Carnegie Mellon University
School of Computer Science

NEULAB

BOSCH
Invented for life

Retrieval as Attention: End-to-end Learning of Retrieval and Reading within a Single Transformer

Zhengbao Jiang[1]*, Luyu Gao[1]*, Jun Araki[2], Haibo Ding[2],
Zhiruo Wang[1], Jamie Callan[1], Graham Neubig[1]

CMU[1]

Bosch Research[2]

zhengbaj@cs.cmu.edu luyug@cs.cmu.edu

* equal contribution

Retrieval is crucial to knowledge-intensive tasks

- Knowledge-intensive tasks requires world knowledge
  - Question answering (QA)
  - Fact checking
  - Dialogue

Retrieval is crucial to knowledge-intensive tasks

- Knowledge-intensive tasks requires world knowledge
  - Question answering (QA)
  - Fact checking
  - Dialogue
- Retrieval-augmented models (i.e., RLMs) are better than parametric models (i.e., LMs).
  - Retrieving from an external corpus offers higher knowledge coverage and robustness
  - Knowledge can be easily updated by updating the corpus

However, RLMs are hard to build, train, and adapt

Widely adopted paradigm: retrieval and reading are two different sub-tasks so they should be treated separately.

However, RLMs are hard to build, train, and adapt

Widely adopted paradigm: retrieval and reading are two different sub-tasks so they should be treated separately.

- Retrievers and readers are two separate models with different architectures and training recipes.
- Retrievers and readers are glued together in an ad-hoc and post-hoc way for downstream tasks.
- Retrievers and readers are hard to optimize end-to-end and adapt.

Existing RLMs for knowledge-intensive tasks

Models	Architecture			Initialization		
	Retriever	Reader	Relationship	Retriever	Reader	Retriever warmup
REALM (Guu et al., 2020)	encoder	encoder	two models	BERT	BERT	yes (ICT, SSM)
RAG (Lewis et al., 2020)	encoder	enc-dec	two models	DPR	BART	no
FiD-KD (Izacard et al., 2021)	encoder	enc-dec	two models	DPR	T5	no
EMDR^2 (Sachan et al., 2021)	encoder	enc-dec	two models	BERT	T5	yes (ICT, SSM)
YONO (Lee et al., 2021)	encoder	enc-dec	same model	T5	T5	yes (SSM)
Atlas (Izacard et al., 2022)	encoder	enc-dec	two models	Contriever	T5	no

Two models with different architectures glued together

Existing RLMs for knowledge-intensive tasks

Models	Architecture			Initialization		
	Retriever	Reader	Relationship	Retriever	Reader	Retriever warmup
REALM (Guu et al., 2020)	encoder	encoder	two models	BERT	BERT	yes (ICT, SSM)
RAG (Lewis et al., 2020)	encoder	enc-dec	two models	DPR	BART	no
FiD-KD (Izacard et al., 2021)	encoder	enc-dec	two models	DPR	T5	no
EMDR^2 (Sachan et al., 2021)	encoder	enc-dec	two models	BERT	T5	yes (ICT, SSM)
YONO (Lee et al., 2021)	encoder	enc-dec	same model	T5	T5	yes (SSM)
Atlas (Izacard et al., 2022)	encoder	enc-dec	two models	Contriever	T5	no

Two models with different architectures glued together

Relying on specifically trained retrievers as initialization

Existing RLMs for knowledge-intensive tasks

Models	Architecture			Initialization		
	Retriever	Reader	Relationship	Retriever	Reader	Retriever warmup
REALM (Guu et al., 2020)	encoder	encoder	two models	BERT	BERT	yes (ICT, SSM)
RAG (Lewis et al., 2020)	encoder	enc-dec	two models	DPR	BART	no
FiD-KD (Izacard et al., 2021)	encoder	enc-dec	two models	DPR	T5	no
EMDR^2 (Sachan et al., 2021)	encoder	enc-dec	two models	BERT	T5	yes (ICT, SSM)
YONO (Lee et al., 2021)	encoder	enc-dec	same model	T5	T5	yes (SSM)
Atlas (Izacard et al., 2022)	encoder	enc-dec	two models	Contriever	T5	no

Two models with different architectures glued together

Relying on specifically trained retrievers as initialization

Relying on retrieval-specific warmup

Motivations

Our path: retrieval and reading share the same goal so “retrievers” and “readers” should be fused more organically and trained together.

Motivations

Our path: retrieval and reading share the same goal so “retrievers” and “readers” should be fused more organically and trained together.

More specifically:

- perform retrieval and reading within a single model
- fully end-to-end training only using end-task (QA) annotations
- achieve competitive performance on both retrieval and end-task

Key idea

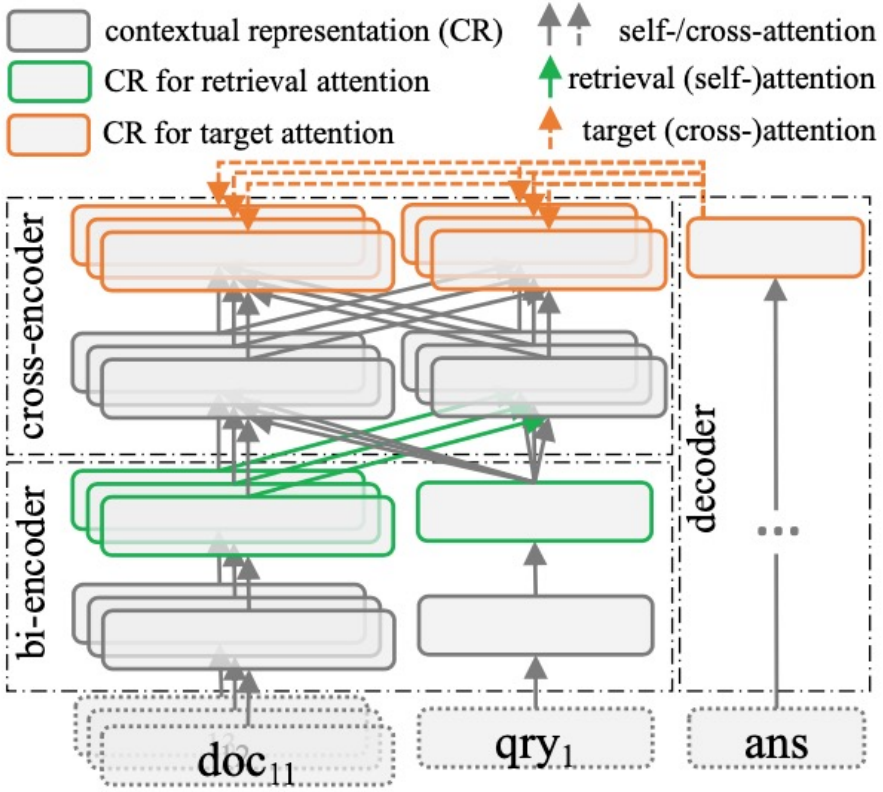
Directly use attention in Transformers to perform retrieval

- Attention is essentially connecting similar tokens in the input.
- Can we use attention to connect question tokens with doc tokens?

Retrieval as **Att**ention (**ReAtt**)

Model architecture: “retriever” and “reader” in a T5 model

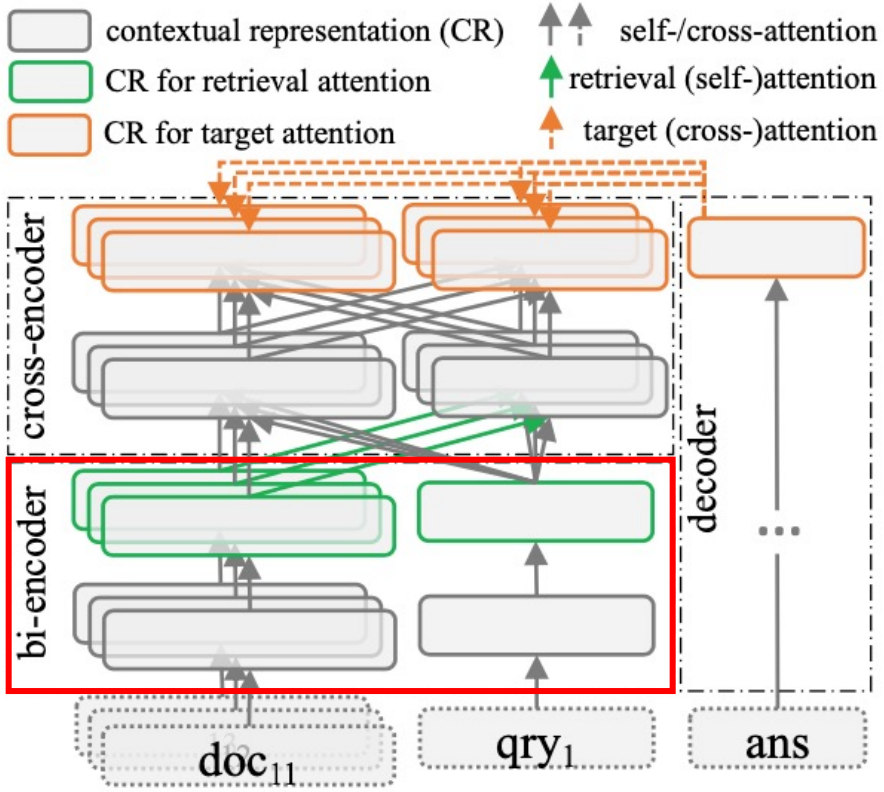
- Based on encoder-decoder architecture T5



Retrieval as **Att**ention (**ReAtt**)

Model architecture: “retriever” and “reader” in a T5 model

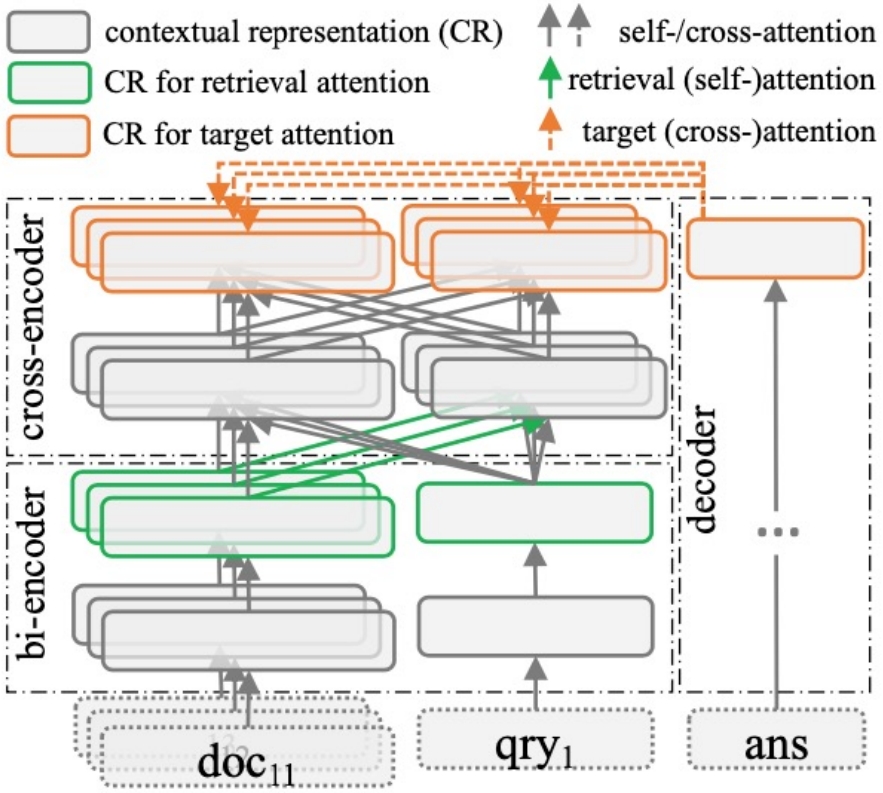
- Based on encoder-decoder architecture T5
- Bi-encoder retriever
  - The bottom half (B=2 layers) of encoder as “retriever”
  - Encode questions and documents independently



Retrieval as **Att**ention (**ReAtt**)

Model architecture: “retriever” and “reader” in a T5 model

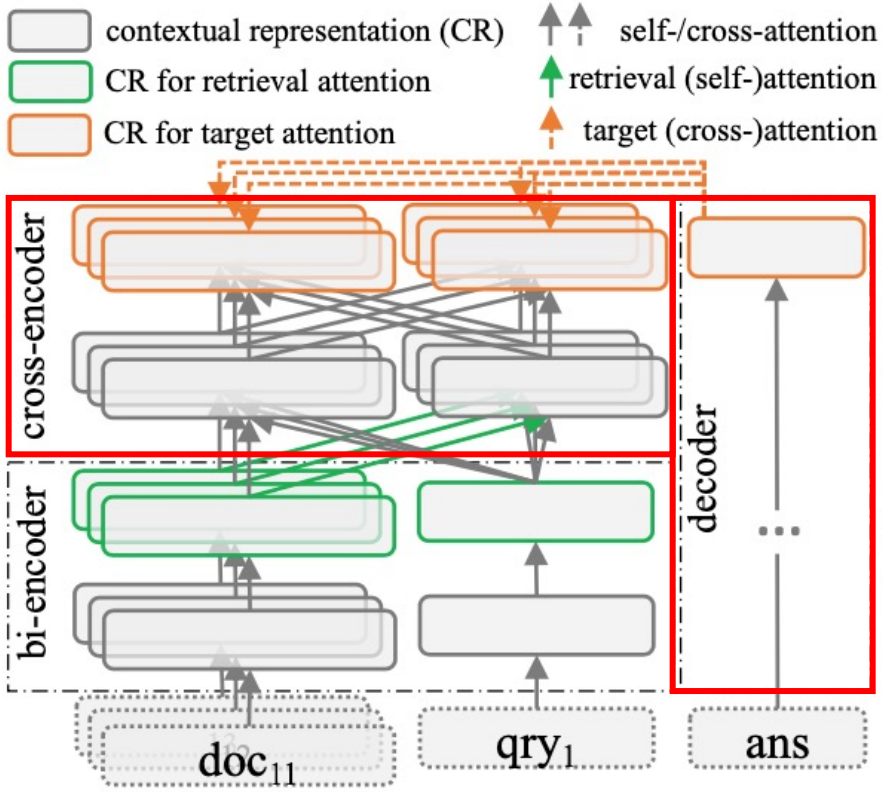
- Based on encoder-decoder architecture T5
- Bi-encoder retriever
  - The bottom half (B=2 layers) of encoder as “retriever”
  - Encode questions and documents independently
- Retrieval attention
  - The attention from question to document at layer B+1



Retrieval as Attention (ReAtt)

Model architecture: “retriever” and “reader” in a T5 model

- Based on encoder-decoder architecture T5
- Bi-encoder retriever
  - The bottom half (B=2 layers) of encoder as “retriever”
  - Encode questions and documents independently
- Retrieval attention
  - The attention from question to document at layer B+1
- Reader
  - The top half of encoder and decoder as “reader”
  - Encode questions documents jointly
  - Decode by fusing all (i.e., fusion-in-decoder)



Retrieval as **Att**ention (**ReAtt**)

From token-level attention to doc-level relevance

- Aggregate the attention matrix from question to doc tokens

Retrieval as **Att**ention (**ReAtt**)

From token-level attention to doc-level relevance

- Aggregate the attention matrix from question to doc tokens
  - The attention matrix of a specific head h in layer $B+1$ (i.e., retrieval attention)

$$A_{q,d}^{B+1,h} \in \mathbb{R}^{|q|\times|d|}$$

Retrieval as **Att**ention (**ReAtt**)

From token-level attention to doc-level relevance

- Aggregate the attention matrix from question to doc tokens
  - The attention matrix of a specific head h in layer $B+1$ (i.e., retrieval attention)

$$A_{q,d}^{B+1,h} \in \mathbb{R}^{|q|\times|d|}$$

  - mean-max aggregation, similar to ColBERT (Khattab et al. 2020)

$$r_h(q,d) = \underset{\mathrm{q}}{\mathrm{mean}}(\underset{\mathrm{d}}{\mathrm{max}}(A_{q,d}^{B+1,h}))$$

Retrieval as **Att**ention (**ReAtt**)

From token-level attention to doc-level relevance

- Aggregate the attention matrix from question to doc tokens
  - The attention matrix of a specific head h in layer $B+1$ (i.e., retrieval attention)

$$A_{q,d}^{B+1,h} \in \mathbb{R}^{|q|\times|d|}$$

  - mean-max aggregation, similar to ColBERT (Khattab et al. 2020)

$$r_h(q,d) = \underset{\mathrm{q}}{\mathrm{mean}}(\underset{\mathrm{d}}{\mathrm{max}}(A_{q,d}^{B+1,h}))$$

  - Softmax-based weight to combine all heads
    - Enforce sparse in the softmax-based weight, ends up with only **a single activate head**.

Learning retrieval as attention

- Preliminary experiments
  - the best T5 head is decent in retrieval but not as good as BM25

Learning retrieval as attention

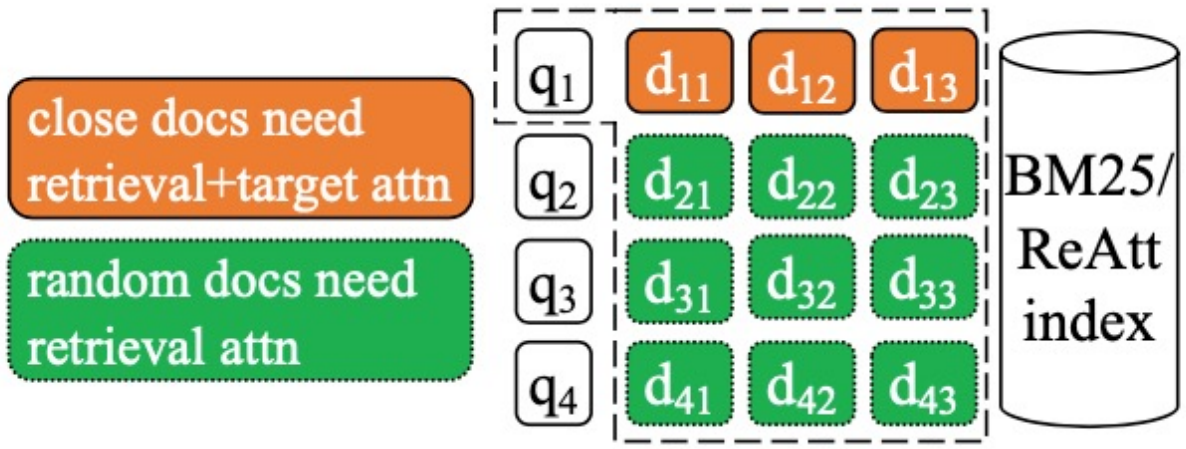
- Preliminary experiments
  - the best T5 head is decent in retrieval but not as good as BM25
- Potential reasons
  - Attention is pre-trained on local context
  - Thus, not reliable when dealing with the enormous space of a corpus

Learning retrieval as attention

- Compute attention between questions and many documents

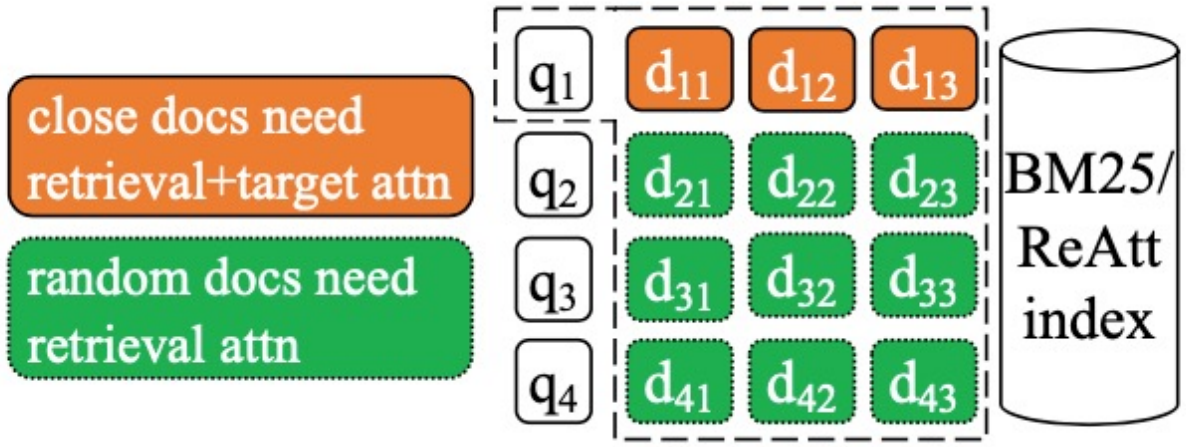
Learning retrieval as attention

- Compute attention between questions and many documents
  - In each training iteration, for each question
    - Retrieve close docs (e.g., 3) using BM25 or ReAtt



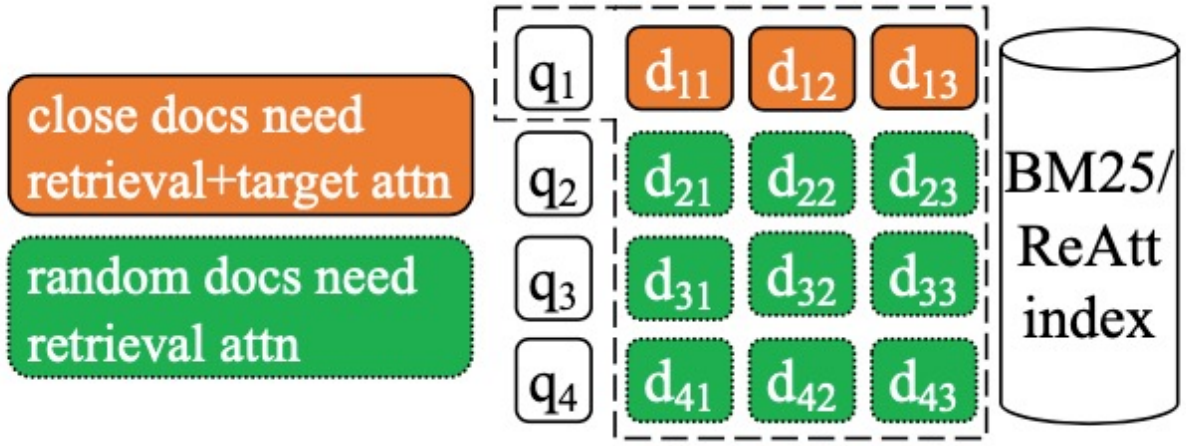
Learning retrieval as attention

- Compute attention between questions and many documents
  - In each training iteration, for each question
    - Retrieve close docs (e.g., 3) using BM25 or ReAtt
    - Use close docs of other question in the same batch as random docs (e.g., 3*3=9)



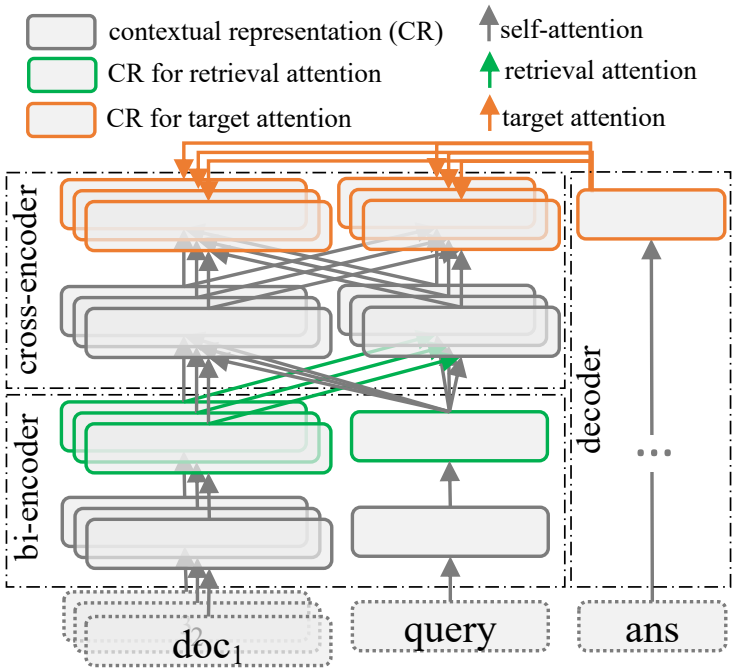
Learning retrieval as attention

- Compute attention between questions and many documents
  - In each training iteration, for each question
    - Retrieve close docs (e.g., 3) using BM25 or ReAtt
    - Use close docs of other question in the same batch as random docs (e.g., 3*3=9)
    - Compute retrieval attention for both close and random docs (e.g., 3+3*3=12)



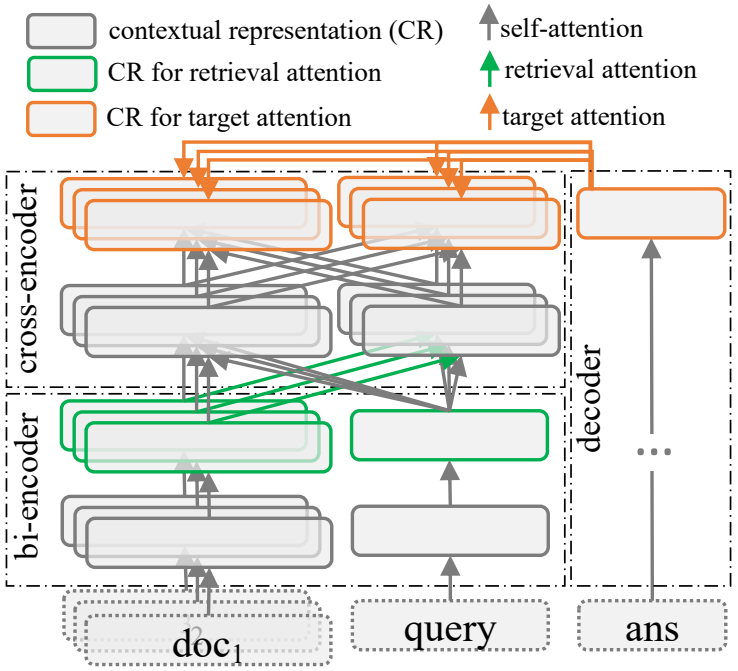
Learning retrieval as attention

- Adjust attention with decoder-to-encoder distillation
  - Cross attention reflects the contribution of each doc to generating the output.



Learning retrieval as attention

- Adjust attention with decoder-to-encoder distillation
  - Cross attention reflects the contribution of each doc to generating the output.
  - Minimize the discrepancy between retrieval attention and cross attention, similar to FiD-KD (Izacard et al., 2021).



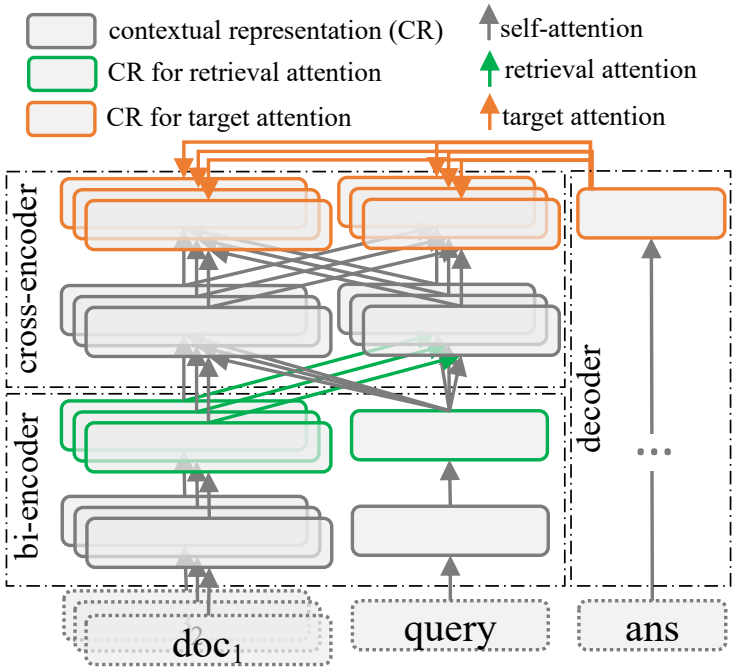
Learning retrieval as attention

- Adjust attention with decoder-to-encoder distillation
  - Cross attention reflects the contribution of each doc to generating the output.
  - Minimize the discrepancy between retrieval attention and cross attention, similar to FiD-KD (Izacard et al., 2021).

distribution to be learned from retrieval attn

$$\mathrm{KL}(P_1^{\mathrm{cross}}, \ldots, P_{12}^{\mathrm{cross}} | P_1^{\mathrm{ret}}, \ldots, P_{12}^{\mathrm{ret}})$$

gold distribution from cross attn



Learning retrieval as attention

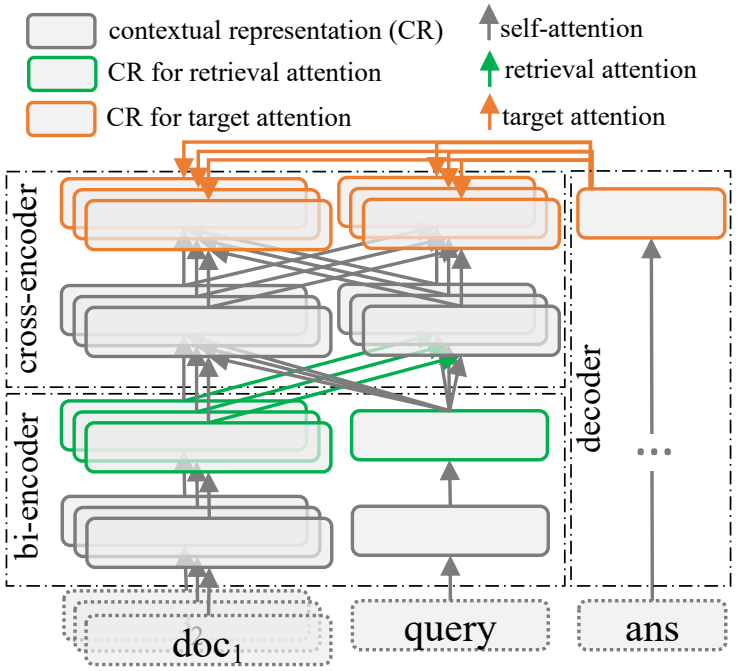
- Adjust attention with decoder-to-encoder distillation
  - Cross attention reflects the contribution of each doc to generating the output.
  - Minimize the discrepancy between retrieval attention and cross attention, similar to FiD-KD (Izacard et al., 2021).

distribution to be learned from retrieval attn

$$\mathrm{KL}(P_1^{\mathrm{cross}}, \ldots, P_{12}^{\mathrm{cross}} | P_1^{\mathrm{ret}}, \ldots, P_{12}^{\mathrm{ret}})$$

gold distribution from cross attn

Final loss: $\mathrm{QA} + \alpha \cdot \mathrm{KL}$



Experiments

- Datasets
  - In-domain (Wikipedia): Natural Questions (NQ)
  - Out-of-domain (biomed, finance, code, science, COVID, etc):
    - Zero-shot evaluation
    - Supervised adaptation: BioASQ (QA)
    - Unsupervised adaptation: CQADupStack, TREC-COVID, SCIDOCS, SciFact

Experiments

- Datasets
  - In-domain (Wikipedia): Natural Questions (NQ)
  - Out-of-domain (biomed, finance, code, science, COVID, etc):
    - Zero-shot evaluation
    - Supervised adaptation: BioASQ (QA)
    - Unsupervised adaptation: CQADupStack, TREC-COVID, SCIDOCS, SciFact
- Metrics
  - Retrieval: Recall@k and nDCG@k
  - QA: exact match (EM)

In-domain experiments

- Retrieval performance
  - ReAtt outperforms retrievers trained with retrieval annotations (e.g., ColBERT).

Models	R@1	R@5	R@20	R@100	#Params.
supervised retrievers					
BM25	23.9	45.9	63.8	78.9	-
DPR	45.9	68.1	80.0	85.9	220M
DPR^{new}	52.5	72.2	81.3	87.3	220M
DPR-PAQ	-	74.2	84.0	89.2	220M
ANCE	-	-	81.9	87.5	220M
coCondenser	-	75.8	84.3	89.0	220M
DensePhrase	51.1	69.9	78.7	-	330M
ColBERT	-	-	79.1	-	110M
ColBERT-NQ	54.3	75.7	85.6	90.0	110M
semi/unsupervised retrievers					
FiD-KD	49.4	73.8	84.3	89.3	220M
YONO$_{w/o\ PT}$	-	-	72.3	82.2	165M
YONO$_{w/\ PT}$	-	75.3	85.2	90.2	165M
ReAtt $_{DPR}$	54.6	77.2	**86.1**	**90.7**	165M
ReAtt $_{BM25}$	**55.8**	**77.4**	86.0	90.4	165M

Retrieval performance on NQ. Blue indicates fair-to-compare baselines.

In-domain experiments

- Retrieval performance
  - ReAtt outperforms retrievers trained with retrieval annotations (e.g., ColBERT).
  - ReAtt outperforms other end-to-end learned retrievers without retrieval-specific initialization and warmup.

Models	R@1	R@5	R@20	R@100	#Params.
supervised retrievers					
BM25	23.9	45.9	63.8	78.9	-
DPR	45.9	68.1	80.0	85.9	220M
DPR^{new}	52.5	72.2	81.3	87.3	220M
DPR-PAQ	-	74.2	84.0	89.2	220M
ANCE	-	-	81.9	87.5	220M
coCondenser	-	75.8	84.3	89.0	220M
DensePhrase	51.1	69.9	78.7	-	330M
ColBERT	-	-	79.1	-	110M
ColBERT-NQ	54.3	75.7	85.6	90.0	110M
semi/unsupervised retrievers					
FiD-KD	49.4	73.8	84.3	89.3	220M
YONO$_{w/o\ PT}$	-	-	72.3	82.2	165M
YONO$_{w/\ PT}$	-	75.3	85.2	90.2	165M
ReAtt $_{DPR}$	54.6	77.2	**86.1**	**90.7**	165M
ReAtt $_{BM25}$	**55.8**	**77.4**	86.0	90.4	165M

Retrieval performance on NQ. Blue indicates fair-to-compare baselines.

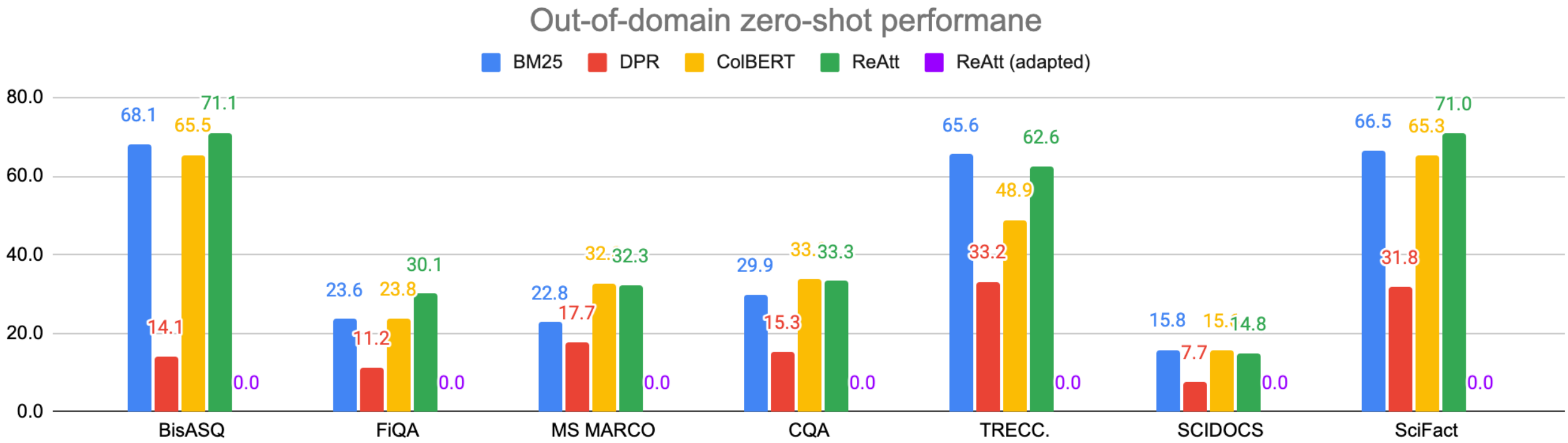
In-domain experiments

- QA performance
  - ReAtt achieves comparable QA performance with strong QA models.

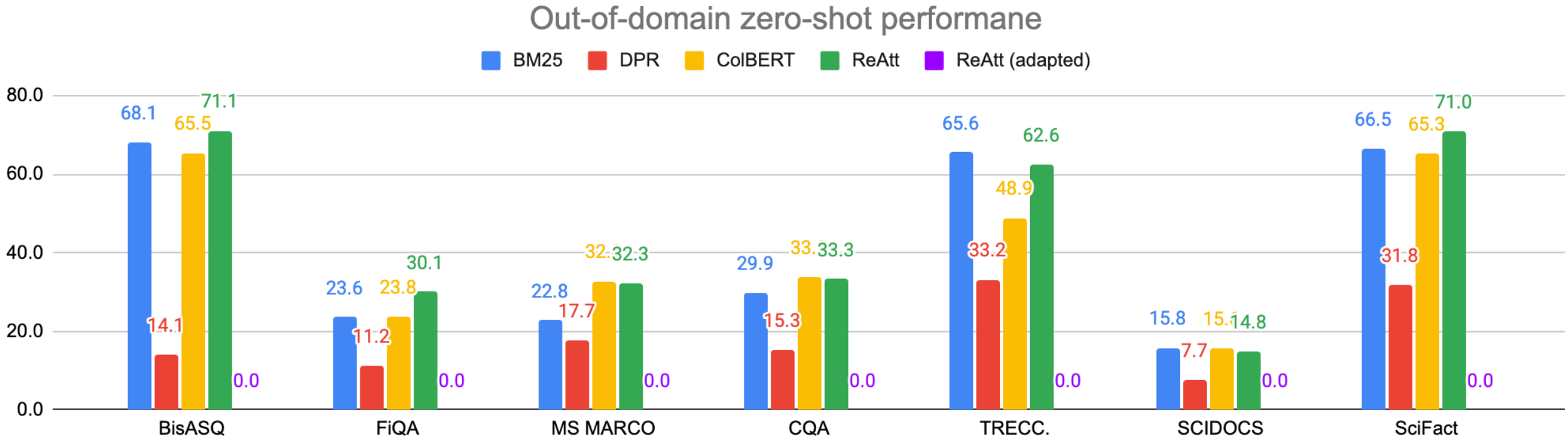
Models	EM	#Params.
ORQA (Lee et al., 2019)	33.3	330M
REALM (Guu et al., 2020)	40.4	330M
RAG (Lewis et al., 2020)	44.5	220M
FiD (Izacard and Grave, 2021b)	51.4	990M
FiD-KD (Izacard and Grave, 2021a)	54.4	990M
EMDR^2 (Sachan et al., 2021)	52.5	440M
YONO$_{\text{w/o PT}}$ (Lee et al., 2021a)	42.4	440M
YONO$_{\text{w/ PT}}$ (Lee et al., 2021a)	53.2	440M
UnitedQA (Cheng et al., 2021)	54.7	1.870B
R2-D2 (Fajcik et al., 2021)	**55.9**	1.290B
ReAtt $_{\text{DPR}}$	54.0	770M
ReAtt $_{\text{BM25}}$	54.7	770M

QA performance on NQ. Blue indicates fair-to-compare baselines.

OOD experiments (zero-shot)

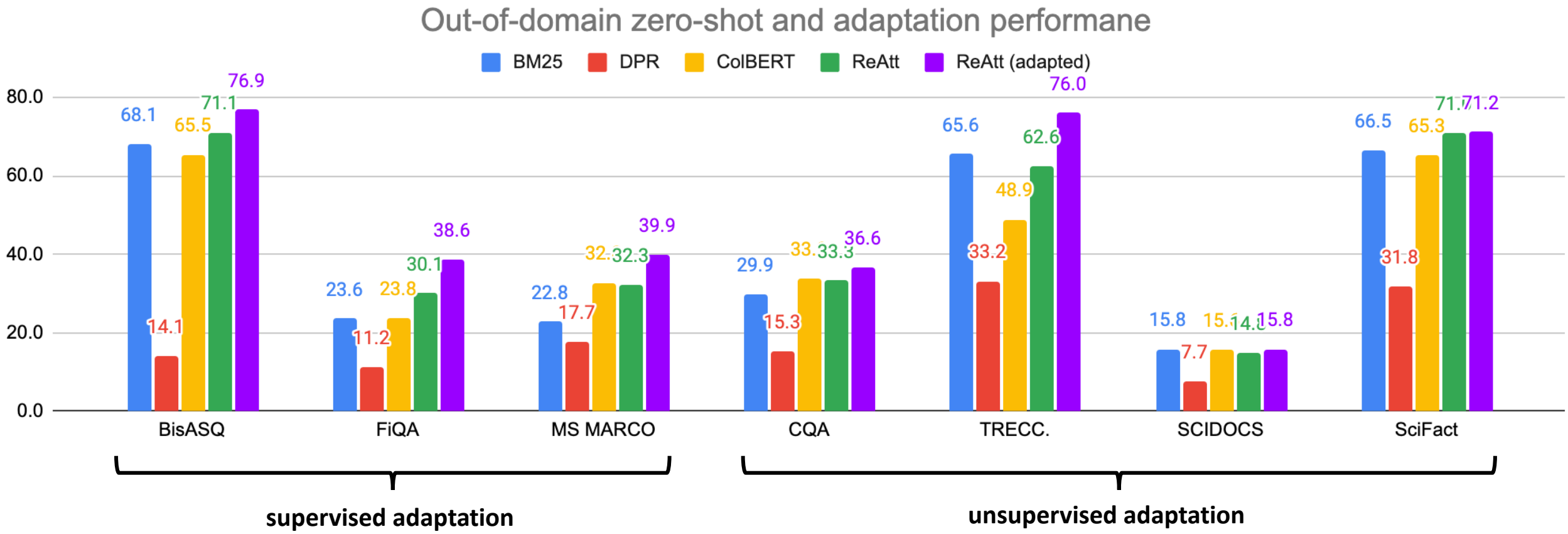


OOD experiments (zero-shot)

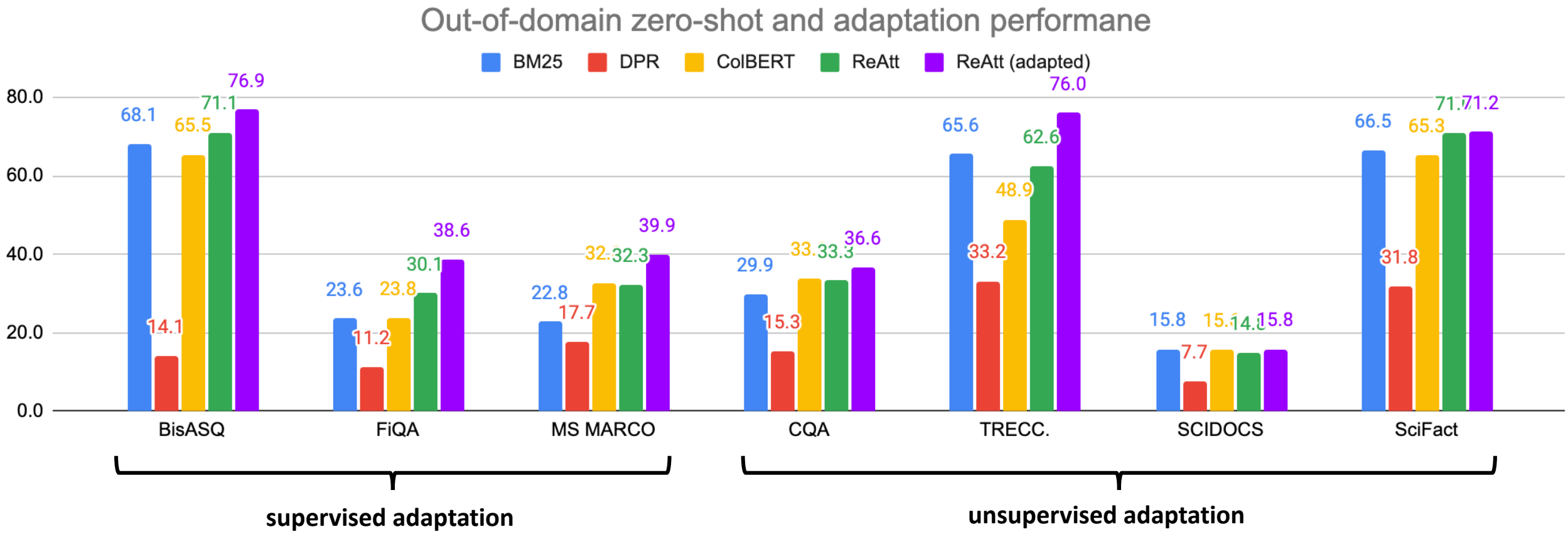


- ReAtt has strong zero-shot performance on other domains.
- ReAtt $\geq$ BM25/ColBERT $\gg$ DPR.

OOD experiments (adaptation)



OOD experiments (adaptation)



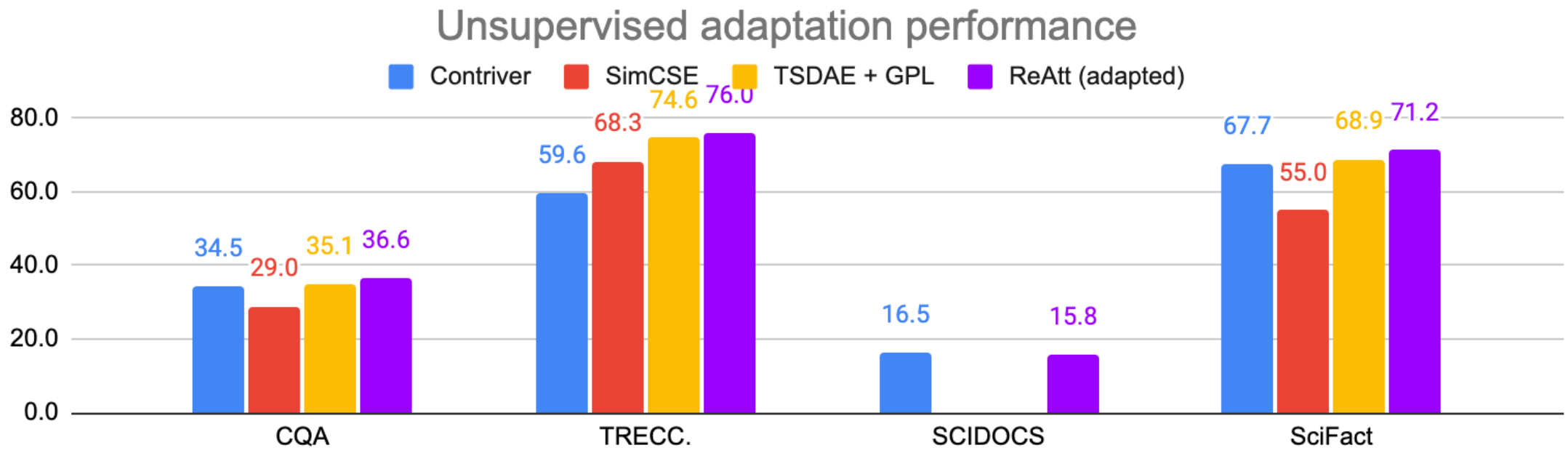
- Adaptation further improve retrieval performance by a large margin in both supervised and unsupervised settings.

OOD experiments (unsupervised adaptation)

- Setting
  - Input: masked sentence
  - Output: masked entity

OOD experiments (unsupervised adaptation)

- Setting
  - Input: masked sentence
  - Output: masked entity
- Conclusion
  - ReAtt achieves comparative or stronger performance than other adaptation methods.



OOD experiments (supervised adaptation - QA)

- Setting
  - Input: question
  - Output: answers

OOD experiments (supervised adaptation - QA)

- Setting
  - Input: question
  - Output: answers
- Conclusion
  - End-to-end adaptation of ReAtt improves both retrieval and QA.
  - Pipeline adaptation works for RAG while end2end adaptation make retrieval collapse.

Methods	**nDCG@5**	**EM**
RAG	13.0	1.3
RAG (pipeline adapted)	27.1	27.8
RAG (end2end adapted)	0.0	25.7
ReAtt	70.1	17.2
ReAtt (end2end adapted)	75.4	47.2

Retrieval and QA performance on BioASQ.

Conclusion & Future work

- Conclusion
  - Retrieval as attention (ReAtt) is a **single-model**, **single-training**, and **adaptable** solution for retrieval-augmented knowledge-intensive tasks.

Conclusion & Future work

- Conclusion
  - Retrieval as attention (ReAtt) is a **single-model**, **single-training**, and **adaptable** solution for retrieval-augmented knowledge-intensive tasks.
- Future work
  - Improve efficiency through pruning, compression, and quantization
  - Extend to multiple heads and layers
  - Better end-to-end training objectives

Q&A

Paper: https://arxiv.org/pdf/2212.02027.pdf

Code and models: https://github.com/jzbjyb/ReAtt