When *Not* to Trust Language Models: Investigating Effectiveness and Limitations of Parametric and Non-Parametric Memories

Alex Mallen[*◇] **Akari Asai**[*◇] **Victor Zhong**[◇] **Rajarshi Das**[◇]
Hannaneh Hajishirzi[◇♡] **Daniel Khashabi**[♠]
[◇]University of Washington [♠]Johns Hopkins University
[♡]Allen Institute for AI
{atmallen,akari,vzhong,rajarshi,hannaneh}@cs.washington.edu
danielk@jhu.edu

Abstract

Despite their impressive performance on diverse tasks, large language models (LMs) still struggle with tasks requiring rich world knowledge, implying the limitations of relying solely on their parameters to encode a wealth of world knowledge. This paper aims to understand LMs' strengths and limitations in memorizing factual knowledge, by conducting large-scale knowledge probing experiments of 10 models and 4 augmentation methods on PopQA, our new open-domain QA dataset with 14k questions. We find that LMs struggle with less popular factual knowledge, and that scaling fails to appreciably improve memorization of factual knowledge in the tail. We then show that retrieval-augmented LMs largely outperform orders of magnitude larger LMs, while unassisted LMs remain competitive in questions about high-popularity entities. Based on those findings, we devise a simple, yet effective, method for powerful and efficient retrieval-augmented LMs, which retrieves non-parametric memories only when necessary. Experimental results show that this significantly improves models' performance while reducing the inference costs.[1]

1 Introduction

Large language models (LMs; Brown et al. 2020; Raffel et al. 2020) have been shown to be competitive on diverse NLP tasks, including knowledge-intensive tasks that require fine-grained memorization of factual knowledge (Chowdhery et al., 2022; Yu et al., 2022). Meanwhile, LMs are shown to have limited memorization for less frequent entities (Kandpal et al., 2022), are prone to hallucinations (Shuster et al., 2021), and suffer from temporal degradation (Luu et al., 2022; Jang et al., 2022). Incorporating *non-parametric knowledge* (i.e., retrieved text chunks) largely helps address those issues stemming from reliance on LM's *parametric knowledge* (i.e., knowledge stored in their parameters) (Izacard et al., 2022b), but it is unclear whether it is strictly superior or complementary to parametric knowledge. Understanding when we should *not* trust LMs' outputs is also crucial to safely deploying them in real-world applications (Kadavath et al., 2022).

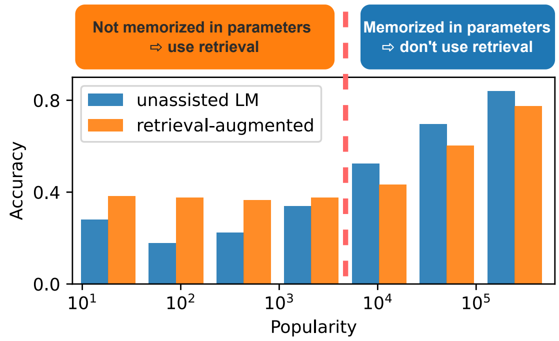

Figure 1: Relationship between subject entity popularity in a question and GPT-3 `davinci-003` performance in entity-centric open-domain question answering, with and without retrieved passages from BM25.

This work conducts a large-scale knowledge probing of LMs on factual knowledge memorization, to understand when we should and should *not* rely on LMs' parametric knowledge, and how scaling and non-parametric memories (e.g., retrieval-augmented LMs) can help. In particular, we aim to address the following research questions:

(RQ_1) How much factual knowledge is memorized by LMs and what factors affect the memorization? (Section 3)

(RQ_2) To what extent can non-parametric memories alleviate the shortcomings of parametric memories of LMs? (Section 4)

(RQ_3) Can we build a system to adaptively combine non-parametric and parametric memories? (Section 5)

We hypothesize that factual knowledge that is often discussed on the web is easily memorized by

[1]Our code and data is available at `https://github.com/AlexTMallen/adaptive-retrieval`. The first two authors contributed equally.

LMs, while the knowledge that is less discussed may not be well captured by LMs and thus they require retrieving non-parametric memories. We evaluate ten large language models with varying scales, namely OPT 1.3B, 2.7B, 6.7B, 13B (Zhang et al., 2022), GPT-3 `text-davinci-002` and `003` (Brown et al., 2020), and GPT-neo 1.3B, GPT-neo 2.7B, GPT-j 6B, and GPT-neox 20B (Radford et al., 2019) on our new open-domain questing answering (QA) datasets in a zero- or few-shot prompting manner. To enable a fine-grained analysis of entities and relationship types, we first construct a new large-scale dataset, PopQA, consisting of 14k questions of entities sampled from Wikidata, to cover factual information in the long tail that might have been missed in popular existing QA datasets (Kwiatkowski et al., 2019).

Experimental results on PopQA demonstrate that LMs' memorization (RQ_1) is often limited to the most popular entities and certain relationship types and that even GPT-3 `davinci-003` fails to answer the majority of the long-tail questions. We found that on those questions, scaling up models does *not* significantly improve the performance (e.g., for the 4,000 least popular questions, GPT-j 6B has 16% accuracy and GPT-3 `davinci-003` has 19% accuracy). This might suggest that prior trends of scaling for fact learning (Kandpal et al., 2022; Brown et al., 2020) can overestimate the effectiveness of scales in long-tail distributions.

We next investigate whether a semi-parametric approach that augments LMs with retrieved evidence can mitigate the low performance on questions about less popular entities (RQ_2). Our experimental results indicate augmenting LMs with non-parametric memories largely improves the performance on long-tail distributions across models. Specifically, we found that retrieval-augmented LMs are particularly competitive when the subject entities are not popular: Contriever-augmented GPT-neo 2.7B outperforms GPT-3 `davinci-003` on the 4,000 least popular questions. We also, surprisingly, find that retrieval augmentation can hurt the performance on questions about entities with high-popularity as retrieved context can mislead powerful LMs.

As a result, we devise a simple-yet-effective retrieval-augmented LM method, which adaptively combines parametric and non-parametric knowledge based on the predicted likelihood of failure (RQ_3). Our adaptive retrieval method further improves the performance on PopQA by up to 10%, while significantly reducing the inference costs (e.g., reducing the GPT-3 API costs by half).

2 Background and Evaluation Setup

This section provides an overview of our methodologies to understand language models' factual knowledge memorization.

2.1 Background

Prior work has shown that large LMs can memorize certain knowledge in their parameters (*parametric memories*) learned during pretraining on large-scale corpora (Petroni et al., 2019; Brown et al., 2020) even without additional training. On the other hand, its reliance on their parameters to encode a wealth of world knowledge learned has several limitations: they often struggle to memorize less popular knowledge and can result in biased predictions (Li et al., 2020; Parrish et al., 2022) and their knowledge gets obsolete quickly. To overcome those limitations, retrieval-augmented LMs leverage *non-parametric memories* stored as text and incorporate such non-parametric knowledge to enhance LMs' abilities on knowledge-intensive tasks (Izacard et al., 2022b). This work aims at understanding the capabilities and limitations of parametric knowledge, and the scenarios where non-parametric knowledge is particularly helpful.

2.2 Focus and Evaluation Setup

Focus: factual knowledge. Among diverse types of world knowledge, this work focuses on factual knowledge (Adams, 2015) of entities—knowledge about specific details of the target entities. We define factual knowledge as a triplet of (subject, relationship, object) as in Figure 2 top left. We consider a model memorizes factual knowledge when it generates the object entity given a subject entity and a relationship type.

Task format: open-domain QA. We formulate the task as an open-domain QA—given a question, a model predicts an answer without any pre-given ground-truth paragraph (Roberts et al., 2020). We focus on both the *closed-book* QA setting (Roberts et al., 2020), which relies on LMs' non-parametric memories, as well as the *open-book* QA setting, where we leverage non-parametric memories by retrieving relevant text chunks at inference time. Some work conducts knowledge probing of encoder-only models by filling out `[MASK]`

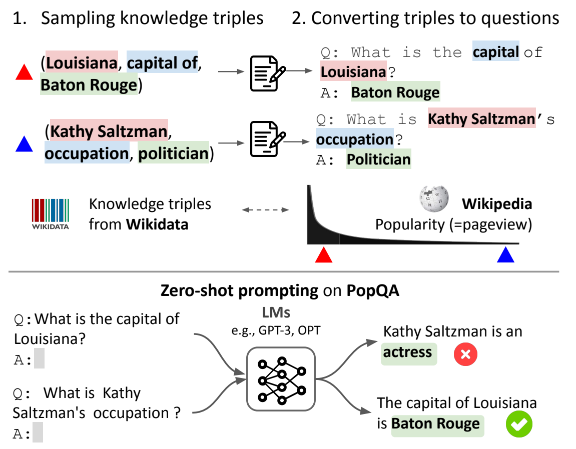

Figure 2: PopQA is created by (1) sampling diverse knowledge triples from Wikidata and (2) converting them to natural language questions. Given these questions, we prompt LMs to check if they can correctly generate the gold answers without any fine-tuning.

tokens (Petroni et al., 2019). This work uses decoder-only models[2] and thus does not use this `fill-in-the-blank` scheme.

Metrics: accuracy. We use accuracy as our primary metric for evaluating performance, judging whether the correct answer is included in a predicted sequence as in Kandpal et al. (2022).

2.3 Dimensions of Analysis

We hypothesize that factual knowledge that is not frequently discussed on the web may not be memorized well. Complementary to prior work focusing on answer entity distributions (Févry et al., 2020; Kandpal et al., 2022) or arithmetic operations (Razeghi et al., 2022), this work focuses on two key aspects of factual knowledge: subject entity popularity and relationship type.

Subject entity popularity. Prior work often uses the occurrence of entities in the pretraining corpus, which requires massive computations to link entities over billions of tokens, or noisy estimations.[3] This work, instead, uses the popularity of the entities measured by Wikipedia monthly page views as a proxy for how frequently the entities are likely to be discussed on the web.

[2]We did not explore widely-used encoder-decoder models such as T5, as their supervised pretraining consists of QA.

[3]Moreover, several recent models like GPT-3 do not release their pretraining corpora, and it is an open question whether the frequencies in pretraining corpora reflect the frequencies in their private corpora.

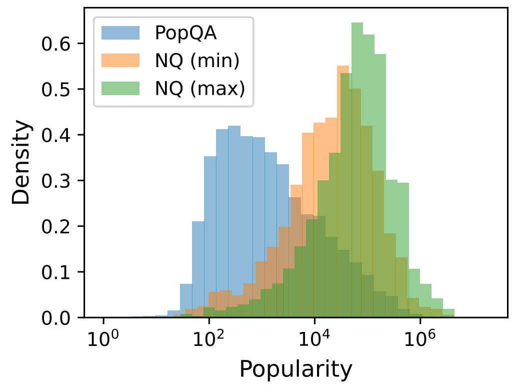

Figure 3: Distribution of subject entity popularity for PopQA and for NQ-open for reference. NQ may have multiple entities so the distribution of the most and least popular entity per question is shown. The entity-linking pipeline for NQ can be found in Appendix A.

Relationship type. We also consider the relationship types as key factors for factual knowledge memorization. For example, even given the same combinations of the subject and object entities, model performance can depend on the relationship types;[4] relationship types widely discussed can be easier to be memorized, while types that are less discussed may not be memorized much.

2.4 Benchmark: PopQA

Construction. In this work, we construct a new large-scale entity-centric QA dataset including *truly* long-tail distribution. PopQA is created by sampling knowledge triples from Wikidata and converting them into natural language questions (Figure 2). In particular, we first sample 16 diverse relationship types and randomly choose knowledge triples including the relationship types from Wikidata. Following Sciavolino et al. (2021), we verbalize a knowledge triple (S, R, O) into a question that involves substituting the subject S of the knowledge triple into a template manually written for the relationship type R. The full list of templates is found in Table 2 of the Appendix. The set of acceptable answers to the question is the set of entities E such that (S, R, E) exists in the knowledge graph. We tried various templates early on in the research process and found that the results were fairly robust to the template.

[4]For example, answering "Which country does Joe Biden serves as the president for?" is relatively easier than answering "In which country did Joe Biden go to high school?" even though they share the same subject and object (answer).

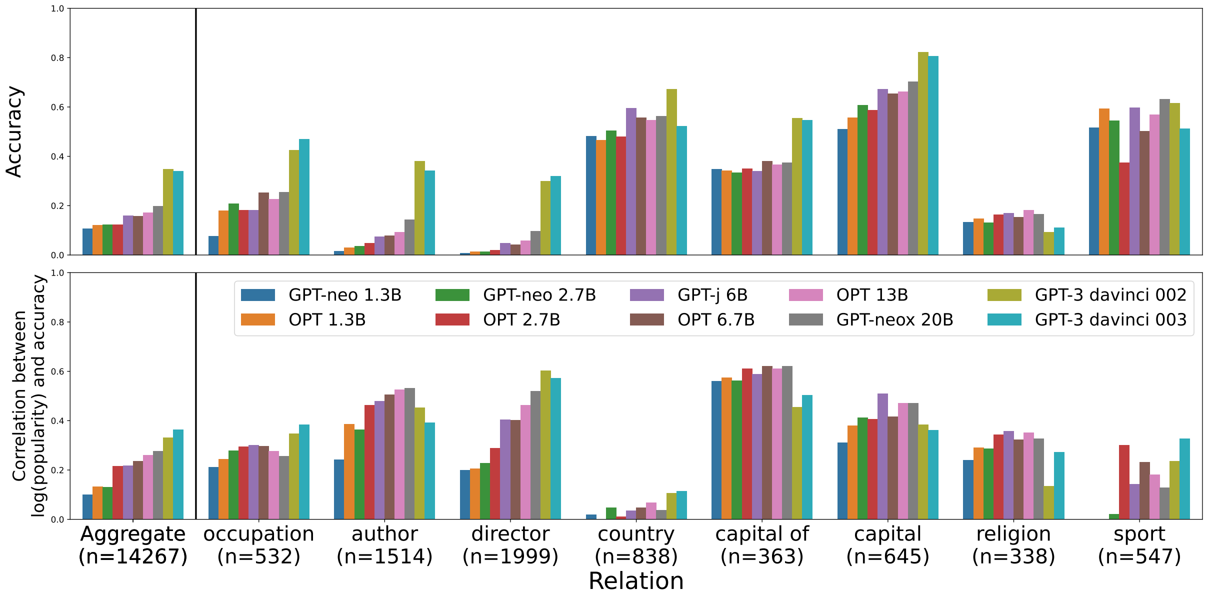

Figure 4: Per relationship type results on PopQA by model, showing overall accuracy and the correlation between accuracy and log popularity. n is the number of questions with the given relationship type. We see that **both subject entity popularity and relationship type are strong predictors of memorization ability across models.** The correlation with popularity **exists across relationship types and is stronger for larger LMs**. Only a representative subset of relationship types is shown here, and the complete results are in Figures 13 and 14 in Appendix C.1.

Comparison to existing datasets. PopQA is constructed to sample more heavily from the tail and has significantly more low-popularity entities than popular QA benchmarks such as Natural Questions (NQ; Kwiatkowski et al. 2019), which can be seen in Figure 3. PopQA is also grounded to a knowledge base (i.e., Wikidata), with links to the subject entities allowing for analysis of popularity, unlike other entity-centric benchmarks such as EntityQuestions (Sciavolino et al., 2021). The question format should be suitable for generative prediction with realistic multi-token targets, which rules out datasets like LAMA (Petroni et al., 2019).

3 Memorization Depends on Popularity and Relationship Type

To address (RQ_1), we evaluate a range of LMs with varying numbers of parameters to quantify how much factual knowledge they memorize and how different factors such as entity popularity, relationship types, and models' size affect those memorization behaviors.

3.1 Experimental Setup

Models. We evaluate ten models with a varying scale of model size: OPT (Zhang et al. 2022; 1.3, 2.7, 6.7, and 13 billion), GPT-Neo (Black et al. 2022; 1.3, 2.7, 6, and 20 billion), and GPT-3 (Brown et al. 2020; `davinci-002`, `davinci-003`) on our benchmark without any fine-tuning.

Instructions. We use a simple template "`Q: <question> A:`" to format all of our questions for generative prediction. More sophisticated instructions were attempted in preliminary experiments but they did not significantly improve the simple template to merit using them, especially given that they may overfit the model.

K-shot samples for few-shot inference. While we use zero-shot prompting for GPT-3 to reduce API costs[5], we use 15-shot prompting for all models in the GPT-neo and OPT series. We sample stratified by relationship type to diversify the samples: for each of the 15 relationship types other than the one in the question, we sample one randomly sampled gold example.

3.2 Results

Overall model performance. The top left column of Figure 4 shows the overall performance on PopQA. As you can see, despite using any text context, larger LMs show reasonable performance: GPT-3 achieves 35% accuracy and GPT-Neo 20B shows 25% accuracy, respectively. This indicates

[5]Using 15-shot prompts for GPT-3 would cost upwards of \$3000 for the combination of vanilla, Contriever, BM25, and GenRead evaluations on `davinci-002` and `davinci-003`.

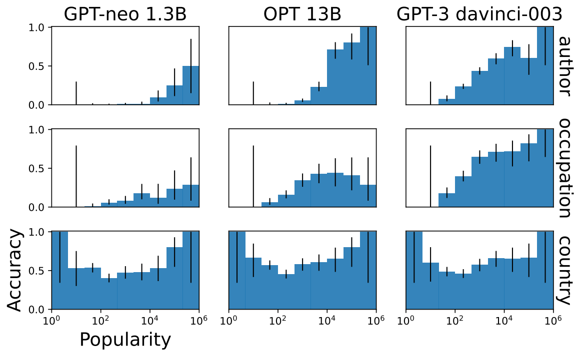

Figure 5: Memorization versus popularity for three models and relationship types. Within a relationship type, generally, there is a **monotonically increasing link between popularity and performance**, except for "country". Error bars show Wilson 95% confidence intervals.

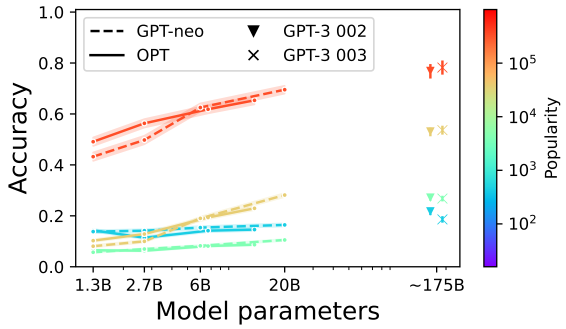

Figure 6: PopQA scaling results for LMs in different families, broken down by question popularity level. We find that most of the improvements from scaling come from **increased memorization of factual information with higher popularity.** 30 questions had popularity less than the smallest popularity bin, and are excluded to avoid showing results for small sample sizes. Error bars are 95% confidence intervals.

that large LMs memorize factual knowledge in their parameters to some extent. In this section, we take a closer look at what type of knowledge is memorized more and what affects their memorization.

Subject entity popularity predicts memorization. As shown in Figure 4 (bottom), across diverse models, we see positive correlations between the subject entity popularity and models' accuracy in almost all relationship types. This supports our hypothesis that subject entity popularity can be a good indicator of LMs' factual knowledge memorization. Overall, the correlations between subject entity popularity and accuracy are higher when the LMs are larger; GPT-3 003 shows the highest positive correlation (roughly 0.4) while GPT-Neo-1.3B shows relatively weak positive correlations, 0.1. We find that smaller LMs often rely on surface-level clues and priors on output distribution spaces in certain relationship types, without actual memorization, making the overall correlation between accuracy and subject popularity weaker. Next, we will discuss this issue in depth.

Relationship types strongly affect memorization. Although we observed positive correlations across types, some relationship types (e.g., director) show much stronger correlations while a few relationship types exhibit much weaker correlations (e.g., country). We found that this is because some relationship types can be more easily guessed, while others require factual memorization, showing stronger correlations.

Particularly, certain relationship types let models exploit surface-level artifacts in subject entity names (Poerner et al., 2020; Cao et al., 2021). For example, nationalities can be often guessed based on the entity surface. Moreover, models often output the most dominant answer entities for the questions about the relationship types with much fewer answer entities (e.g., red for the "color" relationship type).

In Figure 4, some relationship with lower correlation (e.g., color, country, sport) often shows higher accuracy, suggesting that for those relationship types, models exploit either surface-level artifacts or priors on output distribution spaces. On the other hand, on the relationship types with relatively low accuracy (e.g., occupation, author, director), larger LMs often show a high correlation (e.g., GPT-3 obtains 0.6 correlation).

Figure 5 provides a closer look at this phenomenon; it shows model accuracy over the popularity distributions for three relationship types, namely author, occupation, and country. For the first two types, we can see a clear positive trend between popularity and accuracy across models, and as the model size gets larger, the LMs memorize more. On the other hand, in the "country" relationship type, no models show trends, while overall the accuracy is high. The country relationship type is the only one of the 16 relationship types for which we don't see any correlation. This further verifies that for those relationship types, larger models actually memorize certain factual knowledge, but their memorization is strongly affected by how popular those entities are. We show example models' predictions in Appendix Section C.3.

Scaling may not help in long-tailed distributions. As seen in the left column of Figure 4, there are clear overall performance improvements with the scale on the POPQA dataset. However, Figure 6 shows that most of the improvement in parametric knowledge scaling comes from questions with high popularity. In particular, for the questions about the entities whose $\log_{10}(\text{popularity})$ is larger than 4, we can see upward trends of accuracy as the model size increases (red and yellow lines), while the performance on questions with lower popularity stays roughly constant (blue and green lines). For the 4,000 least popular questions, GPT-Neo 6B, 20B, and GPT-3 `davinci-003` have 15%, 16%, and 19% accuracy, respectively. This suggests that for distributions dominated by long-tail questions, scaling model size may not significantly improve models' factual knowledge for reasonable model scales. This somewhat dampens the findings in prior work that have shown that scaling up models significantly improves models' factual knowledge memorization (Roberts et al., 2020; Kandpal et al., 2022). We hypothesize that this is because their evaluations are often conducted on QA datasets with high-popularity entities, giving optimistic predictions about scaling.

4 Non-parametric Memory Complements Parametric Memory

Our analysis shows that even the current best language models struggle with less popular subjects or certain relationship types, and scaling up model sizes doesn't give further performance improvements. Following (RQ_2), we extend the earlier section's analysis on non-parametric sources of knowledge. Specifically, we explore the effectiveness of retrieval-augmented LMs (Borgeaud et al., 2022; Lewis et al., 2020), which enables us to leverage non-parametric memories to overcome the limitations of relying on parametric memories.

4.1 Experimental Setup

We evaluate retrieval-augmented LMs on POPQA to see the effectiveness of using non-parametric memories to overcome the shortage of parametric memories in long-tailed distributions.

Augmenting input. In this work, we try a simple retrieval-augmented LM approach, where we run an off-the-shelf retrieval system off-line to retrieve context from Wikipedia relevant to a question,[6] and then we concatenate the retrieved context with the original question. Although increasing the context size often leads to performance gains (Izacard and Grave, 2021; Asai et al., 2022), we only use the top one retrieved paragraph for simplicity.

[6]We use Wikipedia dump from 2018.

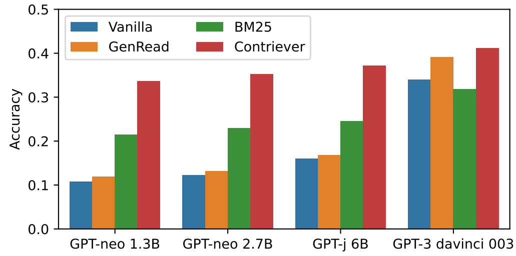

Figure 7: Accuracy of various models with BM25, Contriever, GenRead, and unassisted (vanilla). Retrieving non-parametric memories significantly improves the performance of smaller models. Complete results are found in Figure 12 of the Appendix.

Retrieval models. We use two widely-used retrieval systems: **BM 25** (Robertson et al., 2009) and **Contriever** (Izacard et al., 2022a). BM 25 is a static term-based retriever without training, while Contriever is pretrained on large unlabeled corpora. In this work, we use Contriever-MS MARCO, a Contriever fine-tuned on MS MARCO (Bajaj et al., 2016). We further experiment with a *parametric* retrieval method, **GenRead** (Yu et al., 2022), which prompts LMs to generate rather than retrieve a contextual document to answer a question. We use the ten LMs in Section 3, resulting in 40 different LMs and retrieval-augmented LMs.

4.2 Results

Retrieval largely improves performance. As can be seen in Figure 7, augmenting LMs with non-parametric memories significantly outperforms unassisted vanilla LMs. A much smaller LM (e.g., GPT-1.3 billion) augmented by the Contriever retrieval results outperforms vanilla GPT-3. Large LMs such as GPT-3 also enjoy the benefits of non-parametric memories (e.g., Contriever gives 7% accuracy gains on top of GPT-3 `davinci-003`). GenRead shows little-to-no performance improvement over vanilla parametric knowledge for smaller models, while the technique shows sizeable gains for GPT-3, especially `davinci-003`. It is unclear whether this can be attributed to scale or other differences between GPT-3 and the smaller models. It should also be noted that GenRead has potentially prohibitive inference time costs, with GPT-Neo

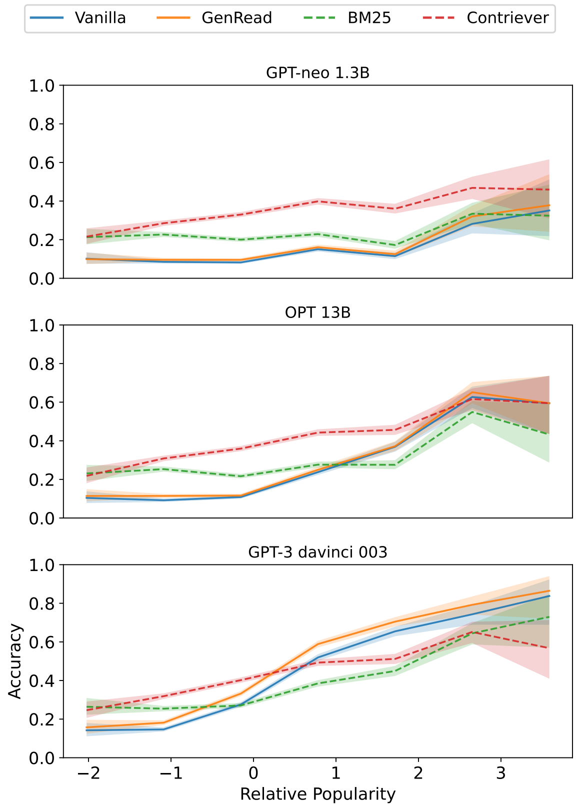

Figure 8: Positive correlations between models' accuracy and how popular a question is relative to other questions is stronger for parametric knowledge (solid lines) than for non-parametric-augmented knowledge (dashed). Relative popularity is defined as the log popularity of a question, normalized by the mean and standard deviation of log popularity for the question's relationship type. Figure 14 shows per-relationship results.

20B taking 70 seconds per query in our setup.

Non-parametric memories are effective for less popular facts. How does retrieval augmentation contribute to such significant improvements? Figure 8 shows the relationship between the entity popularity and models' QA performance. We can see that retrieval-augmented LMs guided by Contriever or BM25 clearly show their advantages over the unassisted vanilla LMs especially on the less popular entities, resulting in a significant performance gain. Overall, Contriever-guided LMs show stronger performance than BM25-based ones, while the BM25-based models show their superiority on extremely less popular entities, consistent with the findings from Sciavolino et al. (2021). Interestingly, GenRead generally outperforms vanilla LMs despite relying on LMs' parametric memory. This demonstrates the effectiveness of elicitive prompting (Wei et al., 2022; Sun et al., 2022) as observed in prior work. As in vanilla LMs, GenRead shows low performance on less popular entities. On the other hand, for more popular entities, parametric knowledge shows higher accuracy, indicating that the LMs have already memorized the answers, and augmenting input with retrieved-context doesn't help much or even hurts the performance.

	Contriever-augmented LM succeeded	Contriever-augmented LM failed
LM succeeded	0.83 (24%)	0.14 (10%)
LM failed	0.88 (17%)	0.11 (49%)

Table 1: The recall@1 of Contriever for questions which GPT-3 `davinci-003` answered correctly and incorrectly with and without retrieval. The percent of questions falling in each category is shown in parentheses. **For 10% of questions, retrieval is *harmful*, which can largely be explained by a poor retrieval recall of 0.14**. For the 17% of questions where retrieval causes the LM to counterfactually answer correctly, recall is especially high (0.88) compared to overall recall (0.42).

Non-parametric memories can mislead LMs. We conduct an in-depth analysis of why retrieval-augmented models suffer in more popular entities. We hypothesize that retrieval results may not always be correct or helpful, and can mislead LMs.

To verify this hypothesis, we first group the questions based on whether unassisted LMs predict correctly or not, and whether retrieval-augmented predictions are correct or not, and then calculate recall@1, which evaluates if a gold answer is included or not in the top 1 document (Karpukhin et al., 2020). Recall@1 for each group has been shown in Table 1, with percentages of the questions falling into each of the four categories. As can be seen, for 10% of questions, retrieval-augmentation causes the LM to incorrectly answer a question it could otherwise answer correctly. We found that on those questions, recall@1 is significantly lower than the overall recall@1 (0.14 v.s. 0.42 overall), indicating that failed retrieval can result in performance drops. Conversely, for the 17% of questions for which retrieval causes the LM to correctly answer a question it would otherwise have failed to answer, the recall@1 is 0.88. We include examples of both cases in Appendix Section C.3.

5 Adaptive Retrieval: Using Retrieval Only Where It Helps

While incorporating non-parametric memories helps in long-tail distributions, powerful LMs have

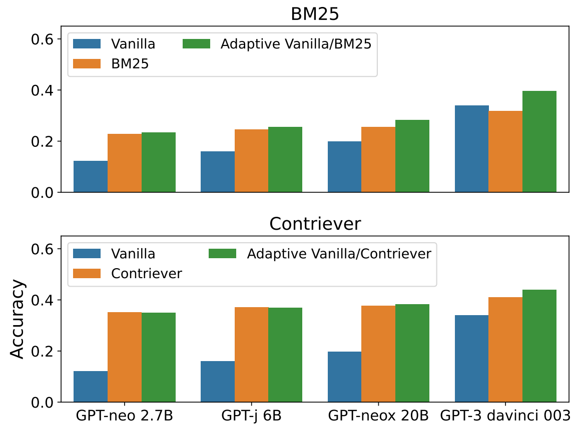

Figure 9: PopQA performance of GPT-neo models and GPT3 `davinci-003`, without retrieval, with retrieval, and with adaptive retrieval. **Retrieval improves upon parametric memory significantly**, and, in the case of Contriever, allows smaller models to outperform even GPT-3's parametric memory. Further, **adaptively retrieving robustly outperforms the retrieval-only approach**. As seen in Figure 10, small models in the adaptive setup will mostly rely on retrieval, which leads to similar results in these cases.

already memorized factual knowledge for popular entities, and retrieval augmentation can be harmful. As stated in (RQ_3), can we get the best of both worlds? We devise a simple-yet-effective method, Adaptive Retrieval, which decides when to retrieve from input query information only, and augments the input with retrieved non-parametric memories if necessary. We show that this is not only more powerful than LMs or retrieval-augmented LMs always retrieving context, but also more efficient than the standard retrieval-augmented setup.

5.1 Method

Adaptive retrieval is based on our findings: as the current best LMs have already memorized more popular entity questions, we can use retrieval only when they do not memorize the factual knowledge and thus need to find external non-parametric knowledge. In particular, we use retrieval for questions whose popularity is lower than a threshold (*popularity threshold*), and for more popular entities, do not use retrieval at all.

Using a development set, the threshold is chosen to maximize the adaptive accuracy, which we define as the accuracy attained by taking the predictions of the retrieval-augmented system for questions below the popularity threshold and the predictions based on parametric knowledge for the rest. We also consider relationship types and determine the popularity threshold independently for each relationship type.

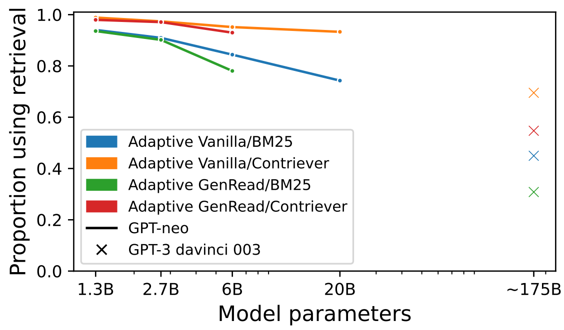

Figure 10: The proportion of questions for which various models use retrieval in the adaptive retrieval setup. When using adaptive retrieval, small models must still rely on non-parametric memory for most questions, while larger models have more reliable parametric memories enabling them to use retrieval less often.

5.2 Results

Adaptive retrieval improves performance. Figure 9 shows the results when we adaptively retrieve non-parametric memories based on the per-relationship type thresholds. We can see that adaptively retrieving non-parametric memories is effective for larger models. In particular, the best performance on PopQA is using GPT-3 `davinci-003` adaptively with GenRead and Contriever. It attains 46.5% accuracy, 5.3 points higher than any non-adaptive method.

The threshold shifts with LM scale. While adaptive retrieval show performance gains for larger models, smaller models do not realize the same benefits; as shown in Figure 9, the performance gain from adaptive retrieval is much smaller when we use models smaller than 10 billion. Why does this happen? Figure 10 shows that this is because smaller models still rely mostly on retrieval even when using adaptive retrieval. This indicates that there don't exist many questions in our dataset with entities popular enough for small LMs' parametric knowledge to be more reliable than non-parametric memory. Large models typically retrieve much less. For example, GPT-3 `davinci-003` only retrieves for 40% of questions, and even the much smaller GPT-NeoX 20B does not retrieve for more than 20% of the questions.

Adaptive retrieval reduces costs and inference-time latency. In addition to stronger performance, adaptive retrieval is also effective to improve efficiency; if we know we do not need to

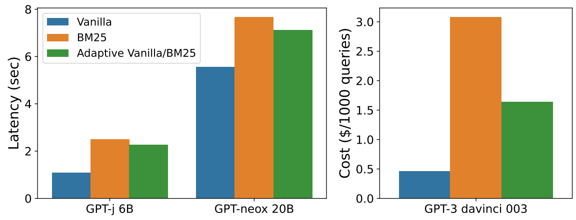

Figure 11: Latency for large GPT-neo models that were run on our machines, and API costs for GPT3. **Adaptive retrieval reduces latency and API costs.**

retrieve documents, we can skip retrieval components and the input length get shorter, which improves latency in both retrieval and language model components. Figure 11 shows the inference latency of GPT-J 6B and GPT-neox 20B, and API costs of GPT-3. As you can see, especially for larger LMs, concatenating retrieved context results in significantly increased costs of latency (e.g., for GPT-J 6B, the inference time latency almost doubles). Adaptive retrieval enables reducing the inference times up to 9%, demonstrating that it's not only more competitive but also more efficient in terms of inference time costs.

6 Related Work

Understanding memorization. Carlini et al. (2022) establish a positive relationship between pretraining term frequency and string memorization. Razeghi et al. (2022) confirm this memorization trend for simple arithmetic problems. Concurrent to our work, Kandpal et al. (2022) show that the co-occurrence of the subject entities and object entities has positive correlations with models' prediction accuracy, demonstrating consistent memorization behaviors in the area of entity factual knowledge. Févry et al. (2020) show similar phenomena in fine-tuned models. This work, instead, uses popularity to obtain a proxy for how frequently an entity is likely to be discussed on the web. Importantly, by constructing a new dataset, we conduct more fine-grained controlled experiments across the entities sampled from the long-tail popularity distributions and diverse relationship types, which might have been missed in prior analysis using existing open QA datasets. We further analyze the effectiveness and limitations of retrieval-augmented LMs, resulting in a new simple-yet-effective approach, adaptive retrieval. Recent work also shows that by prompting LMs, one can even retrieve the knowledge memorized in their parameters (*parametric knowledge*) as evidence to be directly used for downstream QA tasks (Yu et al., 2022). This work shows that those generative retrieval models may suffer in long-tail distributions as they are still relying on LMs' parametric knowledge.

Retrieval-augmented LMs. Relying solely on their parameters to encode a wealth of world knowledge requires a prohibitively large number of parameters (Roberts et al., 2020; Brown et al., 2020) and the knowledge can get obsolete quickly (Kasai et al., 2022; Jang et al., 2022). Recent work shows that augmenting LMs with non-parametric memories (i.e., retrieved text chunks) enables much smaller models to match the performance of larger models (Izacard et al., 2022b). In addition to augmenting LMs with retrieved text chunks in input spaces, several recent works study the effectiveness of incorporating retrieved non-parametric memories in intermediate states or output spaces (Zhong et al., 2022; Khandelwal et al., 2020; Min et al., 2022), which are shown to be effective to overcome LMs' limited memorization or knowledge updates. We conduct an in-depth analysis of when and how retrieval augmentation helps and demonstrates that while it helps in long-tail distributions, it can also mislead LMs. Based on these findings we propose and empirically show the potential effectiveness of adaptive retrieval, pointing to avenues for future research for more efficient and reliable retrieval-augmented LMs.

Calibration. Predicting the reliability of models' outputs is an outstanding issue in QA (Rajpurkar et al., 2018; Asai and Choi, 2021), LM memorization (Jiang et al., 2020), and the wider NLP and machine learning community. Despite its impressive performance on wider tasks, large LMs often hallucinate while their outputs often look plausible, making it difficult to understand when their outputs are unreliable (Liu et al., 2022). Recently, Kadavath et al. (2022) show that large LMs can predict when they do not know the answers to some extent, especially after fine-tuning on in-domain data. In this work, we show that adaptively combining retrieval based on when LMs are likely to fail can be an efficient and powerful approach, and those learned calibrations can be integrated with our adaptive retrieval framework.

7 Discussions and Conclusion

This work conducts large-scale knowledge probing experiments across 40 LMs and retrieval-augmented LMs to see the effectiveness and limitations of relying on LMs' parameters to memorize factual knowledge, and to understand what factors affect factual knowledge memorization. We show that their memorization has a strong correlation with the entity popularity, and on the long-tail distributions, scaling up models may only give marginal improvements. We demonstrate that non-parametric memories can largely help LMs on those long-tail distributions, but they can mislead LMs on the questions about well-known entities as the powerful LMs have already memorized them in their parameters. Based on those findings, we devise simple-yet-effective adaptive retrieval, which only retrieves when necessary, using a heuristic based on entity popularity and relationship types. Our experimental results show that this method is not only more powerful than LMs or previous retrieval-augmented LMs but also more efficient in terms of inference time costs.

Although this work focuses on entity-centric open-domain QA, we conjecture that the concept of the subject topic (entity) as well as the aspect (relationship type) is generally applicable, and future work can quantify memorization following our scheme. In this work, we show that we can find a good threshold for adaptive retrieval even just from static information appearing in the input. A more sophisticated approach to quantifying when LMs may fail to memorize can lead to further performance improvements in adaptive retrieval.

Acknowledgements

We thank the UW NLP group members for their helpful discussions, and Joongwon Kim, Wenya Wang, and Sean Welleck for their insightful feedback on this paper. This research was supported by NSF IIS-2044660, ONR N00014-18-1-2826, a Sloan fellowship, and gifts from Allen Institute for AI. AM is funded by a Goldwater Scholarship.

References

Nancy E Adams. 2015. Bloom's taxonomy of cognitive learning objectives. *Journal of the Medical Library Association*.

Akari Asai and Eunsol Choi. 2021. Challenges in information-seeking QA: Unanswerable questions and paragraph retrieval. In *Annual Meeting of the Association for Computational Linguistics* (ACL).

Akari Asai, Matt Gardner, and Hannaneh Hajishirzi. 2022. Evidentiality-guided generation for knowledge-intensive NLP tasks. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. MS MARCO: A human generated machine reading comprehension dataset.

Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, Usvsn Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*. Association for Computational Linguistics.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack Rae, Erich Elsen, and Laurent Sifre. 2022. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*.

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

Boxi Cao, Hongyu Lin, Xianpei Han, Le Sun, Lingyong Yan, Meng Liao, Tong Xue, and Jin Xu. 2021. Knowledgeable or educated guess? revisiting language models as knowledge bases. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. PaLM: Scaling language modeling with pathways.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. LLM. int8 (): 8-bit matrix multiplication for transformers at scale.

Paolo Ferragina and Ugo Scaiella. 2010. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). *Proceedings of the 19th ACM international conference on Information and knowledge management*.

Thibault Févry, Livio Baldini Soares, Nicholas FitzGerald, Eunsol Choi, and Tom Kwiatkowski. 2020. Entities as experts: Sparse memory access with entity supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022a. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*.

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

Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. 2022. Temporalwiki: A lifelong benchmark for training and evaluating ever-evolving language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How Can We Know What Language Models Know? *Transactions of the Association for Computational Linguistics*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know.

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

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A Smith, Yejin Choi, and Kentaro Inui. 2022. Realtime QA: What's the answer right now?

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*.

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

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*.

Tao Li, Daniel Khashabi, Tushar Khot, Ashish Sabharwal, and Vivek Srikumar. 2020. UnQovering Stereotypical Biases via Underspecified Questions. In *Conference on Empirical Methods in Natural Language Processing* (EMNLP) - *Findings*.

Tianyu Liu, Yizhe Zhang, Chris Brockett, Yi Mao, Zhifang Sui, Weizhu Chen, and Bill Dolan. 2022. A token-level reference-free hallucination detection benchmark for free-form text generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A Smith. 2022. Time waits for no one! analysis and challenges of temporal misalignment. In *Conference of the North American Chapter of the Association for Computational Linguistics* (NAACL).

Sewon Min, Weijia Shi, Mike Lewis, Xilun Chen, Wen-tau Yih, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Nonparametric masked language model.

Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thompson, Phu Mon Htut, and Samuel Bowman. 2022. BBQ: A hand-built bias benchmark for question answering. In *Findings of the Association for Computational Linguistics: ACL 2022*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference* *on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Hong Kong, China.

Nina Poerner, Ulli Waltinger, and Hinrich Schütze. 2020. E-BERT: Efficient-yet-effective entity embeddings for BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2020*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners.

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*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

Yasaman Razeghi, Robert L Logan IV, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot reasoning.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple entity-centric questions challenge dense retrievers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*.

Zhiqing Sun, Xuezhi Wang, Yi Tay, Yiming Yang, and Denny Zhou. 2022. Recitation-augmented language models.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. 2022. Generate rather than retrieve: Large language models are strong context generators.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models.

Zexuan Zhong, Tao Lei, and Danqi Chen. 2022. Training language models with memory augmentation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Appendix

A Details of PopQA Constructions

List of the relationship types and templates. In this work, we use the following 16 relationship types, and the authors of this paper manually annotated templates to verbalize knowledge triple to natural language questions. We show the final list of the templates used to create PopQA in Table 2.

Figure 3 shows the distribution of subject popularity of PopQA versus the popular NQ benchmark. Subject entities from NQ were extracted using TagMe (Ferragina and Scaiella, 2010) on the NQ-open development set with a score threshold of 0.22.

Relationship	**Template**
occupation	What is [subj] 's occupation?
place of birth	In what city was [subj] born?
genre	What genre is [subj]?
father	Who is the father of [subj] ?
country	In what country is [subj] ?
producer	Who was the producer of [subj] ?
director	Who was the director of [subj] ?
capital of	What is [subj] the capital of?
screenwriter	Who was the screenwriter for [subj] ?
composer	Who was the composer of [subj] ?
color	What color is [subj] ?
religion	What is the religion of [subj] ?
sport	What sport does [subj] play?
author	Who is the author of [subj] ?
mother	Who is the mother of [subj] ?
capital	What is the capital of [subj] ?

Table 2: Full list of the manually annotated templated used for PopQA creations. [subj] denotes a placeholder for subject entities.

Knowledge triples sampling. In the construction of the PopQA dataset, knowledge triples are sampled with higher weight given to more popular entities, otherwise the distribution would be dominated by the tail and we would not have enough high-popularity entities to complete our analysis. Specifically, when considering whether to sample a particular knowledge triple, we include the knowledge triple if and only if $f > \exp(8R - 6)$, where $R \sim U(0, 1)$ is a unit uniform pseudo-random number and f is the exact match term frequency of the subject entity's aliases in an 800 MB random sample of C4. To increase diversity, once 2000 knowledge triples of a particular relation type have been sampled, they are no longer sampled.

B Experimental Details

Computational resources and API costs. GPT-3 API usage totaled to \$275. We ran 14,282 questions through two GPT-3 `davinci` models using four different methods: vanilla experiments cost \$13 (\$0.46 per 1000 questions), Contriever-augmented experiments cost \$88 (\$3.08 per 1000 questions), BM25-augmented experiments cost \$81 (\$2.80 per 1000 questions), and GenRead experiments cost \$93 (\$3.25 per 1000 questions).

To run experiments using LMs larger than two billion parameters, we use a single V100 Volta GPU with 32GB GPU memories. We use int8bit (Dettmers et al., 2022) quantization with OPT 13 billion and GPT-Neo 20 billion models to make them fit our GPUs. In our preliminary experiments using GPT-Neo 6 billion, we did not observe a notable performance drop by using the quantization.

Details of deciding thresholds. We 75% of PopQA to determine a popularity threshold for each relation type. Using brute force search, we select the threshold to maximize the adaptive accuracy, which we define as the accuracy attained by taking the predictions of the retrieval-augmented system for questions below the popularity threshold and the predictions based on parametric knowledge for the rest.

We then evaluate adaptive accuracy using the learned thresholds on the remaining 25% of PopQA, and repeat with 100 different random splits and take the mean to obtain the reported adaptive accuracy measurement.

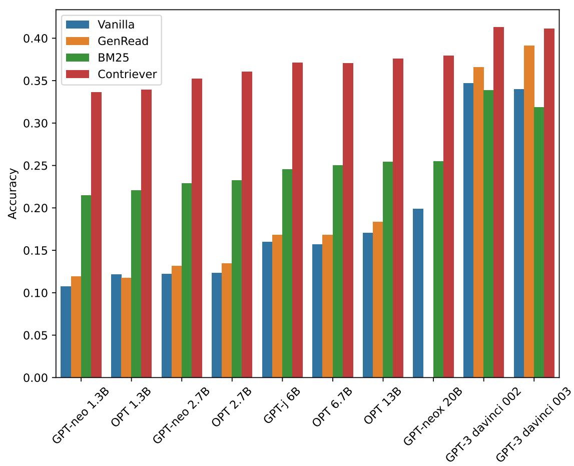

Figure 12: Accuracy by method and model. This is an extension of Figure 7

C Detailed Results on PopQA

C.1 LM results

Full results of per-relationship type accuracy and correlation. Figure 13 shows the full result of per-relationship type accuracy for all relationship types in PopQA. Figure 14 shows the correlations for all relation types.

C.2 Retrieval-augmented LM results

Full results on 16 relationship types. Figure 15 shows the full results of the retrieval-augmented LMs and unassisted LMs on 16 relationship types using three different LMs as backbones.

C.3 Qualitative Results

Table 3 shows several examples on PopQA, where GPT-3 `davinci-003` answers correctly while the Contriever-augmented version fails to answer. Along with the low recall@1 of 0.14 for this group, Table 3 suggests that the most common reason retrieval can be harmful is because it retrieves a document about a mistaken entity, such as a person with the same name as the subject, or an entity that simply is not relevant to the question (as in the case of "Noel Black").

Table 4 shows several examples on PopQA, where GPT-3 `davinci-003` answers correctly only when augmented with Contriever. The recall@1 for this case is 0.88, which is significantly higher than the overall recall. Note that in the second example the retrieval caused the LM to answer correctly, but only by coincidence: the subject entity "Pierre" actually refers to the city in South Dakota, not the Basketball player. Otherwise, retrieval appears to be helpful because it provides the relevant information directly.

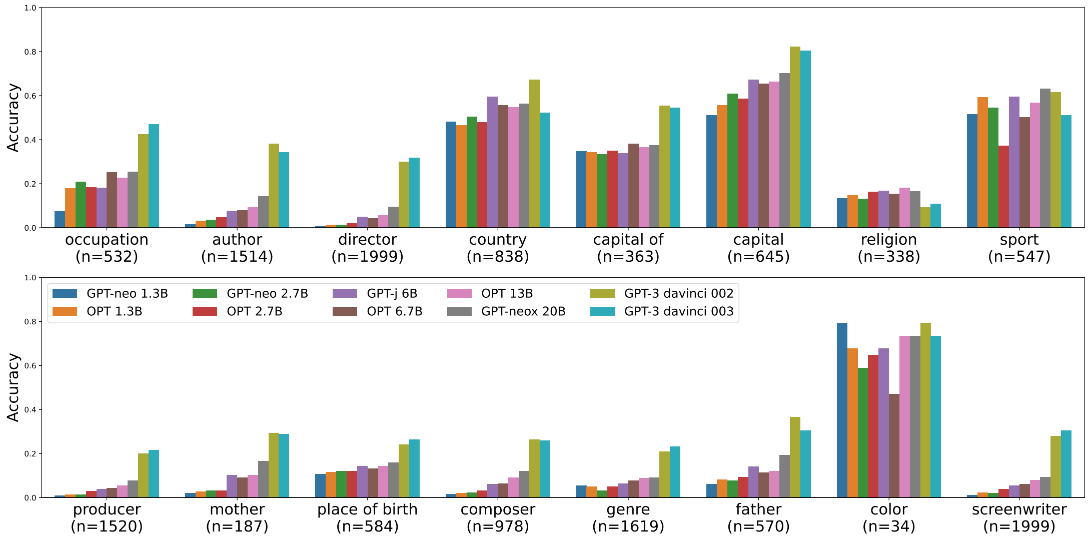

Figure 13: Accuracies for all relationship types and models. This is an extension of Figure 4.

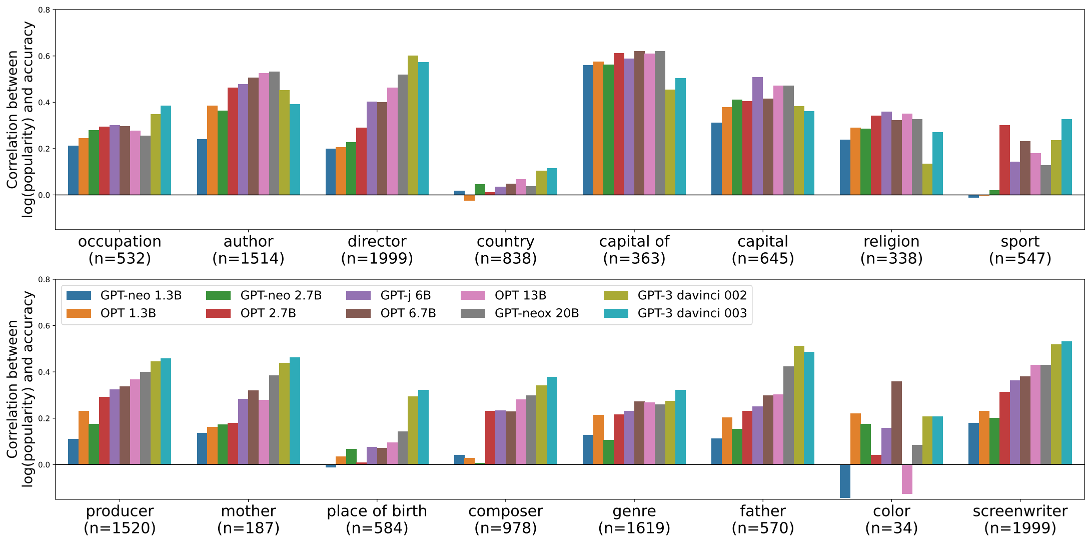

Figure 14: Correlations for all relationship types and models. This is an extension of Figure 4.

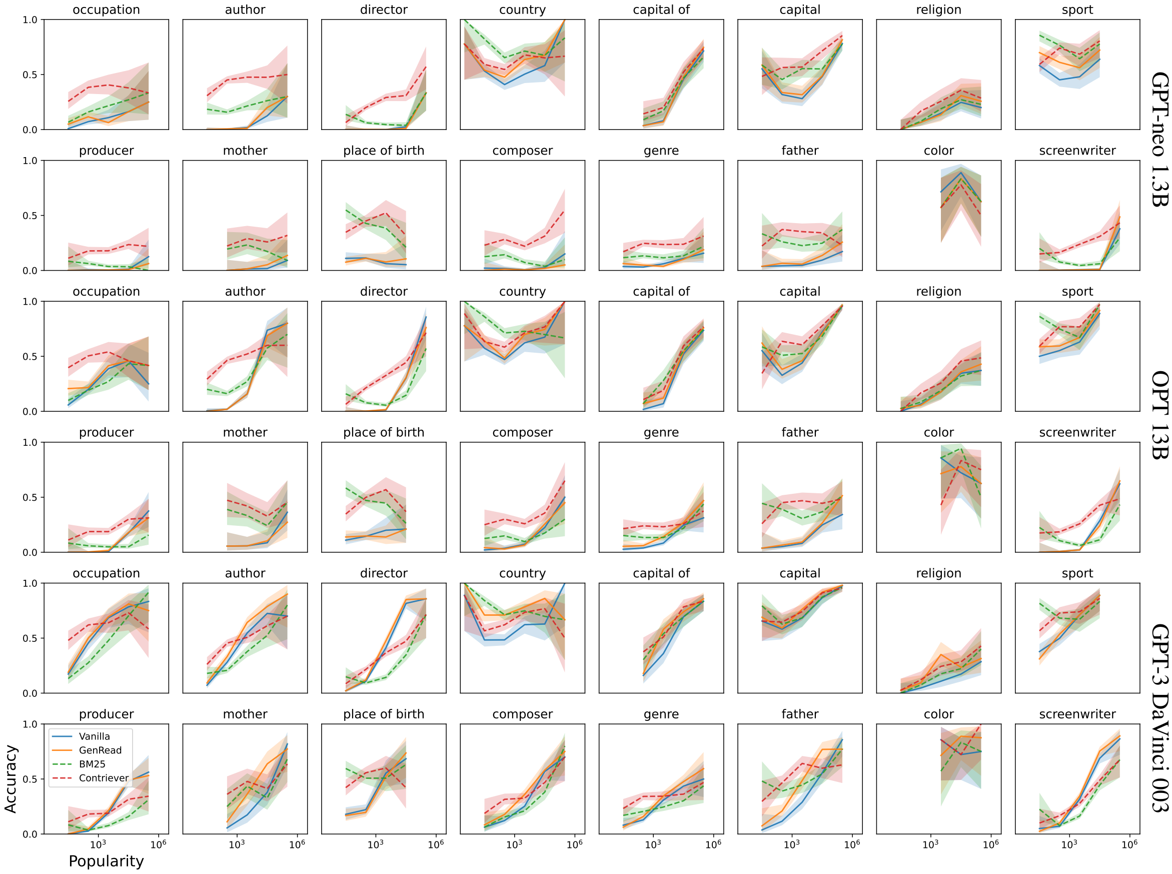

Figure 15: Accuracy versus popularity as shown in Figure 8 broken down by relation type. Popularity bins with less than 5 samples are excluded to avoid cluttering the figures with noisy results that have wide error bars.

Question (answer)	LM	Retrieval-LM	Retrieval
Who was the director of Black? (Sanjay Leela Bhansali)	The director of Black is Sanjay Leela Bhansali. ✓	Noel Black ✗	Noel Black (June 30, 1937 – July 5, 2014) was an American film and television director, screenwriter, and producer. Black was born in Chicago, Illinois. He won awards at the 1966 ... and subsequently concentrated on directing for television, occasionally directing films such as "Private School".
Who was the producer of The Faculty? (Robert Rodriguez)	The Faculty was produced by Elizabeth Avellan and Robert Rodriguez. ✓	The producer of The Faculty was Elizabeth Avellan. ✗	The Faculty is a 1998 American science fiction teen horror film written by Kevin Williamson, directed by **Robert Rodriguez**, and starring Elijah Wood,... The film was released on Christmas Day, 1998, by Miramax Films through Dimension Films. ... The film has developed a cult following in the years since its release.
What is Michael Shelley's occupation? (singer-songwriter)	Michael Shelley is a singer-songwriter and musician. ✓	Michael Shelley is an American applied mathematician. ✗	Michael Shelley (mathematician) Michael J. Shelley (born August 17, 1959) is an American applied mathematician who works on the modeling and simulation of complex systems arising in physics and biology. This has included free-boundary problems in fluids and materials science... He is also the co-founder and co-director of the Courant Institute's Applied Mathematics Lab.
In what city was Zijah Sokolović born? (Sarajevo)	Zijah Sokolović was born in Sarajevo ✓	Zijah Sokolović was born in Orahovac, Kingdom ✗	Ali Sokol ... (born 8 May 1921 in Orahovac, Kingdom of Serbs, Croats and Slovenes, died 23 September 1974) was a Yugoslav pulmonologist . Ali Sokol was born into an agricultural family. He was the third of four children of father and mother Hatixhes Solomon. It is unknown the exact date of birth but the most reliable date is May 8 year in 1921.

Table 3: Qualitative examples of the questions where only retrieval-augmented LMs *fail* to answer correctly. The blue underlined text indicates the sub-strings matching the gold answers in the retrieved context.

Question (answer)	LM	Retrieval-LM	Retrieval
What genre is Unknown? (fantasy)	Unknown is not a specific genre of music. It could refer to ✗	Unknown is a pulp fantasy fiction magazine. ✓	Unknown (magazine) Unknown (also known as Unknown Worlds) was an American pulp **fantasy** fiction magazine, published from 1939 to 1943 by Street & Smith, and edited by John W. Campbell. ... The leading fantasy magazine in the 1930s was "Weird Tales", which focused on shock and horror.
In what country is Pierre? (United States)	Pierre is in France. ✗	Pierre is from the United States. ✓	Jonas Farley Pierre (born May 30, 1980) is an American professional basketball player. A center, he has played with three teams in the National Basketball League of Canada (NBL). Pierre last played with the Saint John Mill Rats in Canada. ... He then joined Lincoln University in Missouri for his third year. In the summer of 2003, he signed a letter of intent to play with Nicholls State in the NCAA Division I.
Who was the producer of The Cocoanuts? (Walter Wanger)	The Cocoanuts was produced by Florenz Ziegfeld. ✗	The Cocoanuts was produced for Paramount Pictures by Walter Wanger, who ✓	The Cocoanuts is a 1929 musical comedy film starring the Marx Brothers. Produced for Paramount Pictures by **Walter Wanger**, who is not credited, the film stars the four Marx Brothers, Oscar Shaw, Mary Eaton, and Margaret Dumont. It was the first sound film to credit more than one director (Robert Florey and Joseph Santley), and was adapted to the screen by Morrie Ryskind from the George S. Kaufman Broadway musical play. ...
Who was the director of The White Suit? (Lazar Ristovski)	The White Suit was directed by Sachin Kundalkar. ✗	Lazar Ristovski ✓	In 1999 "The White Suit" an auteur film by Ristovski (director, writer, lead actor, and producer) was at the Cannes Film Festival in the Critics Week program. "The White Suit" was the Serbian entry for the 1999 Academy Awards. **Lazar Ristovski** is the sole owner of Zillion Film Company In 2006, he made a small appearance in the James Bond film "Casino Royale". He played Caruso in the 2004 movie "King of Thieves". He starred as Đorđe in the award-winning 2009 film "St. George Shoots the Dragon".

Table 4: Qualitative examples of the questions where only retrieval-augmented LMs *successfully* answer correctly. The blue underlined text indicates the sub-strings matching the gold answers in the retrieved context.