RAVEN: In-Context Learning with Retrieval-Augmented Encoder-Decoder Language Models

Jie Huang[1,2,*] **Wei Ping**[2] **Peng Xu**[2] **Mohammad Shoeybi**[2]
Kevin Chen-Chuan Chang[1] **Bryan Catanzaro**[2]
[1]University of Illinois at Urbana-Champaign [2]NVIDIA
jeffhj@illinois.edu, wping@nvidia.com

Abstract

In this paper, we investigate the in-context learning ability of retrieval-augmented encoder-decoder language models. We first conduct a comprehensive analysis of existing models and identify their limitations in in-context learning, primarily due to a mismatch between pretraining and inference, as well as a restricted context length. To address these issues, we propose RAVEN, a model that combines retrieval-augmented masked language modeling and prefix language modeling. We further introduce *Fusion-in-Context Learning* to enhance the few-shot performance by enabling the model to leverage more in-context examples without requiring additional training. Through extensive experiments, we demonstrate that our simple yet effective design significantly improves performance, achieving results comparable to the most advanced language models in certain scenarios, despite having substantially fewer parameters. Our work underscores the potential of retrieval-augmented encoder-decoder language models for in-context learning and encourages further research in this direction.

1 Introduction

Recent advancements in natural language processing have been predominantly driven by the development of large language models (LLMs) (Brown et al., 2020; OpenAI, 2022; 2023; Chowdhery et al., 2023; Smith et al., 2022). These models have demonstrated remarkable performance across a wide range of tasks (Qin et al., 2023; Bubeck et al., 2023; Huang & Chang, 2023). One of the key features that enables these models to excel is their ability to perform in-context learning (Dong et al., 2022). By conditioning on given context, LLMs can adapt to new tasks and domains without the need for task-specific fine-tuning. This enables LLMs to perform well on zero-shot or few-shot learning tasks, where only a limited number of examples are available.

While in-context learning has been extensively studied for decoder-only language models like GPT-3 (Brown et al., 2020) and PaLM (Chowdhery et al., 2023), research on encoder-decoder language models, which have shown to learn stronger representations (Devlin et al., 2019; Raffel et al., 2020), remains limited. Notably, Patel et al. (2023) tap into the potential of mT5 (Xue et al., 2021), a multilingual encoder-decoder LM, by iteratively prompting the model to produce long generations with in-context examples. Chung et al. (2022); Longpre et al. (2023) finetune T5 (Raffel et al., 2020) with a large mixture of tasks using instruction tuning (Mishra et al., 2022; Wei et al., 2022; Sanh et al., 2022) to improve model performance and generalization to unseen tasks in both zero-shot and few-shot settings.

On the other hand, LLMs still face challenges such as hallucination and limitations in representing the long-tail and most recent knowledge (Mallen et al., 2022; Huang et al., 2022; Luu et al., 2022; Jang et al., 2022; Zheng et al., 2023). Retrieval-augmented language models (Izacard et al., 2023; Borgeaud et al., 2022; Wang et al., 2023; Shi et al., 2023) have emerged as a powerful approach to address these issues by retrieving relevant knowledge

*Work done at NVIDIA. Code is available at https://github.com/jeffhj/RAVEN.

from an external corpus. Among these, the encoder-decoder models, such as ATLAS (Izacard et al., 2023), stand out. They benefit from the strong representation ability of a bidirectional encoder, coupled with of the efficacy of a Fusion-in-Decoder architecture (Izacard & Grave, 2021), enabling the effective integration of multiple retrieved passages. Despite these advancements, in-context learning with these models remains underexplored.

In this regard, we first conduct a comprehensive analysis of the state-of-the-art retrieval-augmented encoder-decoder language models by designing and experimenting with different prompting strategies. We find that these models exhibit a certain in-context learning ability; however, due to a mismatch between pretraining and inference and a limited context length—issues that are common to existing encoder-decoder LMs trained with masked language modeling—its few-shot performance is not stable and providing more than, e.g., 8-shot, examples does not lead to further improvement.

Based on the analysis, we develop RAVEN[1] by first mitigating the mismatch between pretraining and inference through a combination of retrieval-augmented masked language modeling and prefix language modeling. Moreover, to enable the model to learn from more in-context examples, we propose *Fusion-in-Context Learning*, a novel approach that allows the model to utilize more in-context examples without modifying the model configuration or requiring additional training. Furthermore, we suggest using the retriever of the model to obtain relevant in-context examples to further enhance few-shot performance. Our empirical results demonstrate that RAVEN significantly outperforms previous retrieval-augmented encoder-decoder LMs in both zero-shot and few-shot settings, even achieving comparable results to decoder-only LLMs in some settings despite having 180 times fewer parameters.

The main contributions of this paper are twofold:

- From an analytical standpoint, we provide a thorough analysis of the in-context learning ability of retrieval-augmented encoder-decoder language models. We demonstrate the possibilities and offer insights for future development.
- From a technological perspective, we introduce RAVEN, coupled with our Fusion-in-Context Learning and In-Context Example Retrieval strategies, building upon the analytical groundwork. These techniques, though simple, are highly effective. They not only enhance the base model's capabilities but also highlight the potential of in-context learning with retrieval-augmented encoder-decoder LMs.

2 Background and Related Work

Retrieval-augmented language models are a class of language models designed to enhance their performance by incorporating external knowledge. These models typically employ an information retrieval mechanism to access relevant information from a large corpus, which is then integrated into the model's prediction process. Retrieval-augmented LMs can be based on both encoder-decoder (Izacard et al., 2023; Lewis et al., 2020) and decoder-only (Khandelwal et al., 2020; Borgeaud et al., 2022; Shi et al., 2022) architectures. For decoder-only LMs, the computational cost typically increases quadratically with the input length, as well as with the number of retrieval passages. In contrast, for encoder-decoder LMs with a Fusion-in-Decoder architecture, the computation cost grows linearly with the number of retrieved passages, as they only perform self-attention over one passage at a time (Izacard & Grave, 2021). This concept is also investigated by Ye et al. (2023) for more efficient in-context learning.

While there has been some research on in-context learning with retrieval-augmented decoder-only LMs, which can be straightforwardly implemented by concatenating retrieved passages with the query as the input of the LM (Mallen et al., 2022; Shi et al., 2023; Khattab et al., 2022), in-context learning with retrieval-augmented encoder-decoder LMs remains unexplored to the best of our knowledge. This is despite the fact that encoder-decoder LMs can be more efficient at incorporating multiple (e.g., 40) retrieved passages.

[1]RAVEN, a bird known for its intelligence and adaptability, has the letters "RA" in its name, which represents "**R**etrieval-**A**ugmented" in our context.

Masked Language Modeling (Pretraining)	Prompting Strategy 1	Prompting Strategy 2
Input to Encoder: Machine learning algorithms build a model based on sample data,<extra_id_0> as training data, in order to<extra_id_1> being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, agriculture, and computer vision,<extra_id_2> unfeasible to develop conventional algorithms to perform the<extra_id_3> Passage: … machine learning models require a high quantity of reliable data in order for the models … **Input to Decoder:** *None* **Output:** <extra_id_0> known<extra_id_1> make predictions or decisions without<extra_id_2> where it is difficult or<extra_id_3> needed tasks.	**Input to Encoder:** Question: What is the capital of the Provence-Alpes-Cote d'Azur region of France? Answer: Marseilles Question: The Greek word Xero (pronounced zero) in xerography and related terminology means what? Answer: Dry Question: In which country was the first permanent bungee jumping site situated? Answer:<extra_id_0> Passage: … first permanent commercial bungee site, the Kawarau Bridge Bungy at the Kawarau Gorge Suspension Bridge near Queenstown in the South Island of New Zealand … **Input to Decoder:** *None* **Output:** <extra_id_0> New Zealand	**Input to Encoder:** Question: What is the capital of the Provence-Alpes-Cote d'Azur region of France? Answer:<extra_id_0> Question: The Greek word Xero (pronounced zero) in xerography and related terminology means what? Answer:<extra_id_1> Question: In which country was the first permanent bungee jumping site situated? Answer:<extra_id_2> Passage: … first permanent commercial bungee site, the Kawarau Bridge Bungy at the Kawarau Gorge Suspension Bridge near Queenstown in the South Island of New Zealand … **Input to Decoder:** <extra_id_0> Marseilles<extra_id_1> Dry **Output:** <extra_id_2> New Zealand

Figure 1: Retrieval-augmented masked language modeling and prompting strategies for in-context learning.

3 Methodology

In this section, we first explore in-context learning with retrieval-augmented encoder-decoder language models in the literature. Building upon the analysis, we develop models with enhanced zero-shot performance and improved in-context learning abilities.

3.1 In-Context Learning with Retrieval-Augmented Encoder-Decoder LMs

To investigate the in-context learning ability of retrieval-augmented encoder-decoder language models, we first aim to gain insights from the state-of-the-art designs in the literature. Among them, the design of ATLAS (Izacard et al., 2023) stands out; it combines a general-purpose dense retriever with a sequence-to-sequence reader (i.e., T5 (Raffel et al., 2020)) using the Fusion-in-Decoder architecture (Izacard & Grave, 2021). The retriever, encoder and decoder are jointly trained during the pretraining process. In this process, the dense retriever, based on the Contriever model (Izacard et al., 2022), is responsible for selecting relevant passages from an external knowledge source, e.g., Wikipedia, based on the given corrupted context. The retrieved passages are then processed along with the context by the encoder, which generates the corresponding output, i.e., the masked spans, at the decoder (Figure 1, left). ATLAS demonstrates exceptional few-shot performance on knowledge-intensive language tasks (Petroni et al., 2021), despite having a lower parameter count compared to other recent LLMs.

However, in Izacard et al. (2023), the few-shot performance is achieved by finetuning the model with few-shot examples, which requires additional training and may limit its applications, such as dealing with dynamic and diverse real-time user queries like GPT-3/4 (Brown et al., 2020; OpenAI, 2023), where in-context learning plays a vital role. Therefore, we take the initiative to explore the in-context learning ability of this type of models, using open-domain question answering (Chen et al., 2017) as a representative task for some preliminary experiments.

Prompting Strategies. To facilitate in-context learning, an effective prompting strategy is paramount. In contrast to decoder-only LMs, where the input can only be fed to the decoder, encoder-decoder LMs can take input in either the encoder or the decoder. In alignment with the pretraining objective, we identify two prompting strategies for in-context learning:

Strategy 1. The first strategy involves feeding all example question-answer pairs and the target question to the encoder, without any input to the decoder. The prompt is designed as:[2]

[2]Here we present a format designed for better demonstration. The actual prompt, which follows the template used in pretraining, can be found in Appendix B.4.

		Natural Questions				TriviaQA			
		0-shot	1-shot	5-shot	8-shot	0-shot	1-shot	5-shot	8-shot
ATLAS	11B S1	26.7	21.3	29.8	**31.3**	56.9	35.5	62.3	**63.9**
ATLAS	11B S2		21.4	16.3	9.8		49.8	48.4	44.4

Table 1: Results of ATLAS 11B with prompting strategy 1 (S1) and strategy 2 (S2).

	Natural Questions	TriviaQA
first	0.7	9.2
random	6.5	19.5
last	**29.8**	**62.3**

Table 2: Results of ATLAS 11B (5-shot) with different target question positions.

Enc: Question: q_1 Answer: a_1 ... Question: q_k Answer: a_k Question: q_0 Answer:<extra_id_0> d

where $(q_1, a_1), \ldots, (q_k, a_k)$ represent example QA pairs, q_0 denotes the target question, <extra_id_0> is a sentinel token (Raffel et al., 2020), and d is the relevant passage retrieved with q_0. An example in a 2-shot setting is illusated in Figure 1 (middle).

Strategy 2. As the decoder of the encoder-decoder model can also accept input, we can feed the answers of in-context examples to the decoder and only feed the questions to the encoder, using multiple sentinel tokens:

Enc: Question: q_1 Answer:<extra_id_0> ... Question: q_k Answer:<extra_id_$(k-1)$> Question: q_0 Answer:<extra_id_k> d

Dec: <extra_id_0> a_1... <extra_id_$(k-1)$> a_k

Figure 1 (right) demonstrates an example. The model is expected to learn from in-context examples by examining both the input to the encoder and input to the decoder.

We select two widely-used datasets in the domain of open-domain question answering for the preliminary study: Natural Questions (NQ) (Kwiatkowski et al., 2019) and TriviaQA (TQA) (Joshi et al., 2017)[3]. Table 1 summarizes the results. We find that the model struggles to learn from in-context examples using strategy 2, as the few-shot performance is worse than the zero-shot performance. We hypothesize that this is because the model has difficulty learning the pattern of S2 with masked language modeling during its pretraining, since it is unlikely to obtain several consecutive question-answer pairs (or something similar) in the form of strategy 2 by randomly masking several spans in a sequence.

On the other hand, we observe that with strategy 1, the model does exhibit some in-context learning ability, where the 5-shot and 8-shot performance is significantly better than the zero-shot performance on both NQ and TriviaQA. Therefore, we choose to focus on strategy 1 for further study and disregard strategy 2 for the remainder of the paper.

Effect of Position. As the encoder of encoder-decoder language models is bidirectional, it can also examine in-context examples that follow the target question to fill in the masked token. This means that we may position the target question at the beginning or middle of a sequence, for example:

Question: q_0 Answer:<extra_id_0> Question: q_1 Answer: a_1 ... Question: q_k Answer: a_k d

Question: q_1 Answer: a_1 ... Question: q_0 Answer:<extra_id_0>... Question: q_k Answer: a_k d

Table 2 summarizes the results. We denote the target question's position as "*first*" for the beginning of the sequence, "*random*" for a random position, and "*last*" for the original setting (S1). Interestingly, placing the target question anywhere other than the last position results

[3]Experimental setup is detailed in the Appendix B.1.

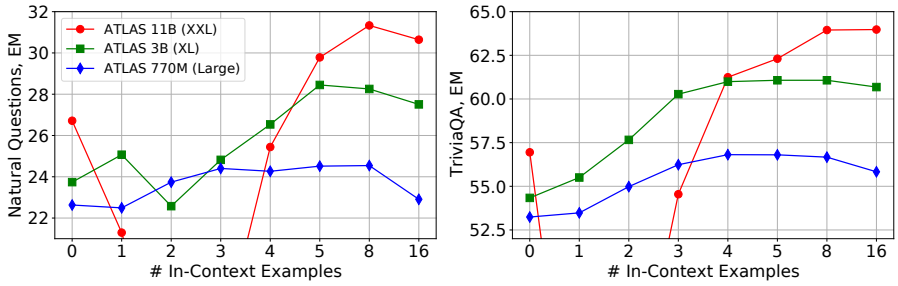

Figure 2: Results of ATLAS with different numbers of in-context examples.

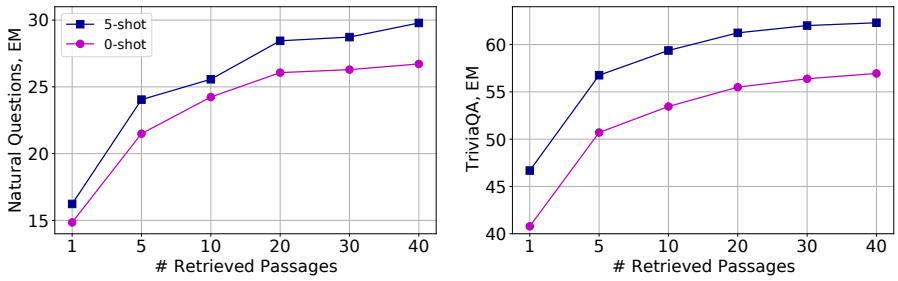

Figure 3: Results of ATLAS 11B with different numbers of retrieved passages.

in a significant performance drop. Upon examining the generated answers, we observe that when the target question is placed at the beginning or in the middle, the model tends to repeat the answer or generate additional text. For example, for the prompt "Question: What number in Bingo is sometimes referred to as Heinz varieties? Answer:<extra_id_0> Question: ...". The generated text is "57 'Heinz varieties' is a term used in Bingo to describe". This indicates that the model does not fully understand and follow the style of in-context examples. Therefore, by default, we position the target question after all the in-context examples.

Effect of Number of In-Context Examples. The number of in-context examples is a crucial hyperparameter for in-context learning. Generally, we expect better performance from a model with more in-context examples, but there is an upper limit due to 1) the maximum context length setup, e.g., 512 tokens, during the pretraining process, and 2) the point at which the model has received sufficient examples and cannot gain additional information from more examples. The optimal number of in-context examples also varies between models. For instance, on TriviaQA, PaLM (Chowdhery et al., 2023) exhibits better 1-shot performance than settings with more examples, while this is not the case for GPT-3 (Brown et al., 2020).

Figure 2 illustrates the impact of varying the number of in-context examples across different model sizes. Interestingly, the 11B model demonstrates poor performance in low-shot settings, e.g., 1-shot, but improves significantly after 4-shot and 5-shot. Upon examining the generated responses, we find that the model tends to produce answers with more tokens in low-shot settings, while the ground truth typically consists of shorter answers with fewer than 5 tokens. By relaxing the criteria for a correct prediction to include instances where the ground-truth answer is a substring of the model output, we find that the 1-shot performance surpasses that of the 0-shot setting (38.3 vs 32.1 on NQ).

All models perform well in the 5-shot and 8-shot settings, but their performance does not continue to improve with more in-context examples (e.g., 16-shot). We believe this plateau may be attributed to two factors: 1) the sequence length constraints during pretraining, where the maximum input length to the encoder is set to 384 tokens, and the average input sequence length (excluding passages) is around 130 tokens; 2) the model's ability to learn adequately with 5 or 8 examples, making additional examples less beneficial.

Effect of Number of Retrieved Passages. Figure 3 illustrates the impact of the number of retrieved passages on model performance. We observe that for both 0-shot and 5-shot

settings, the performance of the models increases significantly with the number of retrieved passages. This highlights the effectiveness of the Fusion-in-Decoder architecture (Izacard & Grave, 2021) for knowledge-intensive tasks like open-domain question answering, and underscores the importance of pretraining language models with retrieval augmentation. Additionally, the 5-shot performance consistently outperforms the 0-shot setting. This observation further emphasizes the value of providing in-context examples to improve the performance of retrieval-augmented encoder-decoder language models.

3.2 RAVEN: Combining Retrieval-Augmented Masked and Prefix Language Modeling

In §3.1, we observe that retrieval-augmented encoder-decoder LMs exhibit a certain ability for in-context learning, which has been overlooked in previous studies. However, there are also some limitations such as unstable performance in low-shot settings, and the fact that providing more in-context examples does not consistently improve performance.

To learn a better retriever and enhance the bidirectional understanding ability of the reader, as demonstrated in Izacard et al. (2023), a practical choice is to pretrain the model with the masked language modeling objective, where the input is a corrupted text with several masked spans placed randomly within the sequence (refer to Figure 1 (left) for an example). However, in testing, based on our analysis in §3.1, it is most effective to place the target question after all the in-context examples, with a masked token (i.e., `<extra_id_0>`) following the question (Figure 1, middle)). Thus, there exists a mismatch between pretraining and inference.

Retrieval-Augmented Prefix Language Modeling

Input to Encoder:
Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, agriculture, and computer vision, where it is difficult or<extra_id_0>
Passage: … machine learning models require a high quantity of reliable data in order for the models …

Input to Decoder:
None

Output:
<extra_id_0> unfeasible to develop conventional algorithms to perform the needed tasks.

Figure 4: Retrieval-augmented prefix language modeling.

To solve this issue, we propose combining retrieval-augmented masked and prefix language modeling. Specifically, in the first stage, following Izacard et al. (2023), the retriever and reader are trained jointly with retrieval-augmented masked language modeling. The training objective for the retriever is to minimize the KL divergence $\mathrm{KL}(p_{\mathrm{READER}} \parallel p_{\mathrm{RETRIEVER}})$ between the passage posterior distribution according to the reader and the passage distribution from the retriever over the top-K retrieved passages, i.e., $p_{\mathrm{READER}}(d) = \frac{\exp(\log p_{LM}(a \mid d,q))}{\sum_{i=1}^{K} \exp(\log p_{LM}(a \mid d_i,q))}$, $p_{\mathrm{RETRIEVER}}(d) = \frac{\exp(s(d,q)/T)}{\sum_{i=1}^{K} \exp(s(d_i,q)/T)}$, where $s(\cdot)$ calculates the dot product between the query q and passage d vectors, and T is a hyperparameter. The training objective for the reader is to maximize the likelihood of the masked spans with n retrieved passages: $\sum_i \log p(a_i \mid q, \{d_k\}_{1,\ldots,n}, a_{1:i-1})$.

In the second stage, for each sequence, we mask 10% of the tokens on average at the end of the sequence with the `<extra_id_0>` token. Then, we use the retriever obtained from the first stage to retrieve relevant passages using the prefix and train the reader to recover the suffix of this sequence with the prefix and the passages as input. An example of input and output for retrieval-augmented prefix language modeling is shown in Figure 4. We can observe that the pretraining objective aligns more closely with prompting strategy 1 in Figure 1. We refer to the model trained with this combined objective as RAVEN.

RAVEN benefits from both the retrieval-augmented masked language modeling, which contributes to a better reader and retriever, and retrieval-augmented prefix language modeling, which mitigates the gap between pretraining and inference. This design is *non-trivial*. In Appendix C.1, we verify the effectiveness of it by exploring different training strategies.

3.3 Fusion-in-Context Learning

In §3.1, we observe that the performance does not further improve with more in-context examples after 8-shot. One major reason for this is the limited sequence length during the

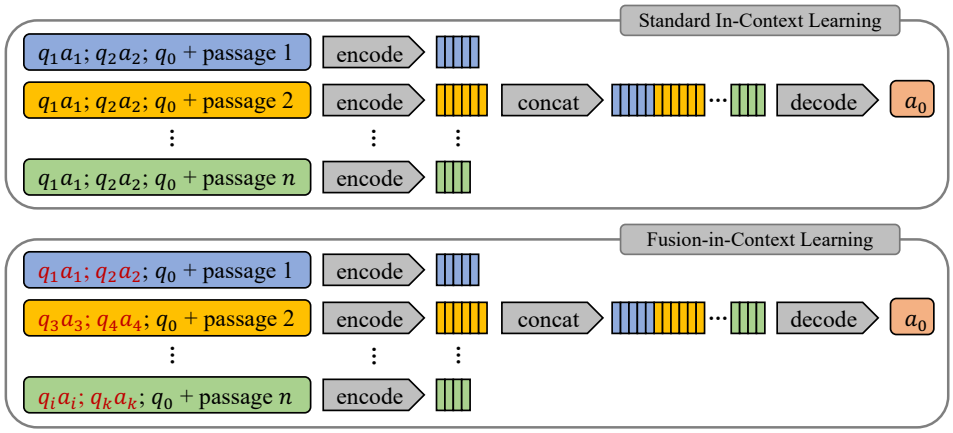

Figure 5: Standard In-Context Learning vs Fusion-in-Context Learning.

pretraining process, which makes it difficult for the model to handle long sequences during inference. Pretraining models with longer contexts would be a potential solution, but it would significantly increase computation cost. Additionally, the maximum input length is also constrained by the maximum sequence length of the retriever, i.e., Contriever, which is based on BERT (Devlin et al., 2019) and has a maximum length of 512 tokens.

As an alternative, we propose an approach to enable models to learn from more in-context examples without requiring additional training. As described in §3.1, the reader is based on the Fusion-in-Decoder architecture (Izacard & Grave, 2021), where multiple passages are retrieved, and each passage, concatenated with the in-context examples and target question, is fed to the encoder separately (Figure 5, top). To allow the model to process more in-context examples, we can feed *different* in-context examples to the encoder with each passage (Figure 5, bottom). In this way, the model can incorporate more in-context examples during its inference process. We refer to this strategy as *Fusion-in-Context Learning (FiCL)*.

In implementation, for a k-shot setting, such as a 64-shot setting, to effectively utilize the 64 examples, we randomly shuffle these examples and select m (e.g., 5) examples in order as the input for the encoder each time. If all the examples have been used, we shuffle the 64 examples again. We denote the configuration of FiCL as [k-m], which stands for [k-shot, m-fusion].

3.4 In-Context Example Retrieval

Liu et al. (2022); Rubin et al. (2022); Su et al. (2023) demonstrate that a well-chosen selection of in-context examples can enhance in-context learning. Building on this insight, we propose utilizing the retriever of RAVEN to retrieve in-context examples. Specifically, we use RAVEN's retriever to build an index during the preparation step, and then, during testing, when the model receives an input, it could efficiently retrieve in-context examples with its retriever.

By integrating RAVEN's retriever in this manner, we aim to: 1) automate in-context learning, which is particularly practical for model owners who have a database of examples. Without this, users would need to manually provide in-context examples; and 2) optimize the selection of in-context examples, thereby improving in-context learning performance.

4 Experiments

4.1 Experimental Setup

Datasets. Following the setup in §3.1, we first evaluate on two widely-used open-domain question answering datasets: Natural Questions (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017). Additionally, we conduct a case study on long-form question answering using the ELI5 dataset (Fan et al., 2019). Furthermore, we assess the models' language understanding ability using the Massively Multitask Language Understanding

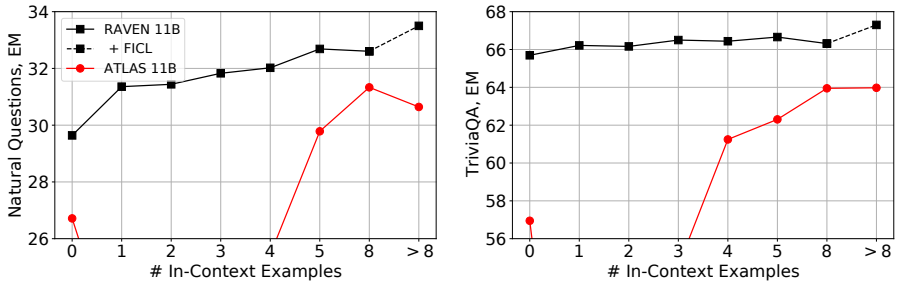

Figure 6: RAVEN vs ATLAS. We report the best observed performance achieved with more than eight shots for "> 8".

(MMLU) benchmark (Hendrycks et al., 2021). Detailed information regarding the MMLU evaluation is in Appendix B.5. Other evaluation settings are the same as §B.1.

Baselines. Since both RAVEN and ATLAS (Izacard et al., 2023) are trained starting from T5, we choose ATLAS as a primary baseline for comparison. We also compare our model with decoder-only LLMs such as GPT-3 (Brown et al., 2020), PaLM (Chowdhery et al., 2023), and LLaMA (Touvron et al., 2023) (in a closed-book setting). Additionally, for open-domain QA, we evaluate our approach against REPLUG (Shi et al., 2023) and RETRO (Borgeaud et al., 2022), as well as its improved version RETRO++ (Wang et al., 2023). These models are decoder-only language models augmented with retrieval. REPLUG is based on Codex (Chen et al., 2021) and Contriever (Izacard et al., 2022), where the passages are retrieved by Contriever (using ensemble and additional adaptation) and fed directly to Codex. RETRO is a GPT model (Radford et al., 2019) augmented with a transformer encoder to encode the retrieved passages. RETRO++ is a variant of RETRO that feeds the most relevant retrieved passage into the GPT decoder while providing other passages to its encoder. For MMLU, we also include FLAN-T5 (Chung et al., 2022), an enhanced version of T5 that has been trained on a large mixture of tasks with instruction finetuning.[4]

4.2 Open-Domain Question Answering

We choose open-domain QA as our primary evaluation task, as it effectively represents knowledge-intensive challenges and is widely employed in real-world applications.

RAVEN vs ATLAS. Figure 6 and Table 3 present the exact match (EM) scores for ATLAS and RAVEN on the NQ and TriviaQA datasets. Both the 3B and 11B RAVEN models significantly outperform ATLAS. For instance, on TriviaQA, RAVEN 11B achieves an improvement of 8.8%, 30.7%, and 2.8% in the 0-shot, 1-shot, and few-shot settings respectively, compared to ATLAS 11B. Furthermore, the performance of RAVEN increases steadily with the number of in-context examples, while the performance of ATLAS experiences a substantial decline in low-shot settings, demonstrating the effectiveness of RAVEN across various shot settings.

Fusion-in-Context Learning. We also report the results of models with Fusion-in-Context Learning (FiCL) in Table 3. For both ATLAS and RAVEN, FiCL contributes to approximately a 1% improvement, which is not attainable by standard in-context learning, where performance does not further improve (or even decreases) with more than 8 in-context examples. This demonstrates the superiority of FiCL for enabling models to learn from more examples.

Comparison to Other Models. In Table 3, we further compare RAVEN to other baselines. On NQ, RAVEN's zero-shot and one-shot performance surpasses all the baselines, including PaLM, even though RAVEN 3B has 180 times fewer parameters than PaLM 540B. The zero-shot performance of RAVEN on TriviaQA is also on par with PaLM 62B. Furthermore, RAVEN's zero-shot performance significantly exceeds that of both RETRO and RETRO++, which are retrieval-augmented language models of a similar scale.

In the few-shot setting, with FiCL, RAVEN achieves performance comparable to GPT-3 175B and PaLM 62B. However, there remains a gap between RAVEN and the larger PaLM 540B

[4]Implementation details are described in Appendix B.2.

		Natural Questions			TriviaQA		
		0-shot	1-shot	few-shot	0-shot	1-shot	few-shot
GPT-3	13B	7.8	13.7	21.0 (64)	41.8	51.3	57.5 (64)
GPT-3	175B	14.6	23.0	29.9 (64)	64.3	68.0	71.2 (64)
PaLM	8B	8.4	10.6	14.6 (5)	39.5	48.5	47.2 (5)
PaLM	62B	18.1	23.1	27.6 (5)	67.3	72.7	70.1 (5)
PaLM	540B	21.2	29.3	39.6 (64)	**76.9**	**81.4**	**81.4** (1)*
Codex	175B	-	-	40.6 (16)	-	-	73.6 (16)
LLaMA	7B	16.8	18.7	26.1 (64)	50.0	53.4	57.6 (64)
LLaMA	65B	23.8	31.0	39.9 (64)	68.2	71.6	73.0 (64)
Retrieval-Augmented Language Models							
Codex + Contriever	175B	-	-	44.2 (16)	-	-	76.0 (16)
Codex + REPLUG	175B	-	-	44.7 (16)	-	-	76.8 (16)
Codex + REPLUG LSR	175B	-	-	**45.5** (16)	-	-	77.3 (16)
RETRO	9.5B	8.9	-	-	36.0	-	-
RETRO++	9.5B	25.8	-	-	48.3	-	-
ATLAS	3B	23.7	25.1	28.4 (5)	54.3	55.5	61.1 (5)
ATLAS + FiCL	3B			29.6 [64-5]			62.0 [64-5]
ATLAS	11B	26.7	21.3	31.3 (8)	56.9	35.5	63.9 (8)
ATLAS + FiCL	11B			32.0 [64-8]			64.9 [64-8]
RAVEN	3B	29.3	**31.7**	31.4 (5)	62.4	63.2	62.6 (5)
RAVEN + FiCL	3B			32.8 [40-1]			63.6 [40-1]
RAVEN	11B	**29.6**	31.4	32.7 (5)	65.7	66.2	66.7 (5)
RAVEN + FiCL	11B			33.5 [64-5]			67.3 [64-5]

* For TriviaQA, PaLM's 1-shot performance surpasses other settings. We follow the original paper to report the 1-shot result. For other models, we select the best k-shot ($k \in \{2, 3, 4, 5, 8, 16\}$) performance or report the number in the original paper.

Table 3: Results on NQ and TriviaQA. Since the performance varies significantly depending on the capability of the base model, the results from models other than ATLAS should only be used for reference to gauge the position. And we assume RAVEN can achieve significant performance improvement when based on a stronger base model.

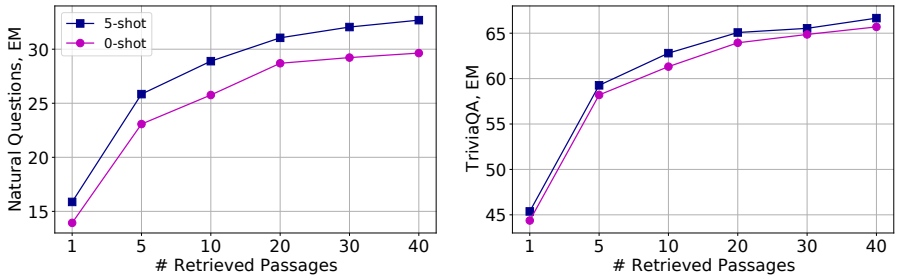

Figure 7: Results of RAVEN 11B with different numbers of retrieved passages.

and Codex 175B models. Nevertheless, given the considerably smaller scale of RAVEN in comparison to PaLM and Codex, its performance can be considered impressive. The performance of RAVEN may be further improved if it is built upon a larger model, in which case its few-shot performance is likely to surpass that of PaLM and Codex.

Effect of Number of Retrieved Passages. Figure 7 illustrates the effect of the number of retrieved passages. As the number of retrieved passages increases, we observe a significant performance improvement of RAVEN 11B in both the 0-shot and 5-shot settings.

In-Context Example Retrieval. §3.4 suggests using RAVEN's retriever for in-context example retrieval. Results in Table 4 show that this approach improves RAVEN's few-shot results, especially on NQ where a ∼10% improvement is observed. This indicates the positive impact of incorporating more relevant in-context examples.

Additional Results. We conduct an **ablation study of different training strategies** in Appendix C.1 and provide a **case study on long-form question answering** in Appendix C.2.

	NQ		TQA	
	1-shot	5-shot	1-shot	5-shot
3B	+9.1	+11.6	+0.0	+1.6
11B	+9.8	+11.1	-0.5	+1.0

Table 4: Performance improvement of RAVEN with In-Context Example Retrieval.

4.3 MMLU

Table 5 summarizes the results (accuracy) on Massive Multitask Language Understanding (MMLU). We find that the zero-shot performance of RAVEN is impressive, surpassing the few-shot performance of GPT-3 175B and being slightly worse than PaLM 62B, despite having a significantly smaller number of parameters. Furthermore, with the same number of parameters, the performance of RAVEN is far superior to T5. Additionally, even without instruction finetuning, RAVEN achieves performance comparable to FLAN-T5, a model finetuned on a large collection of tasks. We expect further improvement of RAVEN by applying instruction tuning as well and leave it for future study.

		0-shot	1-shot	5-shot
GPT-3	13B	-	-	26.0
GPT-3	175B	-	-	43.9
PaLM	8B	-	-	25.3
PaLM	62B	-	-	53.7
PaLM	540B	-	-	69.3
T5	3B	-	-	25.7
T5	11B	-	-	25.9
FLAN-T5	3B	-	-	52.4
FLAN-T5	11B	-	-	55.1
ATLAS	3B	43.7	36.9	38.5
+ FiCL	3B			42.6 [40-1]
ATLAS	11B	47.4	45.3	44.2
+ FiCL	11B			48.0 [40-1]
RAVEN	3B	45.7	40.0	40.4
+ FiCL	3B			44.5 [64-5]
RAVEN	11B	48.9	49.2	48.7
+ FiCL	11B			50.5 [40-1]

Table 5: Results on MMLU.

Interestingly, with standard in-context learning, the few-shot performance of RAVEN is worse than zero-shot, possibly due to the longer questions and answer options in MMLU causing context length issues in the 5-shot setting. Also, in the one-shot setting, since MMLU is a multiple-choice QA task, providing only one example might introduce bias in the model's prediction, favoring a specific option. However, with Fusion-in-Context Learning, the performance improves significantly, leading to better few-shot performance for the 11B model compared to its zero-shot performance, further demonstrating the effectiveness of FiCL.

5 Conclusion

In this study, we have delved into the in-context learning ability of retrieval-augmented encoder-decoder language models. We commenced with a comprehensive analysis of the models in the literature and subsequently developed our model based on the analysis. Our extensive experimental results demonstrated that our model significantly outperforms previous models and achieves results on par with some of the most advanced language models, even with substantially fewer parameters. These findings highlight the potential of retrieval-augmented encoder-decoder language models in the realm of in-context learning.

References

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. Improving language models by retrieving from trillions of tokens. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume

162 of *Proceedings of Machine Learning Research*, pp. 2206–2240. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/borgeaud22a.html.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

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

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1870–1879, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1171. URL https://aclanthology.org/P17-1171.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023. URL http://jmlr.org/papers/v24/22-1144.html.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*, 2022. URL https://arxiv.org/abs/2301.00234.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3558–3567, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1346. URL https://aclanthology.org/P19-1346.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Jie Huang and Kevin Chen-Chuan Chang. Towards reasoning in large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1049–1065, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.67. URL https://aclanthology.org/2023.findings-acl.67.

Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. Are large pre-trained language models leaking your personal information? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 2038–2047, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.findings-emnlp.148.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 874–880, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.74. URL https://aclanthology.org/2021.eacl-main.74.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=jKN1pXi7b0.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251):1–43, 2023. URL http://jmlr.org/papers/v24/23-0037.html.

Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 6237–6250, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.418.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HklBjCEKvH.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive nlp. *arXiv preprint arXiv:2212.14024*, 2022.

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. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026.

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. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pp. 100–114, Dublin, Ireland and Online, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.deelio-1.10. URL https://aclanthology.org/2022.deelio-1.10.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*, 2023. URL https://arxiv.org/abs/2301.13688.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A. Smith. Time waits for no one! analysis and challenges of temporal misalignment. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5944–5958, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.435. URL https://aclanthology.org/2022.naacl-main.435.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*, 2022. URL https://arxiv.org/abs/2212.10511.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3470–3487, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.244. URL https://aclanthology.org/2022.acl-long.244.

OpenAI. Chatgpt: Optimizing language models for dialogue. *OpenAI*, 2022.

OpenAI. Gpt-4 technical report, 2023.

Ajay Patel, Bryan Li, Mohammad Sadegh Rasooli, Noah Constant, Colin Raffel, and Chris Callison-Burch. Bidirectional language models are also few-shot learners. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=wCFB37bzud4.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2523–2544, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.200. URL https://aclanthology.org/2021.naacl-main.200.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. Is chatgpt a general-purpose natural language processing task solver? *arXiv preprint arXiv:2302.06476*, 2023. URL https://arxiv.org/abs/2302.06476.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), jan 2020. ISSN 1532-4435. URL https://dl.acm.org/doi/abs/10.5555/3455716.3455856.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2655–2671, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.191. URL https://aclanthology.org/2022.naacl-main.191.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=9Vrb9D0WI4.

Weijia Shi, Julian Michael, Suchin Gururangan, and Luke Zettlemoyer. Nearest neighbor zero-shot inference. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 3254–3265, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.214.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*, 2023. URL https://arxiv.org/abs/2301.12652.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, et al. Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model. *arXiv preprint arXiv:2201.11990*, 2022. URL https://arxiv.org/abs/2201.11990.

Hongjin Su, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. Selective annotation makes language models better few-shot learners. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=qY1hlv7gwg.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. UL2: Unifying language learning paradigms. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6ruVLB727MC.

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

Boxin Wang, Wei Ping, Peng Xu, Lawrence McAfee, Zihan Liu, Mohammad Shoeybi, Yi Dong, Oleksii Kuchaiev, Bo Li, Chaowei Xiao, et al. Shall we pretrain autoregressive language models with retrieval? a comprehensive study. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7763–7786, 2023. URL https://aclanthology.org/2023.emnlp-main.482.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=gEZrGCozdqR.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 483–498, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.41. URL https://aclanthology.org/2021.naacl-main.41.

Qinyuan Ye, Iz Beltagy, Matthew E Peters, Xiang Ren, and Hannaneh Hajishirzi. Fid-icl: A fusion-in-decoder approach for efficient in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8158–8185, 2023. URL https://aclanthology.org/2023.acl-long.454.

Shen Zheng, Jie Huang, and Kevin Chen-Chuan Chang. Why does ChatGPT fall short in providing truthful answers? In *I Can't Believe It's Not Better Workshop: Failure Modes in the Age of Foundation Models*, 2023. URL https://openreview.net/forum?id=w7o14LCw9P.

A Limitations and Broader Impact

While the performance of RAVEN is impressive considering its scale and training budget, there are also some limitations. One limitation arises from the constrained context length inherent to the base models (i.e., T5 or ATLAS) we employed. This restriction poses challenges to the scalability of in-context learning, especially as the number of in-context examples increases. While our Fusion-in-Context Learning (FiCL) strategy does offer a mitigative approach to this constraint, an alternative and possibly more optimal solution might involve extending the context length. This would be particularly beneficial for tasks requiring extensive inputs.

Furthermore, when compared to some of the prevailing decoder-only language models, particularly those exceeding 100B parameters, the models deployed in our research might appear relatively diminutive in scale (in terms of both the number of parameters and the amount of training data). Our endeavor partially seeks to catalyze further investigations into more powerful encoder-decoder models.

Nonetheless, the insights and methods proposed are transferable and have the potential to enhance other models, including those that are domain-specialized or more powerful, such as mT5 (Xue et al., 2021) and UL2 (Tay et al., 2023). Future work focusing on scaling up the model, applying these methods, and further studying its in-context learning ability is encouraged. Drawing on the benefits of scaling up and combining this with our proposed approaches, we believe that there is potential to develop even more powerful retrieval-augmented language models in the future. Another promising future direction is exploring how to combine the Fusion-in-Decoder architecture with existing decoder-only language models. By doing so, we can harness the advantages of both architectures—employing a bidirectional architecture to effectively encode retrieved passages for the most powerful decoder-only LLMs.

B Additional Experimental Details

B.1 Experimental Setup for §3.1

We select two widely-used datasets in the domain of open-domain question answering: Natural Questions (NQ) (Kwiatkowski et al., 2019) and TriviaQA (TQA) (Joshi et al., 2017). To assess the performance, we follow the previous work (Izacard et al., 2023) to employ the standard exact match (EM) metric. For the few-shot settings, we follow Brown et al. (2020) to evaluate each example in the test set by generating in-context examples through randomly sampling k instances from the respective task's training set. Following Izacard et al. (2023), we use an index composed of December 2018 Wikipedia dump for NQ and an index composed of December 2021 Wikipedia corpora for TriviaQA. We retrieve 40 documents by default. We test the checkpoints released in the official repository of Izacard et al. (2023)[5], covering sizes of 11B (XXL), 3B (XL), and 770M (Large).

B.2 Training Details

We train two versions of RAVEN: 3B and 11B. To isolate the effect of training variance with masked language modeling, we initialize both the retriever and the reader of the models with the weights of ATLAS (3B and 11B) and continue to pretrain the model with prefix language modeling. To isolate the effect of retrieval, we do not update the retriever during the training process for prefix language modeling. We pretrain the reader using the December 2021 Wikipedia corpora preprocessed by Izacard et al. (2023), where the index is also constructed using the same corpora. In accordance with Izacard et al. (2023), we retrieve 20 passages for each masked sequence (excluding passages identical to the original sequence). Both the 3B and 11B models are trained for 5,000 steps, using AdamW optimizer (Loshchilov & Hutter, 2019) with a batch size of 64. We employ a learning rate of 4×10^{-5} for the 3B model and 1×10^{-5} for the 11B model, with linear decay and 100

[5] https://github.com/facebookresearch/atlas

warmup steps. All the models are trained on NVIDIA A100 GPUs (80 GB). For the 3B model, we utilize 8 GPUs, whereas for the 11B model, we employ 32 GPUs. The prompt used for prefix language modeling is detailed in Appendix B.3. During testing, we default to retrieving 40 documents for all tasks. The prompts used can be found in Appendix B.4 and Appendix B.5.

B.3 Retrieval-Augmented Prefix Language Modeling

In alignment with the pretraining of ATLAS, we design the prompt for prefix language modeling as

```
{prefix}<extra_id_0> title: {title} context: {text}
```

where {prefix} represents the prefix of an input sequence. The {title} and {text} elements are retrieved by the model's retriever using the prefix as a query. Here, {text} signifies the retrieved passage, while {title} denotes the corresponding article and section title of the passage. The model is trained to generate

```
<extra_id_0>{suffix}
```

where {suffix} is the suffix (masked by `<extra_id_0>`) of the input sequence.

B.4 Open-Domain Question Answering

In accordance with pretraining, we use the following prompt for open-domain question answering:

```
Question: {question} Answer:<extra_id_0> title: {title} context: {text}
```

For example,

```
Question: In which country was the first permanent bungee jumping site situated?
Answer:<extra_id_0> title: Bungee jumping: Modern sport context: first permanent
commercial bungee site, the Kawarau Bridge Bungy at the Kawarau Gorge Suspension
Bridge near Queenstown in the South Island of New Zealand. Hackett remains one
of the largest commercial operators, with concerns in several countries. Several
million successful jumps have taken place since 1980.  This safety record is
attributable to bungee operators rigorously conforming to standards and guidelines
governing jumps, such as double checking calculations and fittings for every jump.
As with any sport, injuries can still occur (see below), and there have been
fatalities. A relatively common mistake in fatality cases is to use a cord that
```

B.5 MMLU

MMLU comprises 57 multiple-choice question answering datasets that span various domains, including elementary mathematics, US history, computer science, and more. For the evaluation on MMLU, we report the accuracy and use an index composed of December 2021 Wikipedia corpora. We follow Izacard et al. (2023) to apply the "de-biased" inference. Specifically, during inference, we execute four forward passes, each corresponding to a cyclic permutation of the answer letter-option assignment within the question. For instance, the answer option designated to letter 'A' is shifted to 'B', 'B' to 'C', 'C' to 'D', and 'D' to 'A'. The final prediction is obtained by summing up the probabilities from these four forward passes.

We design the prompt in the following format:

```
Question: {question} Options: {candidate answers} Answer:<extra_id_0> title:
{title} context: {text}
```

For example,

		Natural Questions			TriviaQA		
		0-shot	1-shot	5-shot	0-shot	1-shot	5-shot
ATLAS	3B (Mask)	23.7	25.1	28.4	54.3	55.5	61.1
ATLAS	3B (Mask, 5k more steps)	22.9	22.5	28.1	50.8	50.1	61.1
RAVEN$^-$	3B (Prefix)	24.8	29.1	30.1	55.4	61.4	62.3
RAVEN$^-$	3B (Mix)	25.1	28.4	30.9	56.1	61.4	62.2
RAVEN	3B	29.3	31.7	31.4	62.4	63.2	62.6

Table 6: Results of ATLAS and RAVEN trained with different strategies.

```
Question: Over time, non-volcanic mountains can form due to the interaction
of plate boundaries. Which interaction is most likely associated with the
formation of non-volcanic mountains? Options: (A) continental plates colliding
with continental plates (B) continental plates separating from continental plates
(C) oceanic plates colliding with oceanic plates (D) oceanic plates separating
from oceanic plates Answer:<extra_id_0> title: ... context: ...
```

Given that many questions in the MMLU benchmark are quite lengthy, concatenating in-context examples (questions and candidate answers) with the target question in a few-shot setting is likely to exceed the maximum input length. To mitigate this, we only sample examples with question lengths of fewer than 50 tokens to use as in-context examples.

C Additional Results

C.1 Ablation Study

We conduct an ablation study by training ATLAS and RAVEN with different pretraining strategies. First, to isolate the effect of more training steps of RAVEN, we also train ATLAS for 5,000 more steps using the masked language modeling objective. Results in Table 6 (row 2) show that the performance does not improve, indicating that the performance improvement of RAVEN compared to ATLAS is not simply due to training for more steps.

Second, to verify the effectiveness of RAVEN's training strategy (i.e., first masked language modeling, and then prefix language modeling), we train two variants of RAVEN, starting from the T5-lm-adapt checkpoint[6], which is the checkpoint that ATLAS starts from. For the first variant, we use the same prefix language modeling objective of RAVEN. For the second variant, we train the model with a mixture of masked and prefix language modeling. Specifically, we construct corrupted texts by both masking 15% spans in the sequence (same as ATLAS) and replacing the suffix with a special mask token <extra_id_99> (used in testing). We train the model for 10,000 steps and update the retriever and refresh the index during training with the optimal strategy described in Izacard et al. (2023). Table 6 (RAVEN$^-$ in row 3 and 4) summarizes the results. We find that the performance of these two variants is superior to ATLAS, but inferior to RAVEN when trained using the strategy described in §3.2. An explanation for this is that, by training with masked language modeling first, the model can achieve better language understanding ability and is equipped with a more effective retriever (as emperically verified in Izacard et al. (2023)). Subsequently, by training with prefix language modeling, the mismatch between pretraining and inference is mitigated, resulting in improved zero-shot and few-shot performance.

C.2 Long-Form Question Answering

Table 7 presents some example outputs of ATLAS and RAVEN 11B on long-form question answering. The questions are sampled from the ELI5 dataset (Fan et al., 2019). An examination of these results reveals that ATLAS typically generates concise answers, while the output from RAVEN generally encompasses more information. This is a predictable outcome given

[6] https://huggingface.co/google/t5-xl-lm-adapt

that ATLAS is pretrained solely with masked language modeling, where each masked span usually contains only a handful of tokens. Besides, while RAVEN's answers are not always entirely accurate, they generally exhibit higher quality compared to ATLAS. Furthermore, the use of Fusion-in-Context Learning in RAVEN appears to contribute to a more coherent and informative generation.

Question		**How does a painting like "When Will You Marry" sell for $300 million dollars when it doesn't look like anything spectacular**
Human		When it comes to a lot of art, context is more important than the actual material. Rarity, historical context etc. make a thing more valuable than its actual aesthetic properties. For example, some shitty buggy SNES or N64 games will sell for huge amounts if they are well preserved and in the original packaging etc, even when any high school student nowadays could write a better game in a week.
ATLAS	1-shot	It's a masterpiece.
RAVEN	1-shot	It's not the painting that sells for $300 million dollars, it's the story behind the painting.
RAVEN	FiCL [40-1]	It's not the painting that sells for $300 million dollars, it's the reputation of the artist.
Question		**When the society goes fully automatized, what will be the "normal" people needed for?**
Human		Call centers... I am being totally serious. If your Sony TV is being completely fabricated by machines, shipped by machines, sold on amazon and delivered by machines, one of the few value adds companies will be able to deliver is GOOD customer service. So calling that 1-800 number and being instantly connected to a knowledgeable, friendly, human being who is going to patiently help you fix whatever issue you have (FYI... you forgot to plug it in).
ATLAS	1-shot	'> ELI5: What will the society go fully automatized? >
RAVEN	1-shot	The ones who will program the robots.
RAVEN	FiCL [40-1]	The "normal" people will be needed to program the robots.
Question		**Why is a 2 liter of Coca-cola $1, a 12 oz bottle $1.49 and the same size bottle of water $1.99?**
Human		Because people will pay. Plain and simple. Capitalism at its best.
ATLAS	1-shot	The price of the product is based on the cost of the packaging.
RAVEN	1-shot	The cost of packaging is proportional to the size of the container.
RAVEN	FiCL [40-1]	The cost of packaging is the largest portion of the cost of a product. The larger the container, the less packaging is required.
Question		**If Matter Cannot Be Created or Destroyed... How do trees form?**
Human		Say you have a box of Legos with 500 pieces. With those 500 pieces, you build a bulldozer. Then one day, you decide to rearrange those pieces from a bulldozer to a monster truck. It's still the same 500 pieces, just in a different layout. This is how trees are formed. The matter is just rearranged from other objects. Soil, nutrients, water, and even air. This is called "Conservation of Mass." EDIT: added air. Thanks. :)
ATLAS	1-shot	Trees are made of carbon and oxygen.
RAVEN	1-shot	Matter can be created or destroyed by combining two or more different types of matter.
RAVEN	FiCL [40-1]	The tree is a living organism. It is made up of matter. The tree is made up of carbon, hydrogen, oxygen, and water.
Question		**If the flu shot contains a non-active version of the flu, then why are "flu like symptoms" common for the days following?**
Human		the non-active proteins trigger an immune system response which can involve some inflammatory system responses. once the immune system is activated for that infection your body will react more readily if the actual infection gets into you.
ATLAS	1-shot	The flu shot contains live virus particles.
RAVEN	1-shot	The flu shot contains a live attenuated virus. This means that it is not able to infect the body, but it can still cause some side effects.
RAVEN	FiCL [40-1]	The flu shot contains a weakened version of the virus. This weakened virus is able to trigger an immune response in the body. This immune response can cause flu like symptoms.

Table 7: Example outputs of ATLAS and RAVEN 11B on long-form question answering.