RA-DIT: Retrieval-Augmented Dual Instruction Tuning

Xi Victoria Lin*, Xilun Chen*, Mingda Chen*, Weijia Shi, Maria Lomeli, Rich James, Pedro Rodriguez, Jacob Kahn, Gergely Szilvasy, Mike Lewis, Luke Zettlemoyer, Scott Wen-tau Yih (*Equal Contribution)

Meta

Background

Retrieval-augmentation (RA) has become an increasingly popular approach for enriching LLMs with long-tail knowledge and keeping them up-to-date. Existing efforts of RALM development aim to address two high-level challenges: (i) enhancing the LLM's capability to incorporate retrieved knowledge and (ii) updating the retrieval component to return more relevant content. We present a lightweight fine-tuning approach that can retrofit any pre-trained LLMs with retrieval capabilities in an efficient manner, leading to new possibilities for customizable AI models.

Fine-tuning Datasets

Dialogue		
Name	LM	R
OASST	✓	✓

Open-domain QA		
Name	LM	R
Commonsense QA	✓	✓
MathQA	✓	✓
Web Questions	✓	✓
WikiQA	✓	✓
YahooAnswersQA	✓	✓
Freebase QA		✓
MS_MARCO		✓

Reading Comprehension		
Name	LM	R
CoQA	✓	
DROP	✓	
NarrativeQA	✓	
NewsQA	✓	
PubMed QA	✓	✓
QUAIL	✓	
QUAREL	✓	✓
SQuAD v2	✓	

Knowledge Utilization

Chain-of-thought Reasoning		
Name	LM	R
AQuA	✓	
ECQA	✓	
GSM8k	✓	
Math	✓	
StrategyQA	✓	

Summarization		
Name	LM	R
CNN_Dailymail	✓	

Contextual Awareness

Inference: Parallel In-Context Retrieval-Augmentation

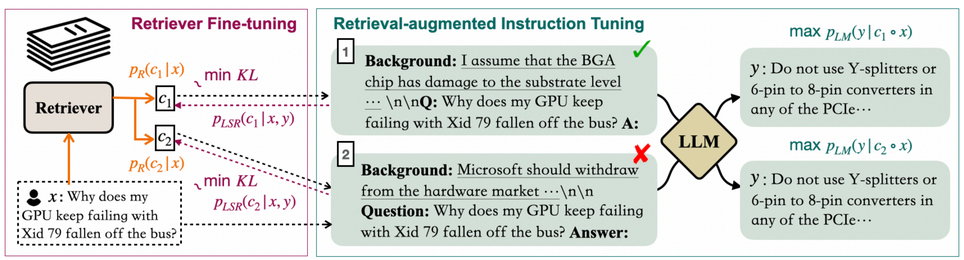
Following REPLUG (Shi et al. 2023), for a given model prompt x, we retrieve the top-k relevant text chunks $C' \subset C, |C'| = k$. Each retrieved chunk $c \in C'$ is independently prepended to the prompt, and the LM predictions from multiple augmented prompts are ensembled to derive the final prediction.

$$p_{LM}(y|x, C') = \sum_{c \in C'} p_{LM}(y|c \circ x) \cdot p_R(c|x)$$

Here $\circ$ denotes sequence concatenation, and $p_R(c|x) = \frac{\exp s(x,c)}{\sum_{c' \in C'} \exp s(x,c')}$ are the retriever scores re-normalized among the top-k relevant text chunks.

Approach

Given a pair of pre-trained LM (🦙Llama, Touvron et al. 2023) and dense retriever (🐉 DRAGON+, Lin et al. 2023), our approach operates in two independent steps: (1) one fine-tunes the LM to better use retrieved information via retrieval-augmented instruction tuning; (2) the other updates the retriever to return more content as preferred by the LM.



Main Figure. For a given example, the LM-ft component updates the LM to maximize the likelihood of the correct answer given the retrieval-augmented instructions; the R-ft component updates the retriever to minimize the KL-Divergence between the retriever score distribution and the LM preference.

Retriever Fine-tuning Loss

$$\mathcal{L}(\mathcal{D}_R) = \mathbb{E}_{(x,y)\in\mathcal{D}_R} KL\big(p_R(c|x) \parallel p_{LSR}(c|x,y)\big)$$

$$p_{LSR}(c|x,y) = \frac{\exp\left(p_{LM}(y|c \circ x)/\tau\right)}{\sum_{c' \in C} \exp\left(p_{LM}(y|c' \circ x)/\tau\right)} \approx \frac{\exp\left(p_{LM}(y|c \circ x)/\tau\right)}{\sum_{c' \in C'} \exp\left(p_{LM}(y|c' \circ x)/\tau\right)},$$

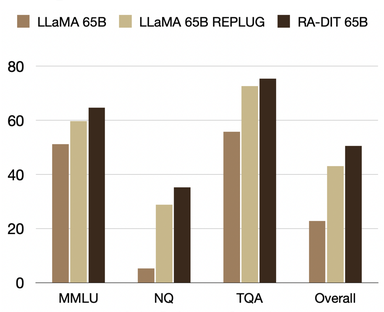
LM Fine-tuning Loss

$$\mathcal{L}(\mathcal{D}_L) = -\sum_i \sum_j \log p_{LM}(y_i|c_{ij} \circ x_i)$$

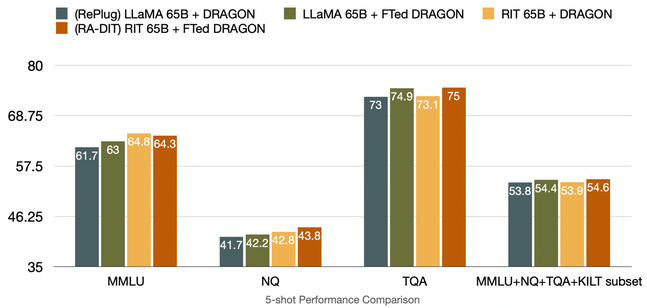
Experiment Setup

- **Retriever:** Our retrieval corpus consists of 37M text chunks from the Dec 20, 2021 Wikipedia dump released by Izavard et al. (2022) and 362M text chunks from the 2017-2020 CommonCrawl dump.
- **LM Fine-tuning:** We use the top-3 retrieved text chunks for a give example to generate the fine-tuning instances. The best generalization performance on the dev set can be achieved using a small number of fine-tuning steps. All of our models were fine-tuned with <500 steps.
- **Retriever Fine-tuning:** We adopt a mixture of 95% unsupervised text and 5% downstream tasks for retriever fine-tuning.

Knowledge-Intensive Task Performance

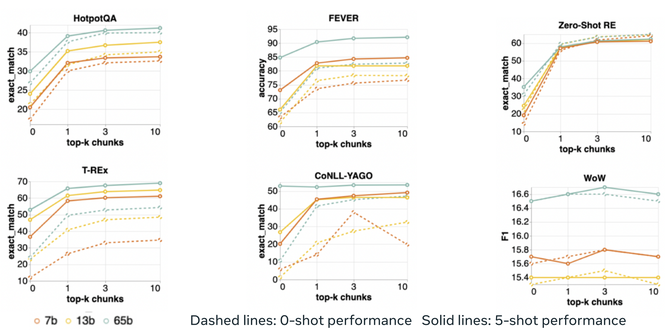


Dual Instruction Tuning Ablation



Scaling Laws

- Smaller LMs in general benefits more from retrieval augmentation.
- For tasks that measures single-hop knowledge lookup (e.g. zero-shot relation extraction) retrieval-augmentation enable smaller LMs (e.g. 7B) to match the performance of larger LMs (e.g. 70B). For tasks that requires more sophisticated reasoning (e.g. HotpotQA), smaller LMs still demonstrate a performance gap comparing to larger models after retrieval augmentation.



Further Questions?

Reach out to {victorialin,xilun,mingdachen,scottyih}@meta.com