中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES

Augmentation-Adapted Retriever Improves Generalization of Language Models as Generic Plug-In

Zichun Yu[1] Chenyan Xiong[2] Shi Yu[1] Zhiyuan Liu[13]
[1]Dept. of Comp. Sci. & Tech., Institute for AI, Tsinghua University, Beijing, China
[2]Microsoft Research, Redmond, USA
[3]Beijing National Research Center for Information Science and Technology, Beijing, China
{yuzc19, yus21}@mails.tsinghua.edu.cn; chenyan.xiong@microsoft.com
liuzy@tsinghua.edu.cn

- **Motivation:**

Existing retrieval augmented methods jointly fine-tune the retriever and the LM, which can be expensive when more and more unique demands emerge. More importantly, some LMs can only be accessed through black-box APIs and does not support fine-tuning. This paper aims to retrieve useful documents for unseen LMs.

- **Methods:**

This paper introduced Augmentation- Adapted Retriever (AAR) to assist black-box LMs with downstream tasks as generic plug-in.

  - Leverage a small *source LM* to pro- vide LM-preferred signals for retriever's training.
  - The retriever after training (i.e., AAR) can be directly utilized to assist a large *target LM* by plugging in the retrieved documents.

- **Experiemnts:**
  - Evaluate AAR on a multi-task language understanding dataset MMLU and an entity-centric question answering dataset PopQA

Introduction

- **Experiemnts:**
  - Evaluate AAR on a multi-task language understanding dataset MMLU and an entity-centric question answering dataset PopQA
  - Evaluate ARR with different backbones
  - Analysis reveals that the preferences obtained from different-sized source LMs are similar, and LMs with near capacities tend to yield closer preferred document sets.
  - As a result, AAR model trained from a small source LM can be considered as a generic plug-in to enhance the zero-shot generalization of a significantly larger target LM.

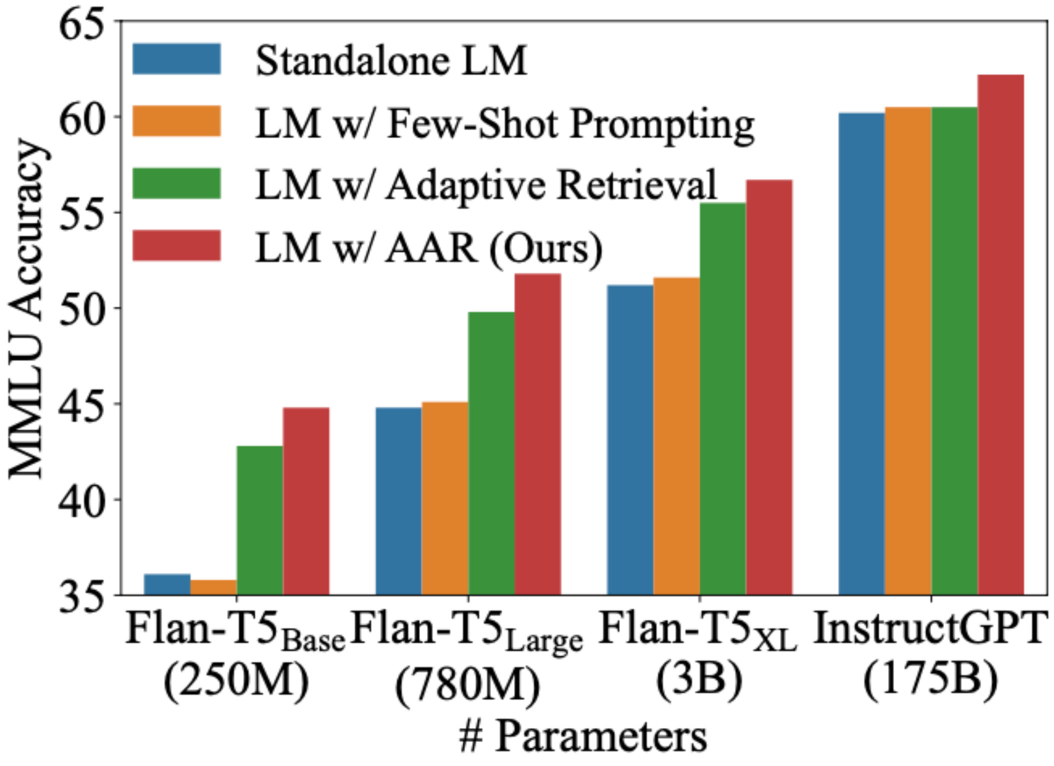


Figure 1: Performance of LM w/ AAR (Ours).

Assisted with a generic AAR, LMs of different sizes and architectures can consistently outperform the standalone LMs;
The performance of smaller LMs can sometimes surpass the standalone counterparts of significantly larger sizes;
AAR demonstrates advantages over other augmentation approaches such as few-shot prompting and adaptive retrieval.

Preliminary

A dense retrieval model first represents q and the document d into an embedding space using a pre-trained encoder g,

$$\boldsymbol{q} = g(q); \boldsymbol{d} = g(d), d \in C, \quad (1)$$

and match their embeddings by dot product function f, which supports fast approximate nearest neighbor search (ANN) (André et al., 2016; Johnson et al., 2021). We then define D^a that contains top-N retrieved documents as:

$$D^a = \{d_1^a \dots d_N^a\} = \mathrm{ANN}_{f(\boldsymbol{q},\circ)}^N. \quad (2)$$

For the LM backbones, the decoder-only and the encoder-decoder models are the two primary choices of the retrieval-augmented LMs (Izacard and Grave, 2021b; Yu et al., 2023).

Given a decoder-only LM like GPT-3 (Brown et al., 2020), the LM input can be a simple concatenation of the query and all the augmentation documents $\{d_1^a \dots d_N^a\}$. Then, the LM will generate the answer based on the inputs auto-regressively.

For an encoder-decoder LM like T5 (Raffel et al., 2020), taking simple concatenation as the encoder input may still be effective. However, this method may not scale to a large volume of documents due to the quadratic self-attention computation associated with the number of documents. To aggregate multiple documents more efficiently, Izacard and Grave (2021b) propose the fusion-in-decoder (FiD) mechanism, which soon becomes the mainstream in the development of encoder-decoder retrieval-augmented LMs. It first encodes each concatenation of the (d_i^a, q) pair separately and then lets the decoder attend to all parts:

$$\mathrm{FiD}(q) = \mathrm{Dec}(\mathrm{Enc}(d_1^a \oplus q) \dots \mathrm{Enc}(d_N^a \oplus q)). \quad (3)$$

Method: Augmentation-adapted Retriever

- Augmentation-Adapted Retriever (AAR) is a generic plug-in for black-box LMs.
- AAR can learn the preferences of LMs without the need for fine-tuning them.
- AAR utilizes an encoder-decoder LM as source LM (Ls) to provide LM-preferred signals on a source task (Ts) for fine-tuning a pre-trained retriever.
- AAR plug the fine-tuned retriever into unseen target LM (Lt) on a set of target tasks (Tt) non-intersecting with Ts .

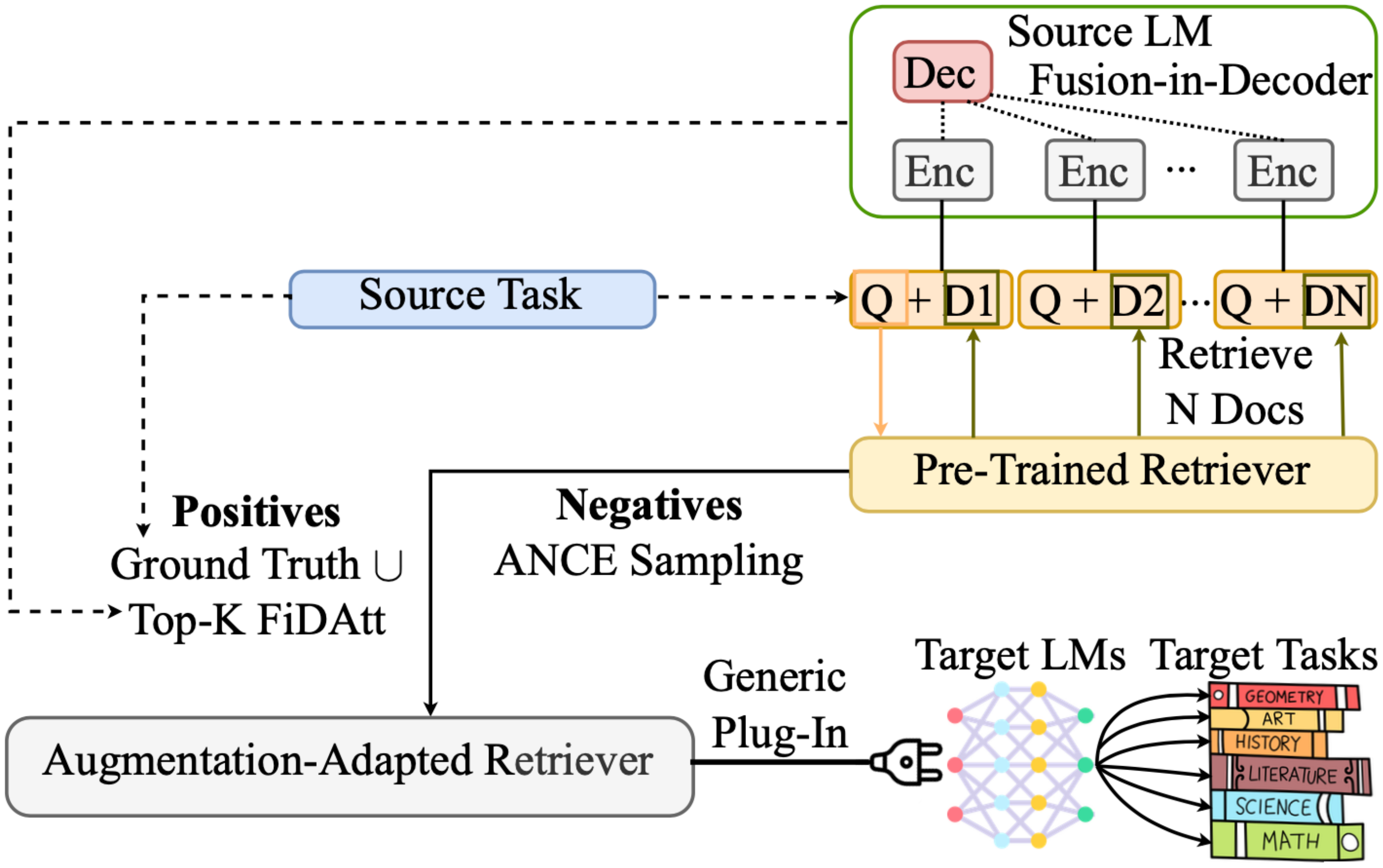


Figure 2: Illustration of augmentation-adapted retriever.

Method: Augmentation-adapted Retriever

Our training method starts from a source task T_s, where we aggregate the source LM L_s's average FiD cross-attention (FiDAtt) scores S_i^a corresponding to document d_i^a from the first decoder token over all the layers, all the heads and all the input tokens t of $d_i^a \oplus q$:

$$S_i^a = \frac{1}{\text{ln} * \text{hn} * \text{tn}} \sum_{\text{layers}} \sum_{\text{heads}} \sum_{t \in d_i^a \oplus q} \text{FiDAtt}(\text{FiD}(q)). \quad (4)$$

where ln, hn, tn are the numbers of the layers, the heads and the input tokens.

To make the training process more robust, we utilize the FiDAtt scores to annotate the LM-preferred positive documents in a discrete way:

$$D^{a+} = D^{h+} \cup \text{Top-}K_{S_i^a, D^a}, \quad (5)$$

Then, we sample hard negatives following ANCE (Xiong et al., 2021) and formulate the training loss $\mathcal{L}$ of the retriever as:

$$D^- = \text{ANN}^M_{f(\boldsymbol{q},\circ)} \backslash D^{a+}, \quad (6)$$

$$\mathcal{L} = \sum_q \sum_{d^+ \in D^{a+}} \sum_{d^- \in D^-} l(f(\boldsymbol{q}, \boldsymbol{d}^+), f(\boldsymbol{q}, \boldsymbol{d}^-)), \quad (7)$$

where M is the hyperparameter of the negative sampling depth and l is the standard cross entropy loss. After fine-tuning the retriever, we directly use it to augment unseen target LM L_t on each task from target task set T_t.

Experiments

Settings	Methods	# Parameters	MMLU All	MMLU Hum.	MMLU Soc. Sci.	MMLU STEM	MMLU Other	PopQA All
Base Setting: T5 Base Size								
Few-shot	Flan-T5$_{\text{Base}}$ (Chung et al., 2022)	250M	35.8	39.6	39.8	26.3	41.2	8.0
Zero-shot	Flan-T5$_{\text{Base}}$	250M	36.1	40.4	39.8	27.0	40.6	8.8
	Flan-T5$_{\text{Base}}$ w/ AR (Mallen et al., 2022)	250M	42.8	43.5	44.0	35.8	50.0	29.4
	Flan-T5$_{\text{Base}}$ w/ AAR$_{\text{Contriever}}$ (Ours)	250M	44.4	**44.7**	**47.7**	35.8	52.2	31.9
	Flan-T5$_{\text{Base}}$ w/ AAR$_{\text{ANCE}}$ (Ours)	250M	**44.8**	42.2	46.4	**39.0**	**53.2**	**37.7**
Large Setting: T5 Large Size								
Few-shot	Atlas$_{\text{Large}}$ FT (Izacard et al., 2022)	770M	38.9	37.3	41.7	32.3	44.9	n.a.
	Flan-T5$_{\text{Large}}$	780M	45.1	47.7	53.5	34.4	49.2	9.3
Zero-shot	Flan-T5$_{\text{Large}}$	780M	44.8	46.3	51.4	34.8	50.6	7.2
	Flan-T5$_{\text{Large}}$ w/ AR	780M	49.8	50.0	55.6	38.4	59.5	29.6
	Flan-T5$_{\text{Large}}$ w/ AAR$_{\text{Contriever}}$ (Ours)	780M	**51.8**	**50.8**	**59.7**	**39.4**	**61.8**	33.4
	Flan-T5$_{\text{Large}}$ w/ AAR$_{\text{ANCE}}$ (Ours)	780M	50.4	48.0	58.1	39.3	60.2	**39.3**
XL Setting: T5 XL Size								
Few-shot	Atlas$_{\text{XL}}$ FT	3B	42.3	40.0	46.8	35.0	48.1	n.a.
	Flan-T5$_{\text{XL}}$	3B	51.6	55.0	61.1	36.8	59.5	11.1
Zero-shot	Flan-T5$_{\text{XL}}$	3B	51.2	55.5	57.4	38.1	58.7	11.3
	Flan-T5$_{\text{XL}}$ w/ AR	3B	55.5	56.7	64.5	43.0	62.6	33.7
	Flan-T5$_{\text{XL}}$ w/ AAR$_{\text{Contriever}}$ (Ours)	3B	**56.7**	57.7	**65.4**	**43.6**	**65.1**	31.5
	Flan-T5$_{\text{XL}}$ w/ AAR$_{\text{ANCE}}$ (Ours)	3B	56.2	**59.4**	64.8	41.5	64.9	**38.0**
Giant Setting: Over 70B Size								
Few-shot	Chinchilla (Hoffmann et al., 2022)	70B	67.5	63.6	79.3	55.0	73.9	n.a.
	OPT-IML-Max (Iyer et al., 2022)	175B	47.1	n.a.	n.a.	n.a.	n.a.	n.a.
	InstructGPT (Ouyang et al., 2022)	175B	60.5	62.0	71.8	44.3	70.1	35.2
Zero-shot	GAL (Taylor et al., 2022)	120B	52.6	n.a.	n.a.	n.a.	n.a.	n.a.
	OPT-IML-Max	175B	49.1	n.a.	n.a.	n.a.	n.a.	n.a.
	InstructGPT	175B	60.2	**65.7**	68.0	46.1	66.5	34.7
	InstructGPT w/ AR	175B	60.5	62.2	71.3	44.7	69.7	43.3
	InstructGPT w/ AAR$_{\text{Contriever}}$ (Ours)	175B	61.5	64.5	**73.1**	45.0	69.9	43.9
	InstructGPT w/ AAR$_{\text{ANCE}}$ (Ours)	175B	**62.2**	62.0	72.0	**49.2**	**70.7**	**52.0**

Caption: Main results on MMLU and PopQA. We group the methods by the parameters. Our L_s is Flan-T5$_{\text{Base}}$. **AAR$_{\text{Contriever}}$**: AAR initialized from Contriever; **AAR$_{\text{ANCE}}$**: AAR initialized from ANCE; **FT:** fine-tuning; AR: adaptive retrieval. Unspecified methods represent direct prompting.

- The main results demonstrate that, with the assistance of a generic AAR, target LMs of different sizes and architectures can significantly outperform their standalone baselines in the zero-shot setting.
- AAR outperforms other augmentation methods like few-shot prompting and adaptive retrieval, as they may not offer as extensive evidence text as AAR does.

Experiments

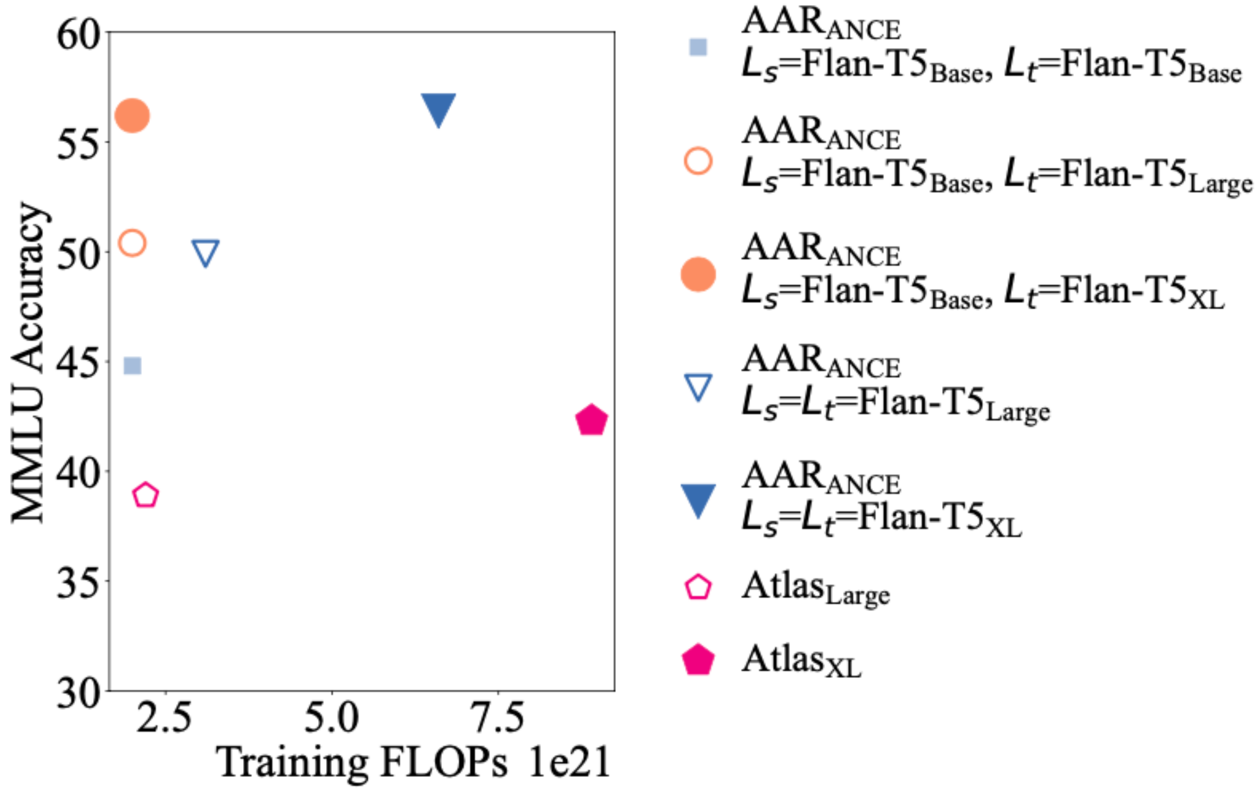
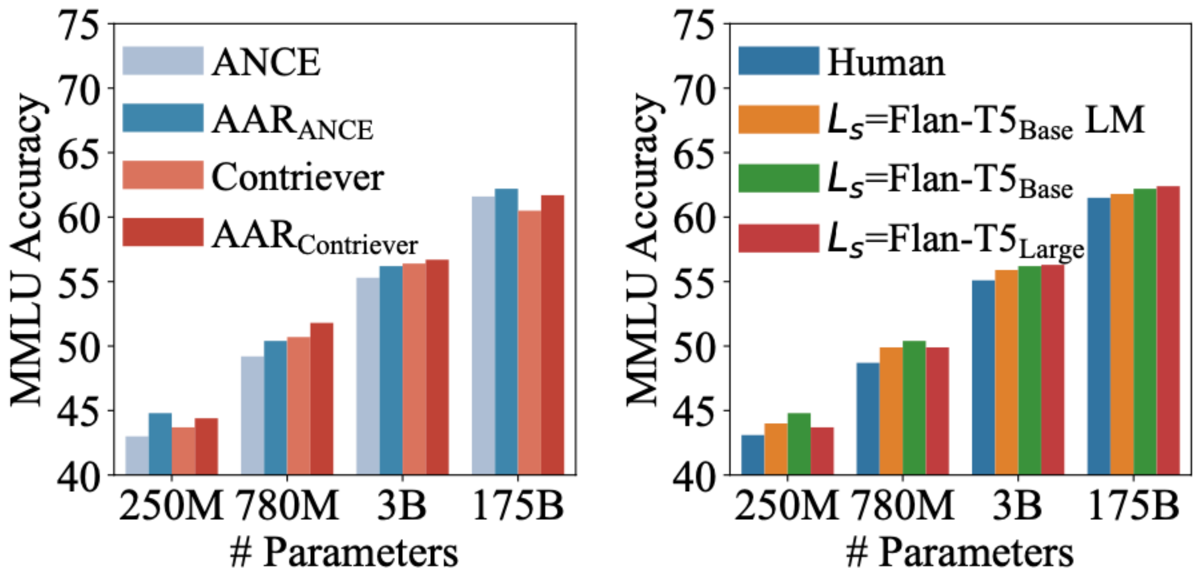


Figure 3: Training FLOPs of $\mathrm{AAR}_{\mathrm{ANCE}}$ and Atlas.

- AAR is a highly efficient augmentation approach since it only relies on a small source LM Flan-T5Base (250M) to provide training signals and can generalize well to target LMs of larger capacities.
- Solely setting the source LM as the target LM (represented by the inverted triangles) does not significantly enhance the MMLU accuracy.
- However, it may triple the training budget required. Only using a small source LM is able to outperform the powerful Atlas by large margins with fewer training FLOPs.

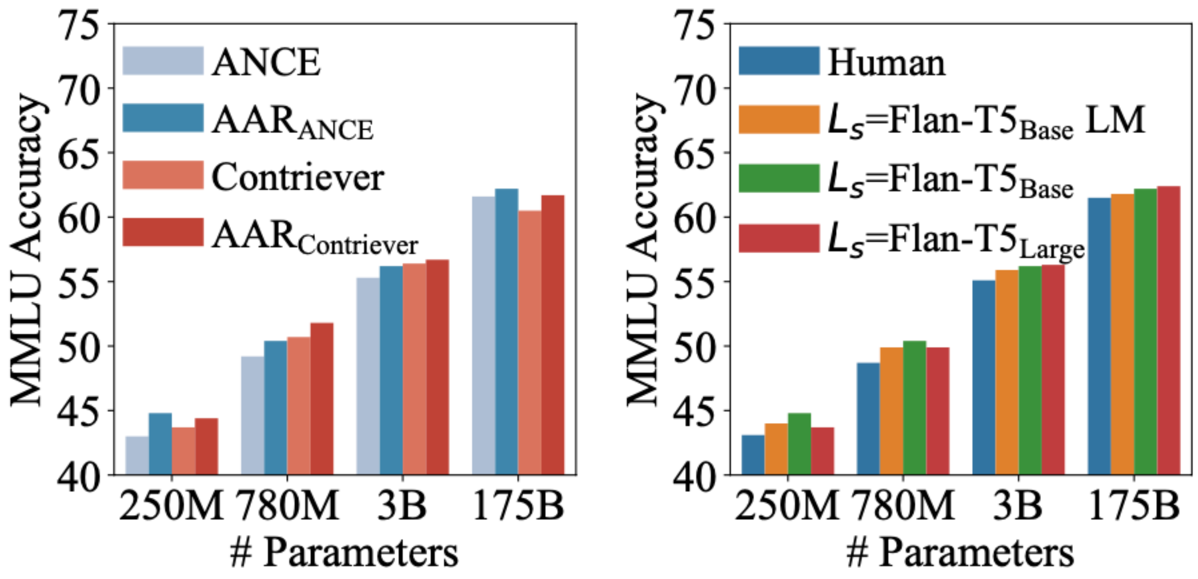
(Atlas is a SOTA retrieval-augmented LM, which jointly pre-trains the retriever with the LM using unsupervised data and fine-tunes the retriever via the attention distillation on few-shot data.)



(a) Pre-trained retrievers. (b) Positive docs selection.

Figure 4: AAR's performance when (a) using different pre-trained retrievers and (b) trained with different positive documents, using Flan-T5$_{\text{Base}}$ (250M), Flan-T5$_{\text{Large}}$ (780M), Flan-T5$_{\text{XL}}$ (3B), InstructGPT (175B) as L_t. The retriever in (b) is initialized from ANCE.

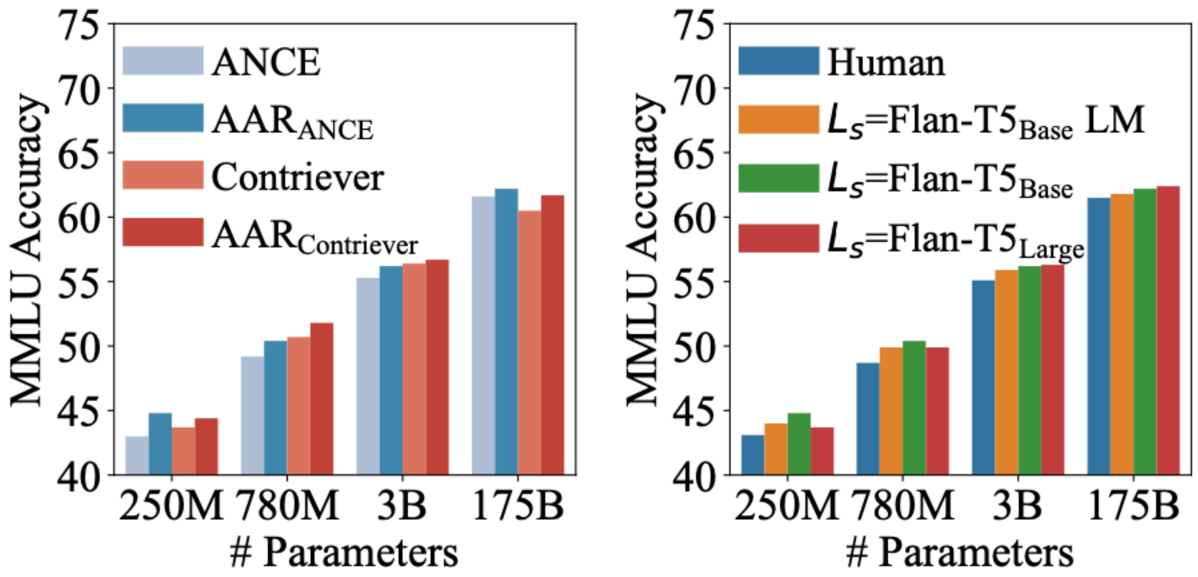
- Augmentation-adapted training can bring additional improvements compared to the pre-trained retrievers.
- ANCE benefits more from augmentation-adapted training than Contriever.
- This may be due to the fact that Contriever has been already intensively pre-trained on massive data augmentations as well as MS MARCO whereas ANCE is trained only on MS MARCO.



(a) Pre-trained retrievers. (b) Positive docs selection.

Figure 4: AAR's performance when (a) using different pre-trained retrievers and (b) trained with different positive documents, using Flan-T5$_{Base}$ (250M), Flan-T5$_{Large}$ (780M), Flan-T5$_{XL}$ (3B), InstructGPT (175B) as L_t. The retriever in (b) is initialized from ANCE.

- We compare retrievers trained with different positive documents, including human- preferred documents annotated by search users (the blue bar), LM-preferred documents obtained by the source LM (the orange bar), and their combinations (the green bar and the red bar).
- Since the retriever has been pre-trained on user-annotated MS MARCO, simply using human-preferred documents to train it may be meaningless and therefore performs the worst among all approaches.
- Only using LM-preferred documents(橙色) demonstrates notable gains over only using human-preferred documents, and merging both human-preferred and LM-preferred documents(红色、绿色) further enhances the retriever's performance.

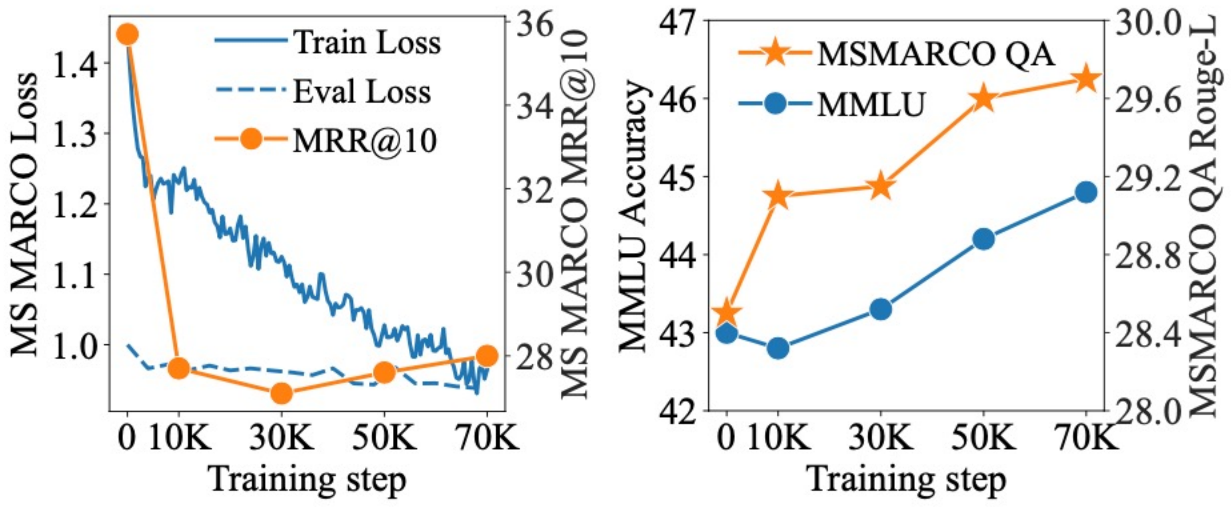


(a) Pre-trained retrievers. (b) Positive docs selection.

Figure 4: AAR's performance when (a) using different pre-trained retrievers and (b) trained with different positive documents, using Flan-T5$_{\text{Base}}$ (250M), Flan-T5$_{\text{Large}}$ (780M), Flan-T5$_{\text{XL}}$ (3B), InstructGPT (175B) as L_t. The retriever in (b) is initialized from ANCE.

- Using Flan-T5Base as source LM yields better results compared to using Flan-T5Large when the target LMs are relatively small.
- As the target LM's size increases, both approaches achieve comparable performance. Hence, our choice to utilize a small source LM in the augmentation-adapted training is reasonable and effective.

Experiments

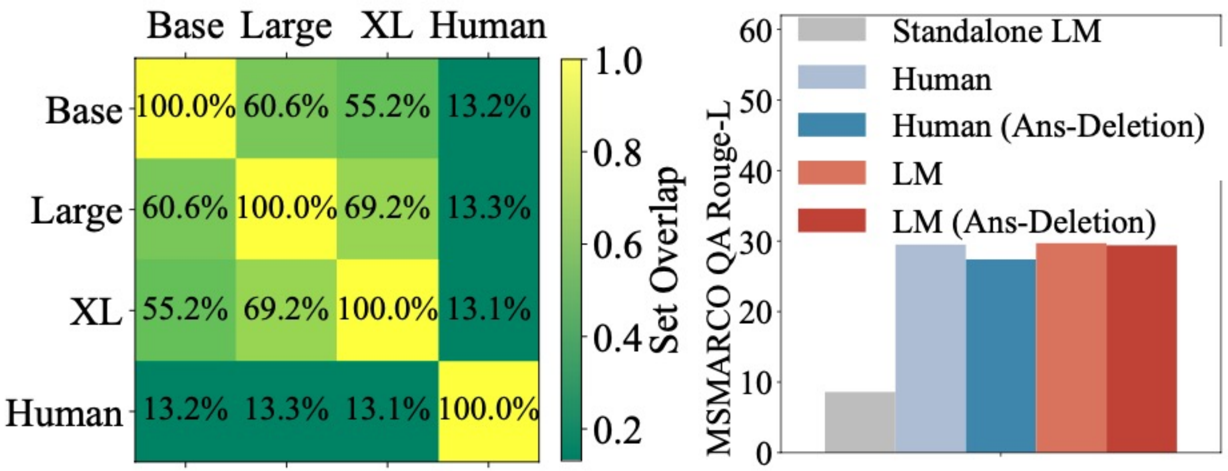


(a) Retriever's performance. (b) L_t's performance.

Figure 5: AAR's training process. (a) exhibits the retriever's (ANCE) performance on MS MARCO. (b) presents the L_t's (Flan-T5$_{\text{Base}}$) performance on MSMARCO QA and MMLU.

- At the beginning of the training, the retriever's MRR@10 on the MS MARCO drops dramatically, indicating a large distribution gap between human-preferred and LM-preferred documents.
- As the retriever's train and dev loss continually decline, the retrieval-augmented LM gradually performs better on MSMARCO QA and eventually, on MMLU.
- This result implies that LMs on different task may share common preferences, making AAR generalize well from single source task to heterogeneous target tasks.

Experiments



(a) Positive docs overlap. (b) Answer-deletion test.

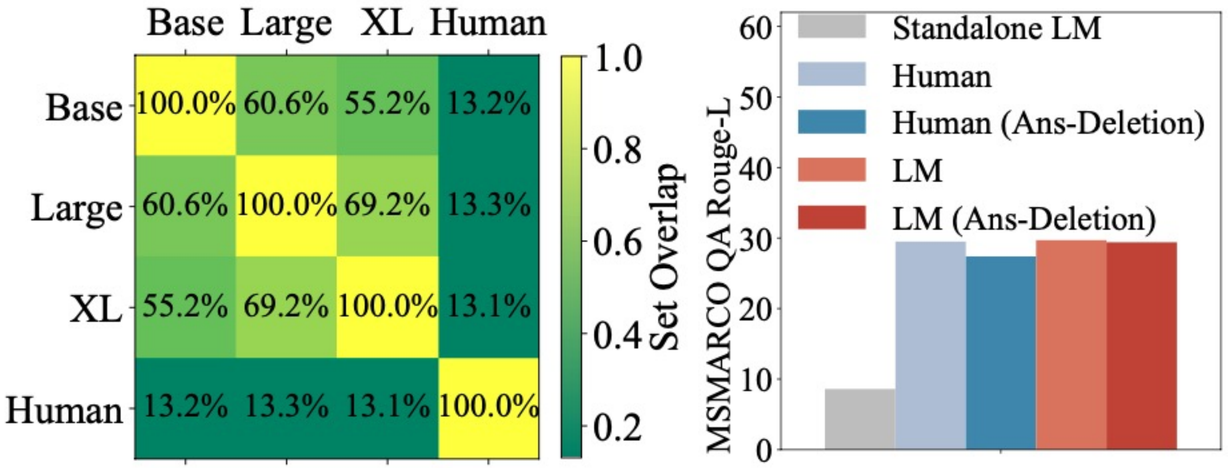
Figure 6: Analysis of LM-preferred documents. (a) shows the overlaps of positive document sets, where used LMs are Flan-T5 series. (b) presents the answer-deletion experiments on the MSMARCO QA dataset. The retriever is initialized from ANCE.

Overlap

$$O = (D_1^+ \cap D_2^+)/(D_1^+ \cup D_2^+). \tag{8}$$

- the set overlaps of the positive document sets annotated by human users and LMs are quite low (near 13%), demonstrating their distinct tendencies in selecting valuable documents.
- the overlaps between different LMs are relatively high (over 55%). This evidence provides a strong rationale for the generalization ability of AAR since LMs with different sizes tend to annotate similar positive documents.
- LMs whose sizes are closer generally possess higher overlaps. This implies a better generalization ability of the AAR to the LMs whose capacity is near the source LM.

Experiments



(a) Positive docs overlap. (b) Answer-deletion test.

Figure 6: Analysis of LM-preferred documents. (a) shows the overlaps of positive document sets, where used LMs are Flan-T5 series. (b) presents the answer-deletion experiments on the MSMARCO QA dataset. The retriever is initialized from ANCE.

- We further examine the unique characteristics of LM-preferred documents through the answer-deletion test (i.e., deleting the exact answer span from the retrieved documents).
- After the answer-deletion, the performance of LM with the human-preferred retriever declines more significantly than with the LM-preferred retriever.
- LM-preferred documents provide helpful information from alternative perspectives.

Experiments

Question	Human-preferred Document	LM-preferred Document
what happens if you miss your cruise ship	*If you do miss the ship, go into the cruise terminal and talk with the port agents, who are in contact with both shipboard and shoreside personnel.* They can help you decide the best way to meet your ...	*The cruise line is not financially responsible for getting passengers to the next port if they miss the ship.* Your travel to the subsequent port, or home, is on your dime, as are any necessary hotel stays and meals...
what is annexation?	*Annexation is an activity in which two things are joined together, usually with a subordinate or lesser thing being attached to a larger thing.* In strict legal terms, annexation simply involves...	Annexation (Latin ad, to, and nexus, joining) is the administrative action and concept in international law relating to the *forcible transition of one state's territory by another state*. It is generally held to be an illegal act...

Table 2: Case study on MSMARCO QA. We show the human-preferred documents and the Top-1 LM-preferred documents. Red texts are gold answer spans. Green texts are related spans covering other aspects of the question.

- human-preferred document can always present the gold answer at the beginning of the text, while the LM-preferred document may not contain the exact answer.
- an LM-preferred document may (1) deliver a new perspective to answer the given question, e.g., the cruise line's responsibility if you miss your cruise ship, or (2) give a specific explanation instead of an abstract definition, e.g., "forcible transition of one state's territory by another state",
- These characteristics differ from search users who want the full information and can further assist LMs in knowledge-based reasoning.

Experiments

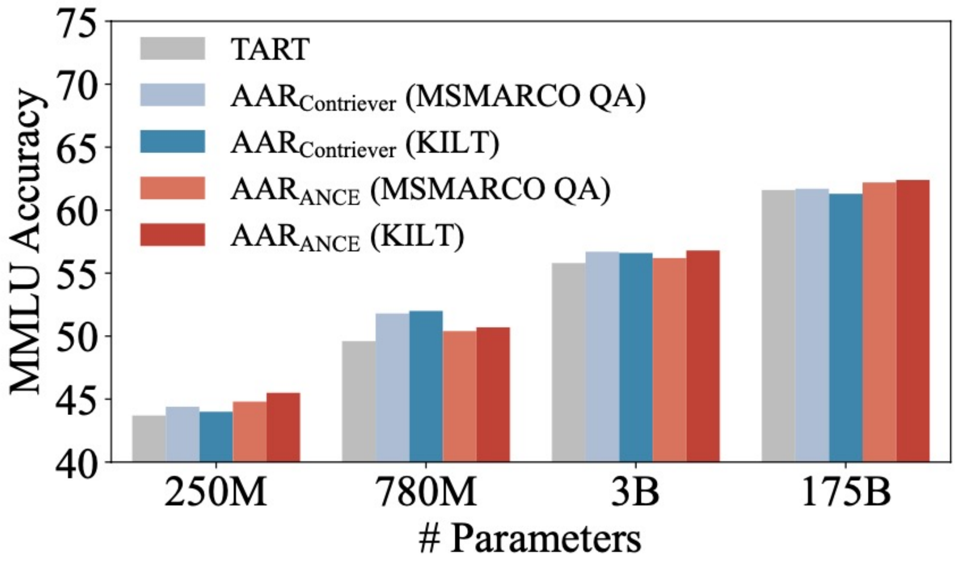


Figure 7: Comparison between single-task (MS-MARCO QA) and multi-task (KILT) trained AAR. TART (Asai et al., 2022) is a multi-task instruction-finetuned retriever that has not been finetuned with LM-preferred signals.

- ANCE trained with multi-task KILT can consistently outperform the single-task MSMARCO QA, proving the better generalization ability brought by multi-task augmentation-adapted training.
- Contriever does not benefit greatly from multi-task training. We conjecture that this is because Contriever has been pre-trained with multiple for-mats of data augmentations and thus generalizes better to new data distribution than ANCE.

Experiments

Corpora	MMLU					PopQA
	All	Hum.	Soc. Sci.	STEM	Other	All
MS MARCO	**44.8**	42.2	**46.4**	**39.0**	**53.2**	13.6
KILT-Wikipedia	42.6	**42.5**	45.9	34.3	50.5	**37.7**
Standalone LM	36.1	40.4	39.8	27.0	40.6	8.8

Table 3: Performance with different retrieval corpora, using Flan-T5$_{\text{Base}}$ as L_t and AAR$_{\text{ANCE}}$ as retriever.

- On MMLU, using MS MARCO as the retrieval corpus improves the LM more compared to KILT-Wikipedia. (the retriever has been trained with MS MARCO corpus and thus holds better retrieval performance on it.)
- On PopQA, model performance will drop by large margins if we use MS MARCO as the retrieval corpus instead of KILT-Wikipedia. (the PopQA dataset is sampled from Wikidata and designed for long-tail questions)

Experiments

Settings	Methods	MMLU All	PopQA All
Few-shot	OPT (Zhang et al., 2022)	26.0	12.3
	GPT-neo (Black et al., 2021)	28.7	11.3
Zero-shot	OPT	22.7	12.0
	GPT-neo	25.3	9.9
	OPT GenRead	22.3	12.2
	GPT-neo GenRead	24.4	11.9
	OPT w/ $\mathrm{AAR}_{\mathrm{Contriever}}$ (Ours)	23.2	29.1
	GPT-neo w/ $\mathrm{AAR}_{\mathrm{Contriever}}$ (Ours)	25.2	27.8
	OPT w/ $\mathrm{AAR}_{\mathrm{ANCE}}$ (Ours)	23.7	**32.9**
	GPT-neo w/ $\mathrm{AAR}_{\mathrm{ANCE}}$ (Ours)	**26.6**	30.1

Table 4: Results of using models that have not been multi-task instruction-finetuned as L_t. We experiment with the 1.3B version of OPT and GPT-neo.

- To examine if AAR works for unseen LMs that may lack zero-shot generalization ability, we report the results of using OPT and GPT-neo as Lt, which have not been multi-task instruction-finetuned.
- AAR improves both LMs marginally on MMLU while achieving significant gains on PopQA. (LMs can benefit more easily from retrieval augmentation on the knowledge-probing task like PopQA, where the answer span can be directly acquired from the retrieved documents.)