1] For this experiment, the supervised and human accuracies are computed over the validation set whereas the scaling trend is computed using the train and validation sets.
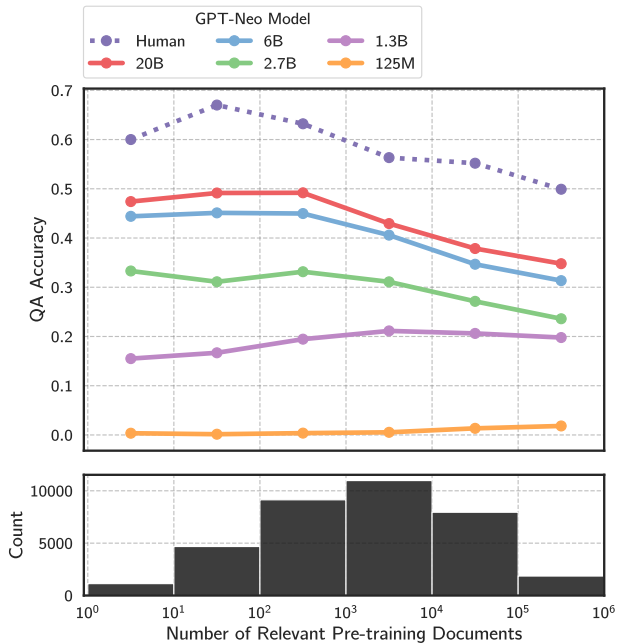
Figure 7. Models with access to the required background context do not struggle on questions with low relevant document count. Concretely, we provide questions and gold paragraphs to GPT-Neo models on Natural Questions, and their accuracy trends roughly match the trends of humans.

4.3. Can We Use Retrieval Augmentation?

Thus far, we use LMs as isolated systems that do not leverage external information. However, for knowledge-intensive tasks, a natural alternative is to make LMs *retrieval-augmented*, i.e., combine them with a retrieval module that returns relevant textual contexts (Lewis et al., 2020; Guu et al., 2020; Karpukhin et al., 2020). Here, we study whether retrieval-augmented models can mitigate the dependence on the amount of relevant knowledge in the pre-training data.

Oracle Retrieval We first study an oracle setting where we provide LMs with a gold paragraph from Wikipedia that supports each answer in Natural Questions (Petroni et al., 2020). We use the 300-word segment that surrounds the ground-truth answer from the gold Wikipedia page and evaluate the 2-shot accuracy of GPT-Neo. Figure 7 shows that oracle retrieval-augmentation dramatically boosts accuracy over closed-book models, especially on rarer instances. Similar to Liu et al. (2022), we also find that QA accuracy actually goes *down* as the number of relevant documents increases—the opposite trend of closed-book LLMs. As discussed in Section 3.1, humans exhibit the same trend, likely because rare questions are easier on average when relevant context information.

BM25 Retrieval We next follow a common retrieval-augmented baseline, where we use a BM25 retriever (Robertson & Zaragoza, 2009) to select paragraphs from Wikipedia. We add the top-3 highest scoring paragraphs into the prompt for both the in-context training examples and the test question. We verify that at least one of the retrieved paragraphs contains the answer for each in-context training example, to ensure that the LM learns to utilize on the documents.

We first evaluate the BM25 retriever's top-k recall on its knowledge corpus (Wikipedia) as a function of relevant document count, and plot the results in Figure 8. We find that BM25 attains reasonably high recall, especially for larger values of k. However, the BM25 retriever still shows a mild dependence on relevant document count. We next evaluate the accuracy of BM25-augmented GPT-Neo models on Natural Questions and plot the results in Figure 9. Overall, retrieval-augmented models outperform their closed-book counterparts across all ranges of relevant document counts, and especially on rare examples. These results suggest that retrieval augmentation provides a promising path towards improving performance on questions with few relevant documents in the pre-training dataset.

5. Related Work

Identifying The Origins of Few-shot Learning Our work contributes to an emerging line of research that explains the success of zero- and few-shot learning in language models by tracing their behavior back to the pre-training data. For example, Razeghi et al. (2022) show mathematical reasoning capabilities can be correlated with training data frequency, and Shin et al. (2022) and Han & Tsvetkov (2022) show that training corpus source can influence few-shot accuracies.

The most similar work to ours in this context is Elazar et al. (2022), who use causal inference to measure the effect of pre-training data statistics on QA performance. Their main focus is testing the extent to which LMs answer questions using heuristics based on co-occurrences between subjects, objects, and textual patterns in the pre-training data. Our main focus is to measure the relationship between the knowledge learned by an LLM and the prevalence of that knowledge in the pre-training data. Moreover, we also conduct re-training experiments and study how model scaling and retrieval-augmentation affect knowledge learning.

Memorization and Privacy Past work studies training data memorization from the perspective of privacy, i.e., how LMs inadvertently reveal private text (Carlini et al., 2019; 2021; Lee et al., 2021). These works focus on how LMs memorize and repeat *verbatim* text samples, and the effect of duplicating those texts in the training set (Kandpal et al.,
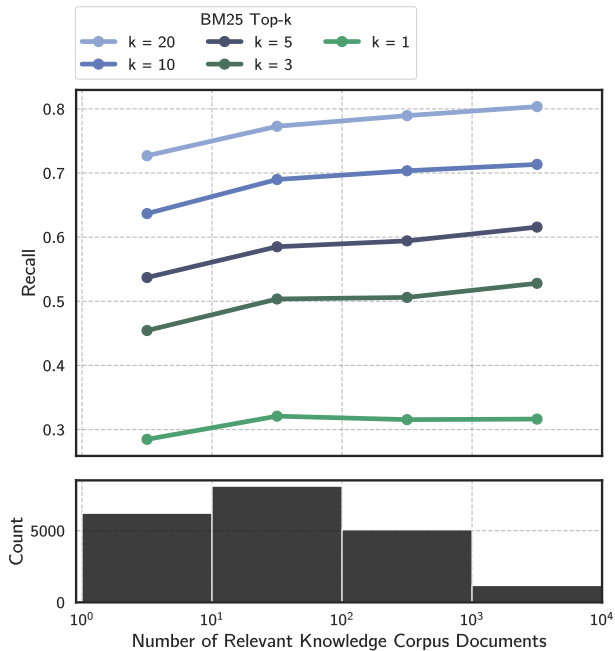
Figure 8. Retrieval systems such as BM25 have a mild dependence on document count. Above we plot the top-k recall for BM25 on Natural Questions for different values of k.

Figure 9. Retrieval-augmented LMs no longer exhibit low accuracy on rare instances. We plot GPT-Neo accuracy on Natural Questions when augmented with three paragraphs from BM25.

2022). Doing so has various limitations, as memorization can be harmful or beneficial even in non-verbatim cases (Ippolito et al., 2022). Our work takes studies non-verbatim memorization in the context of QA—our LMs memorize facts in text form and then answers questions about those facts at test time.

Memorization and Fact Learning Existing work also analyzes the relationship between the pre-training data and the factual knowledge of LLMs. Akyürek et al. (2022) look to automatically identify which documents were most influential for a language model's QA predictions. Our work instead directly identifies and estimates the number of relevant documents via entity linking large corpora. Other work notices a correspondence between model accuracy and data frequency for different knowledge-intensive tasks (Petroni et al., 2019; Kassner et al., 2020; De Cao et al., 2021; Wei et al., 2021; Févry et al., 2020) and for domains outside of NLP (Rao et al., 2021). Our paper reports similar findings, but scales this analysis to massive LM pre-training datasets and model sizes.

In concurrent and independent work, Mallen et al. (2022) study how QA performance correlates with frequency in the pre-training data. Unlike our work, they do not use entity linking methods to count occurrences and instead use proxies such as entity popularity on Wikipedia. They also find QA accuracy is highly correlated with pre-training
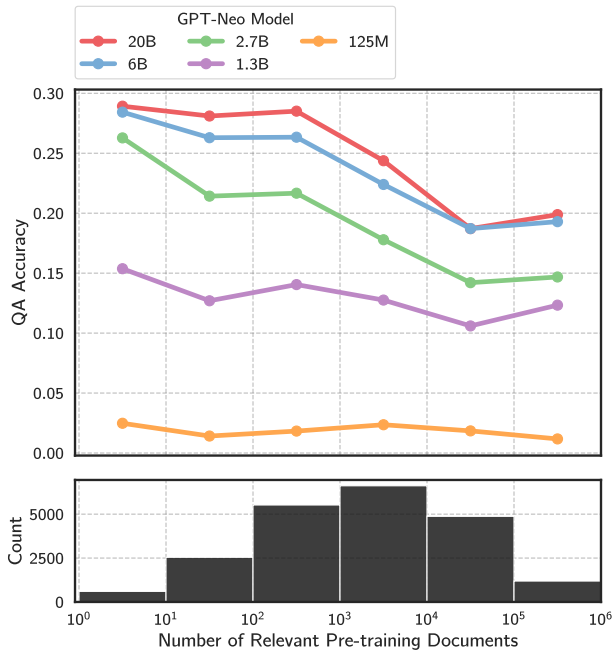
data frequency and show that retrieval models can improve long-tail knowledge. Our work differs in that we conduct causal re-training experiments and find that model scaling is highly beneficial to long-tail QA performance.

6. Conclusion and Future Work

Large language models demonstrate impressive few-shot learning capabilities that arise from simply training on large-scale internet text. With the open-source release of LLMs—and their associated pre-training datasets—the research community can now begin to understand the origins of these capabilities. Our work is one of the first to relate an observed phenomenon in LLMs back to the pre-training data itself. In our case, our results are negative: while LLMs achieve moderate performance on open-domain QA benchmarks, they are mainly successful on questions that probe knowledge that appears widely in their pre-training datasets.

Our work raises numerous directions for further inquiry, namely, how to improve retention of long-tail knowledge given that simply scaling up model and dataset size will likely be insufficient. We are personally excited about improving retrieval-augmented LMs, especially with regards to their efficiency and retrieval accuracy. Moreover, our work focuses on knowledge learning as it relates to factoid question answering, but we leave open the question as to whether similar relationships exist for other types of

tasks, be it knowledge-intensive or otherwise. Relatedly, even though our work analyzes the impact of memorization on question answering, our results may have implications for other tasks that require using (or avoiding) memorized knowledge, e.g., analyzing private text, performing commonsense reasoning, or predicting source code. Finally, we hope that future evaluations of few-shot learning can continue to shed light into model behavior by tracing accuracy back to properties of the pre-training data. In particular, our work shows that by performing such an analysis, one can help elucidate the successes and failures of existing models, as well as help to identify possible paths forward to improve today's systems.

Acknowledgements

References

Akyürek, E., Bolukbasi, T., Liu, F., Xiong, B., Tenney, I., Andreas, J., and Guu, K. Tracing knowledge in language models back to the training data. In *Findings of EMNLP*, 2022.

Black, S., Leo, G., Wang, P., Leahy, C., and Biderman, S. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, 2021.

Black, S., Biderman, S., Hallahan, E., Anthony, Q., Gao, L., Golding, L., He, H., Leahy, C., McDonell, K., Phang, J., et al. GPT-Neox-20B: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*, 2022.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. In *NeurIPS*, 2020.

Carlini, N., Liu, C., Erlingsson, Ú., Kos, J., and Song, D. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX Security Symposium*, 2019.

Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U., Oprea, A., and Raffel, C. Extracting training data from large language models. In *USENIX Security Symposium*, 2021.

De Cao, N., Izacard, G., Riedel, S., and Petroni, F. Autoregressive entity retrieval. In *ICLR*, 2021.

Elazar, Y., Kassner, N., Ravfogel, S., Feder, A., Ravichander, A., Mosbach, M., Belinkov, Y., Schütze, H., and Goldberg, Y. Measuring causal effects of data statistics on language model's factual predictions. *arXiv preprint arXiv:2207.14251*, 2022.

Elsahar, H., Vougiouklis, P., Remaci, A., Gravier, C., Hare, J., Laforest, F., and Simperl, E. T-REx: A large scale alignment of natural language with knowledge base triples. In *LREC*, 2018.

Févry, T., Soares, L. B., FitzGerald, N., Choi, E., and Kwiatkowski, T. Entities as experts: Sparse memory access with entity supervision. In *EMNLP*, 2020.

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., et al. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Gokaslan, A. and Cohen, V. Openwebtext corpus, 2019.

Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M. Retrieval augmented language model pre-training. In *ICML*, 2020.

Han, X. and Tsvetkov, Y. ORCA: Interpreting prompted language models via locating supporting data evidence in the ocean of pretraining data. *arXiv preprint arXiv:2205.12600*, 2022.

Ippolito, D., Tramèr, F., Nasr, M., Zhang, C., Jagielski, M., Lee, K., Choquette-Choo, C. A., and Carlini, N. Preventing verbatim memorization in language models gives a false sense of privacy. *arXiv preprint arXiv:2210.17546*, 2022.

Izacard, G. and Grave, E. Distilling knowledge from reader to retriever for question answering. In *ICLR*, 2021.

Jagielski, M., Thakkar, O., Tramer, F., Ippolito, D., Lee, K., Carlini, N., Wallace, E., Song, S., Thakurta, A., Papernot, N., et al. Measuring forgetting of memorized training examples. In *ICLR*, 2023.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*, 2017.

Kandpal, N., Wallace, E., and Raffel, C. Deduplicating training data mitigates privacy risks in language models. In *ICML*, 2022.

Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. Dense passage retrieval for open-domain question answering. In *EMNLP*, 2020.

Kassner, N., Krojer, B., and Schütze, H. Are pretrained language models symbolic reasoners over knowledge? In *CoNLL*, 2020.

Kwiatkowski, T., Palomaki, J., Rhinehart, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Kelcey, M., Devlin, J., et al. Natural Questions: A benchmark for question answering research. In *TACL*, 2019.

Laurençon, H., Saulnier, L., Wang, T., Akiki, C., del Moral, A. V., Scao, T. L., Werra, L. V., Mou, C., Ponferrada, E. G., Nguyen, H., Frohberg, J., Šaško, M., Lhoest, Q., McMillan-Major, A., et al. The BigScience ROOTS corpus: A 1.6TB composite multilingual dataset. In *NeurIPS*, 2022.

Lee, K., Chang, M.-W., and Toutanova, K. Latent retrieval for weakly supervised open domain question answering. In *ACL*, 2019.

Lee, K., Ippolito, D., Nystrom, A., Zhang, C., Eck, D., Callison-Burch, C., and Carlini, N. Deduplicating training data makes language models better. In *ACL*, 2021.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *NeurIPS*, 2020.

Liu, L., Lewis, P., Riedel, S., and Stenetorp, P. Challenges in generalization in open domain question answering. In *Findings of NAACL*, 2022.

Mallen, A., Asai, A., Zhong, V., Das, R., Hajishirzi, H., and Khashabi, D. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*, 2022.

Mendes, P. N., Jakob, M., García-Silva, A., and Bizer, C. DBpedia Spotlight: Shedding light on the web of documents. In *International Conference on Semantic Systems*, 2011.

Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A. H., and Riedel, S. Language models as knowledge bases? In *EMNLP*, 2019.

Petroni, F., Lewis, P. S. H., Piktus, A., Rocktäschel, T., Wu, Y., Miller, A. H., and Riedel, S. How context affects language models' factual predictions. In *AKBC*, 2020.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P. J., et al. Exploring the limits of transfer learning with a unified text-to-text transformer. In *JMLR*, 2020.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. SQuAD: 100,000+ questions for machine comprehension of text. In *EMNLP*, 2016.

Rao, R. M., Liu, J., Verkuil, R., Meier, J., Canny, J., Abbeel, P., Sercu, T., and Rives, A. Msa transformer. In *ICML*, 2021.

Razeghi, Y., Logan IV, R. L., Gardner, M., and Singh, S. Impact of pretraining term frequencies on few-shot reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, 2022.

Roberts, A., Raffel, C., and Shazeer, N. How much knowledge can you pack into the parameters of a language model? In *EMNLP*, 2020.

Robertson, S. and Zaragoza, H. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in IR*, 2009.

Scao, T. L., Fan, A., Akiki, C., Pavlick, E.-J., Ili'c, S., Hesslow, D., Castagn'e, R., Luccioni, A. S., Yvon, F., Gallé, M., Tow, J., Rush, A. M., et al. BLOOM: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022.

Shin, S., Lee, S.-W., Ahn, H., Kim, S., Kim, H., Kim, B., Cho, K., Lee, G., Park, W., Ha, J.-W., et al. On the effect of pretraining corpora on in-context learning by a large-scale language model. In *NAACL*, 2022.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

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

Wang, T., Roberts, A., Hesslow, D., Scao, T. L., Chung, H. W., Beltagy, I., Launay, J., and Raffel, C. What language model architecture and pretraining objective work best for zero-shot generalization? In *ICML*, 2022.

Wei, J., Garrette, D., Linzen, T., and Pavlick, E. Frequency effects on syntactic rule learning in transformers. In *EMNLP*, 2021.

A. Additional Results: Relevant Document Scaling

Here we show how QA performance is related to the number of relevant pre-training documents for the BLOOM on Natural Questions (Figure 10) and the GPT-3 model family on TriviaQA and Natural Questions (Figure 11). Like the results in the main text, models are significantly better at answering questions about facts that are well supported in the pre-training data and model scale improves knowledge acquisition.

Note that our estimates for the number of relevant pre-training documents for the GPT-3 model family may be inaccurate since the training data for GPT-3 is not public. Instead, we estimate these relevant document counts using the open source dataset OpenWebText, which was collected with a similar process to the reported collection methodology for the GPT-3 pre-training dataset.
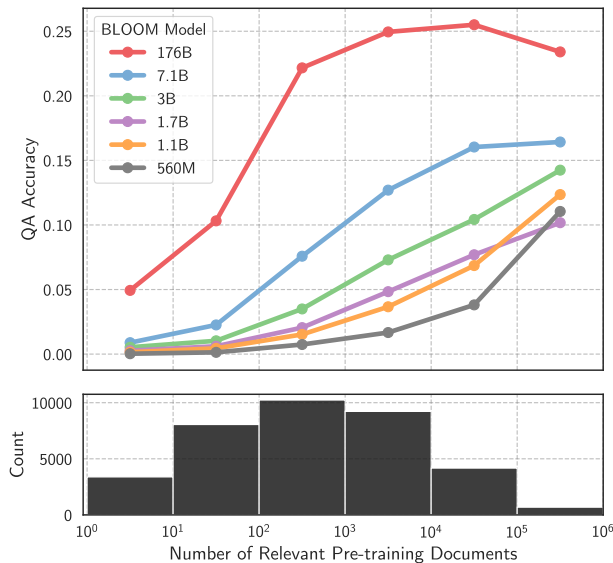

Figure 10. We show results for Natural Questions for BLOOM. The trends match those seen in TriviaQA, although the accuracy is lower overall for Natural Questions.
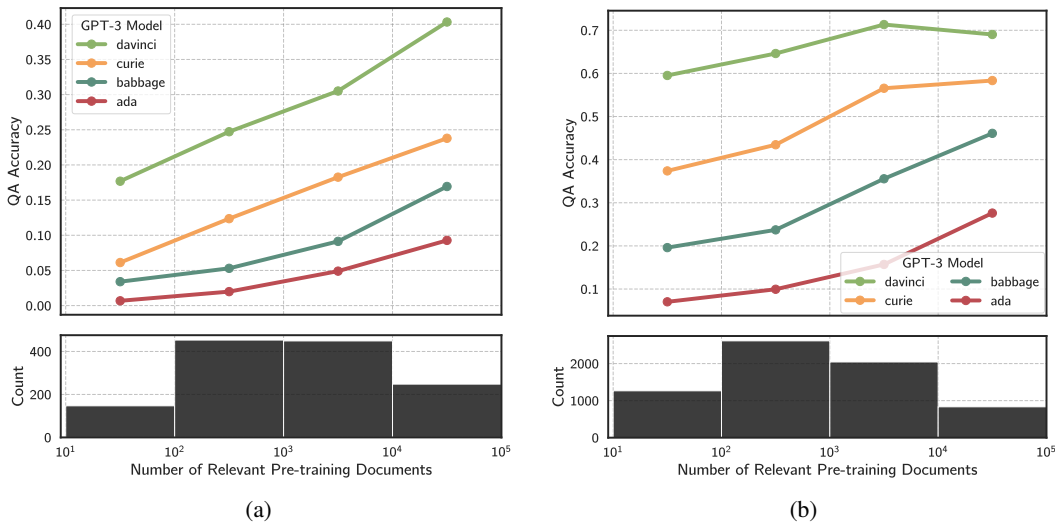

(a)   (b)

Figure 11. We present the QA results for GPT-3, with Natural Questions shown in **(a)** and TriviaQA shown in **(b)**. The trends match those seen in BLOOM and GPT-Neo. Note that our estimates for the number of relevant pre-training documents may be inaccurate because the training data for GPT-3 is not public.

B. Additional Results: Model Scaling

In this section we show additional results for how long-tail QA accuracy scales with model size for the BLOOM model family on TriviaQA and the GPT-Neo model family on Natural Questions and TriviaQA (Figure 12). The log-linear trend matches the results shown in the main text.
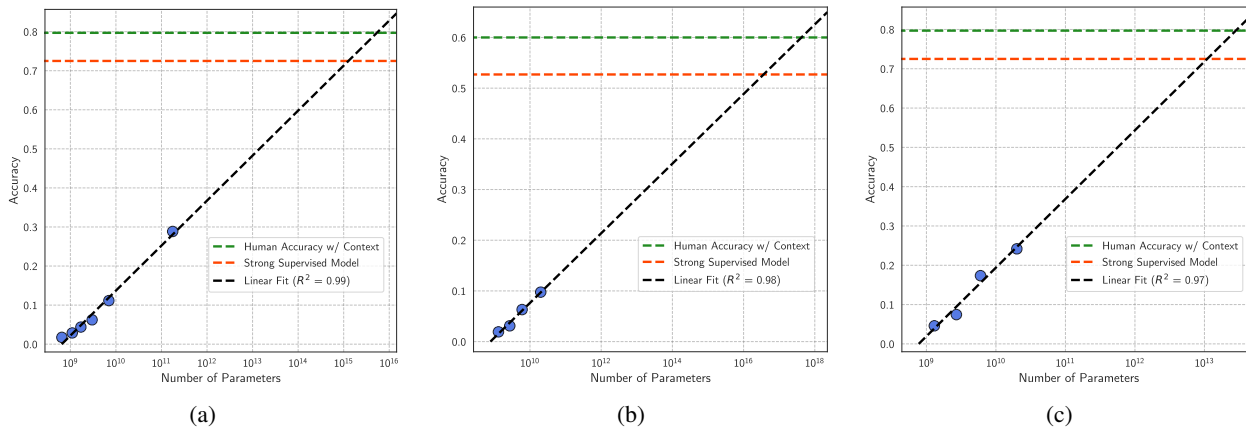

(a)   (b)   (c)

Figure 12. We present additional scaling laws for BLOOM on TriviaQA **(a)**, and GPT-Neo on Natural Questions **(b)** and TriviaQA **(c)**. All the trends are similar—we will need to scale up models dramatically to reach high QA accuracy—but the exact degree to how much we would need to scale models changes across the different settings.

C. Relevant Document Counting Heuristics

In this section, we analyze the difference between our relevant document heuristic, which counts documents where the salient question and answer entity co-occur, compared to two simple baselines: counting documents containing the question entity and documents containing the answer entity. In Figure 13(a) we show that all three document counting heuristics are correlated with QA accuracy. However, as seen in Figure 13(b) the correlation of the two baseline counting methods with QA accuracy disappears when only considering QA examples where the question and answer entity co-occur few (< 5) times in the pre-training data. Thus, these baseline counting heuristics appear correlated with QA accuracy simply because they are simply correlated with question and answer entity co-occurrence (i.e., common entities tend to co-occur with other entities more frequently) rather than causally related to QA performance.
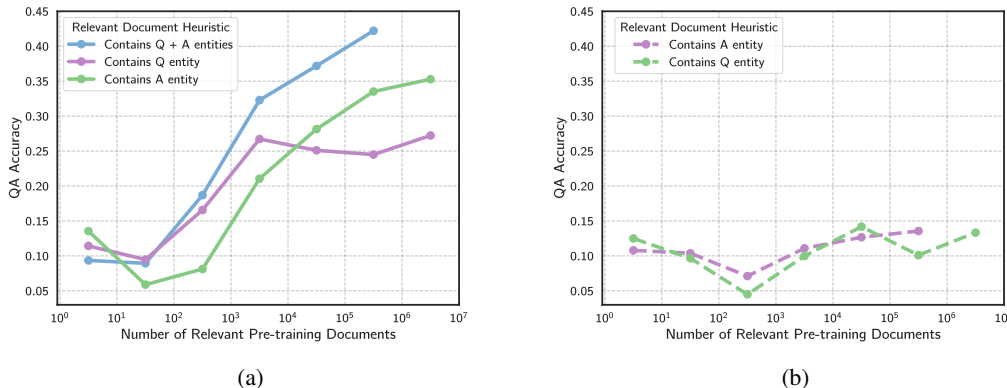

(a)   (b)

Figure 13. In **(a)**, we plot the relationship between model accuracy and the count of the question entity alone, as well as the answer entity alone. QA accuracy increases as both of these counts increase. In **(b)**, we consider only f QA pairs with few question and answer entity co-occurrences (< 5 documents). For this subpopulation of QA pairs, neither of the baseline heuristics are correlated with QA accuracy.