

# Retrieval-based LMs

- Improving language models by retrieving from trillions of tokens
- Scaling Retrieval-Based Language Models with a Trillion-Token Datastore

**Bhavya Chopra, Yichuan Wang**

October 10, 2025

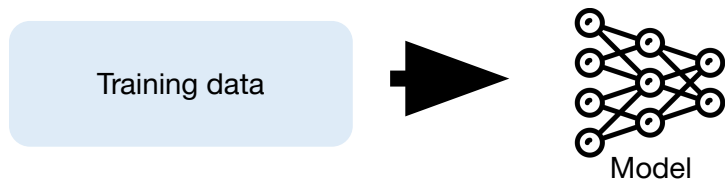
# Motivation

# Motivation

- Expensive to train and deploy large models and scale them
- Scaling up model parameters hits compute + memory bottlenecks
- LLMs must memorize factual or rare information in their weights, which is inefficient and hard to update
- LLM must be retrained from scratch when new facts emerge

# What is test-time data

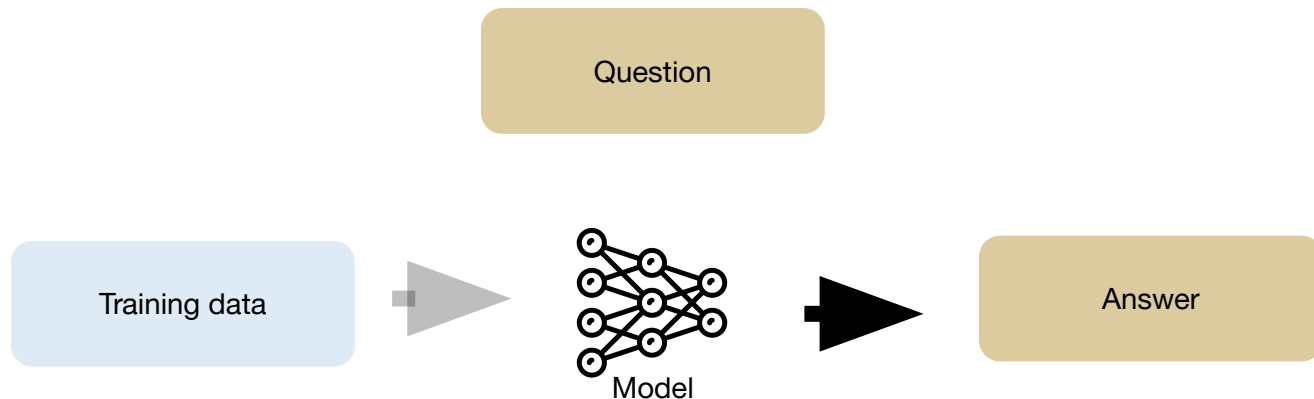
## Training:





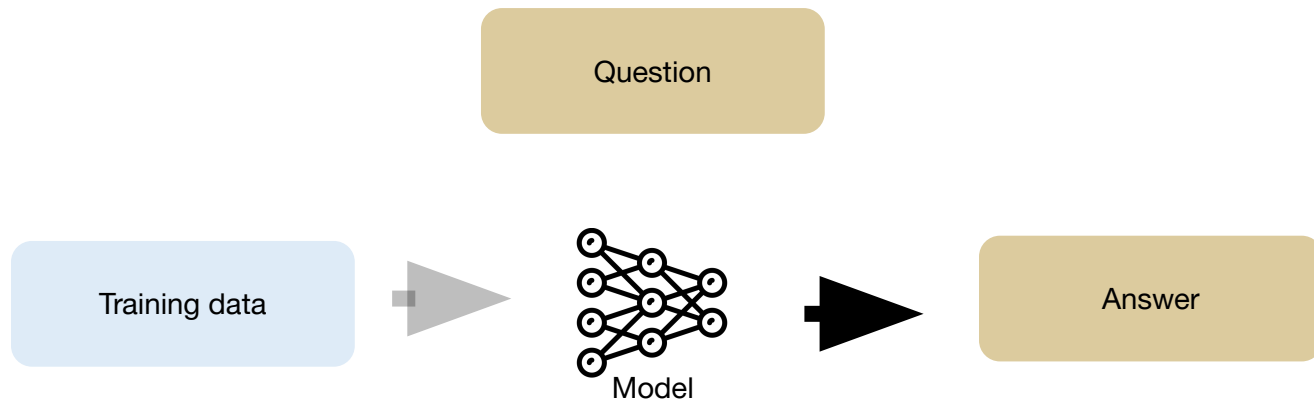
# What is test-time data

**Inference (w/o test-time data):**



# What is test-time data

## Inference (w/o test-time data):

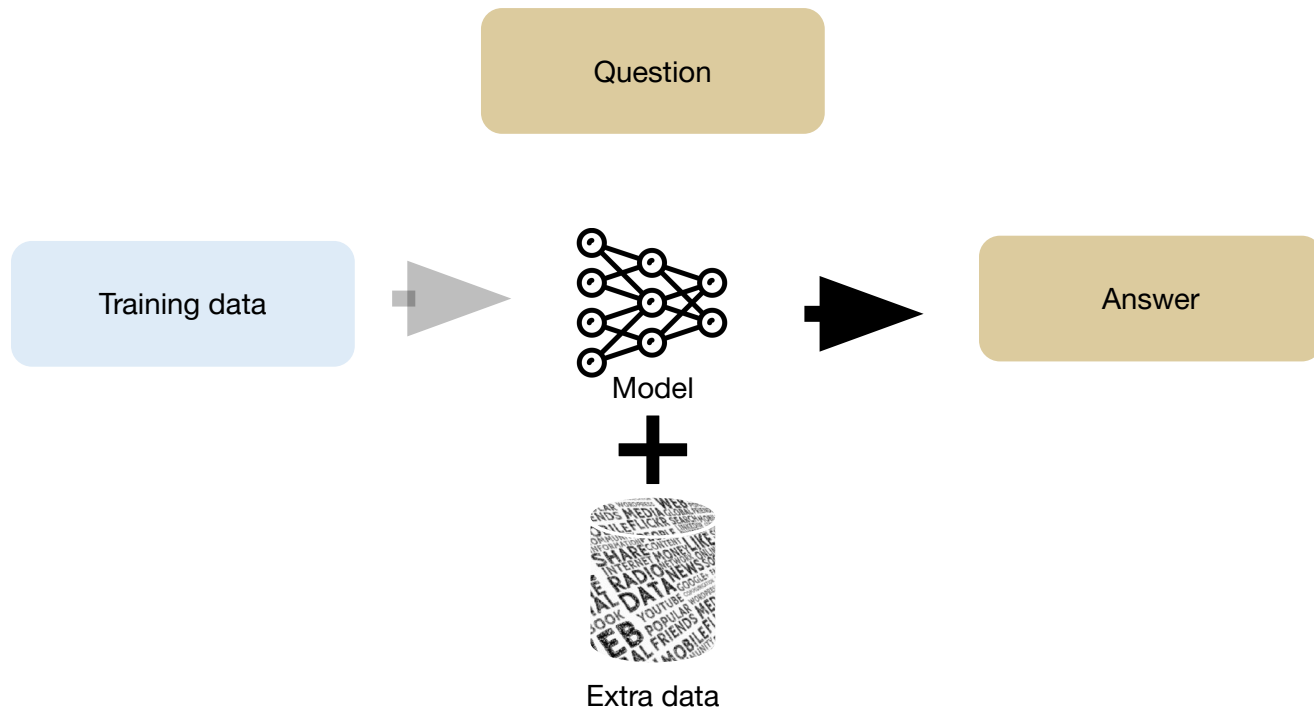


### Limitations:

1. Difficult to attribute/verify the data source;
2. The model can hallucinate a lot on long-tail knowledge;
3. Difficult to update with new knowledge.

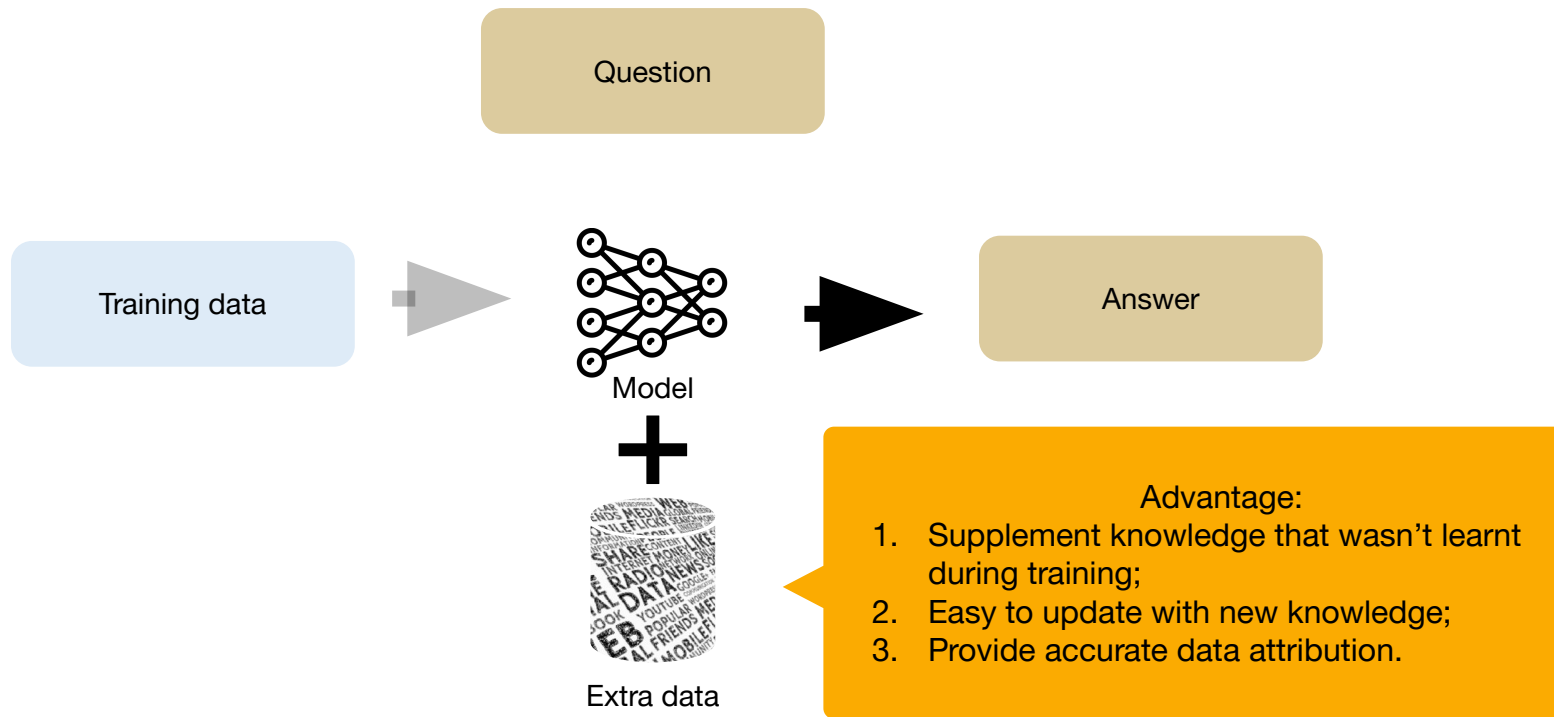
# What is test-time data

Inference (w/ test-time data):



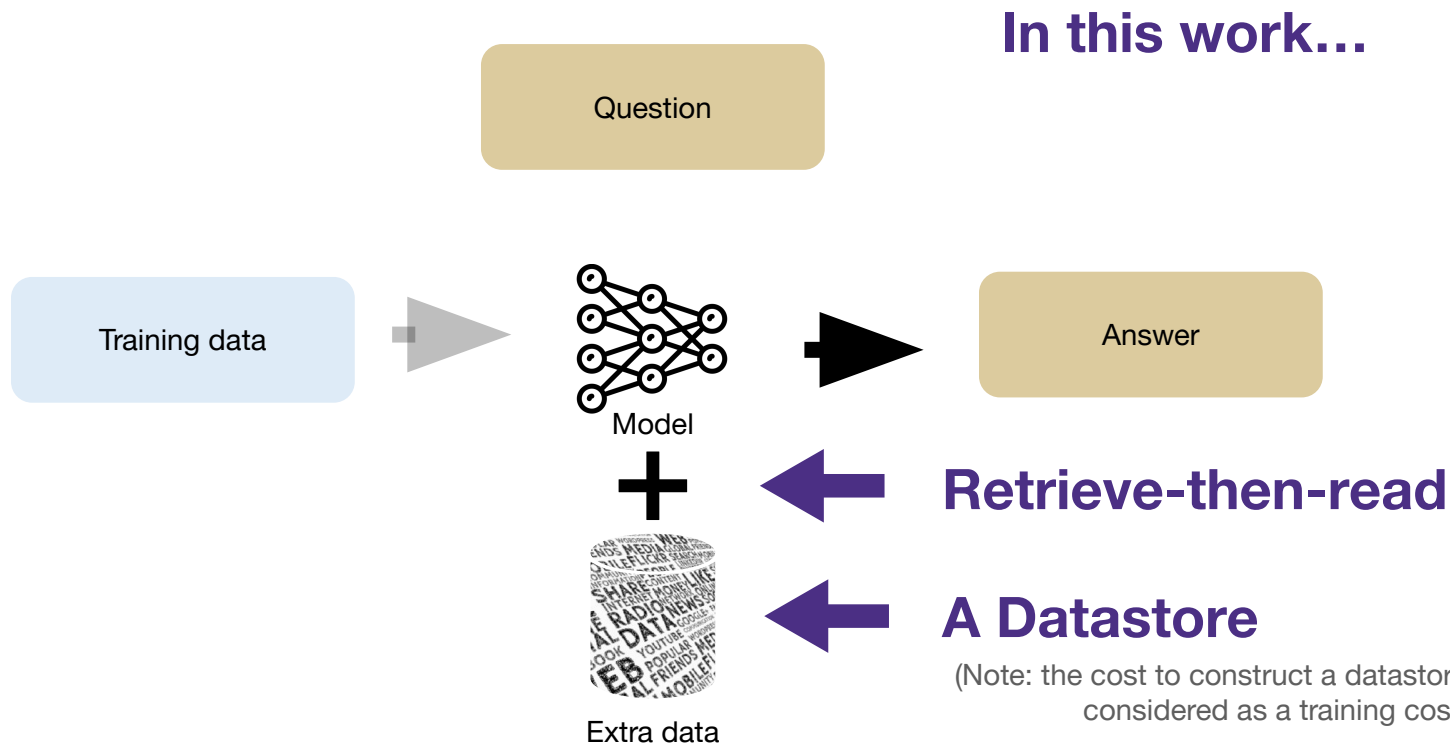
# What is test-time data

## Inference (w/ test-time data):



# What is test-time data

Inference (w/ test-time data):

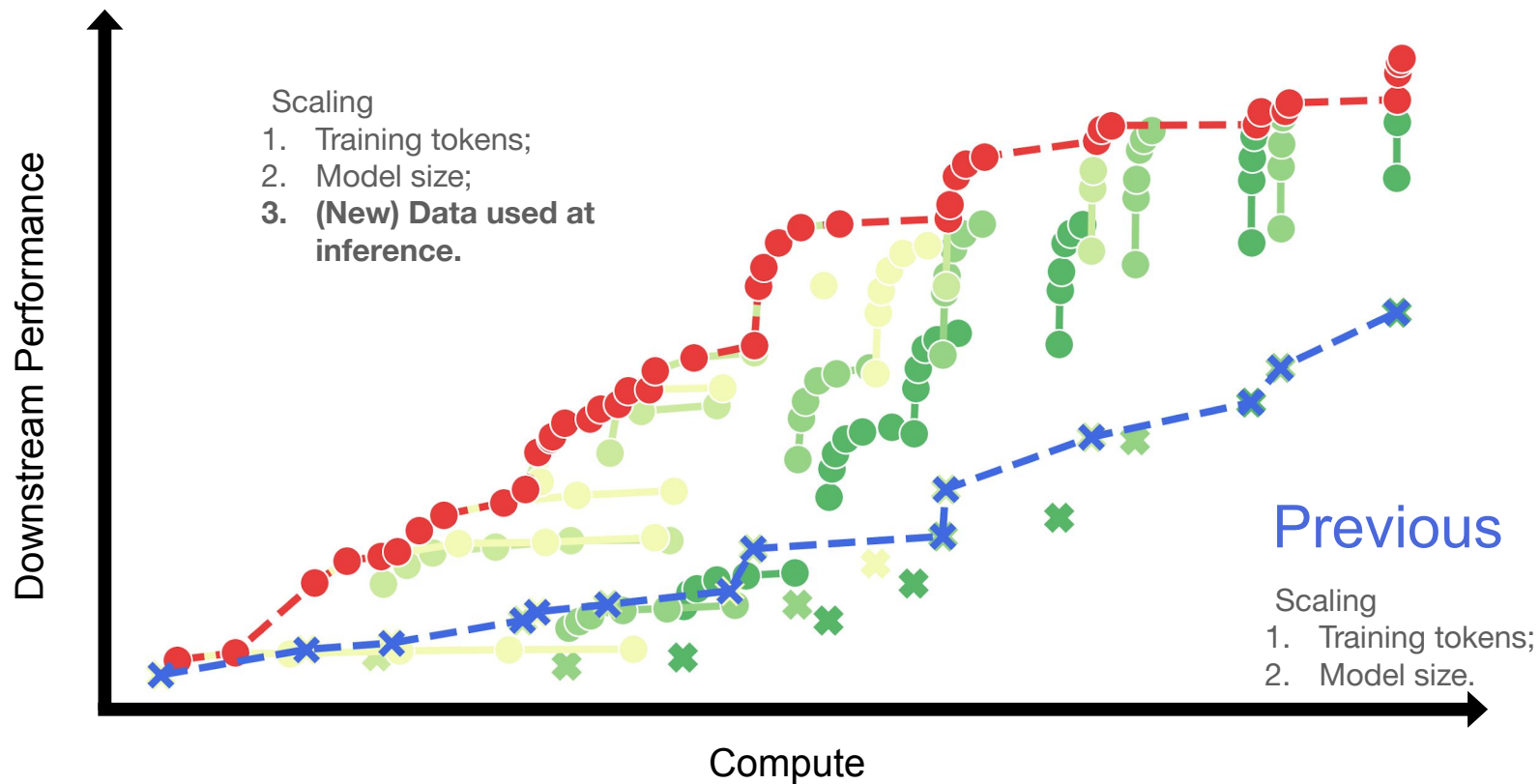


**Why scaling test-time data?**

# Why scaling test-time data?

- Standard scaling dimensions:
  - Num. Parameters
  - Num. Training Tokens

# Why scaling test-time data?





# Prior Work

# Prior work: Question Answering with DrQA

Combined **Document Retriever** (bigram hashing, TF-IDF) + **Document Reader** (multi-layer bidirectional LSTM) to identify potential answer spans in the document

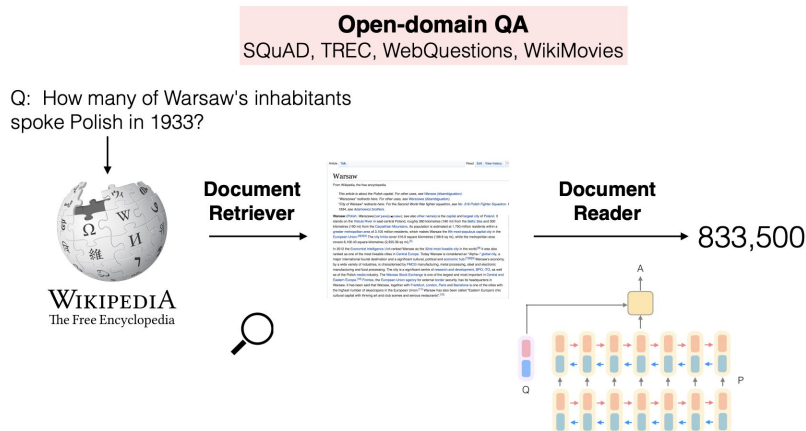


Figure 1: An overview of our question answering system DrQA.

Example	Article / Paragraph
<b>Q:</b> How many provinces did the Ottoman empire contain in the 17th century? <b>A:</b> 32	<b>Article:</b> Ottoman Empire <b>Paragraph:</b> ... At the beginning of the 17th century the empire contained 32 provinces and numerous vassal states. Some of these were later absorbed into the Ottoman Empire, while others were granted various types of autonomy during the course of centuries.
<b>Q:</b> What U.S. state's motto is "Live free or Die"? <b>A:</b> New Hampshire	<b>Article:</b> Live Free or Die <b>Paragraph:</b> "Live Free or Die" is the official motto of the U.S. state of New Hampshire, adopted by the state in 1945. It is possibly the best-known of all state mottos, partly because it conveys an assertive independence historically found in American political philosophy and partly because of its contrast to the milder sentiments found in other state mottos.
<b>Q:</b> What part of the atom did Chadwick discover? <sup>†</sup> <b>A:</b> neutron	<b>Article:</b> Atom <b>Paragraph:</b> ... The atomic mass of these isotopes varied by integer amounts, called the whole number rule. The explanation for these different isotopes awaited the discovery of the neutron, an uncharged particle with a mass similar to the proton, by the physicist James Chadwick in 1932. ...

# Prior work: REALM (Retrieval Augmented LM Pretraining)

Jointly learn to **retrieve + read** relevant documents from a large corpus during pre-training → dynamically access external knowledge instead of storing in parameters.

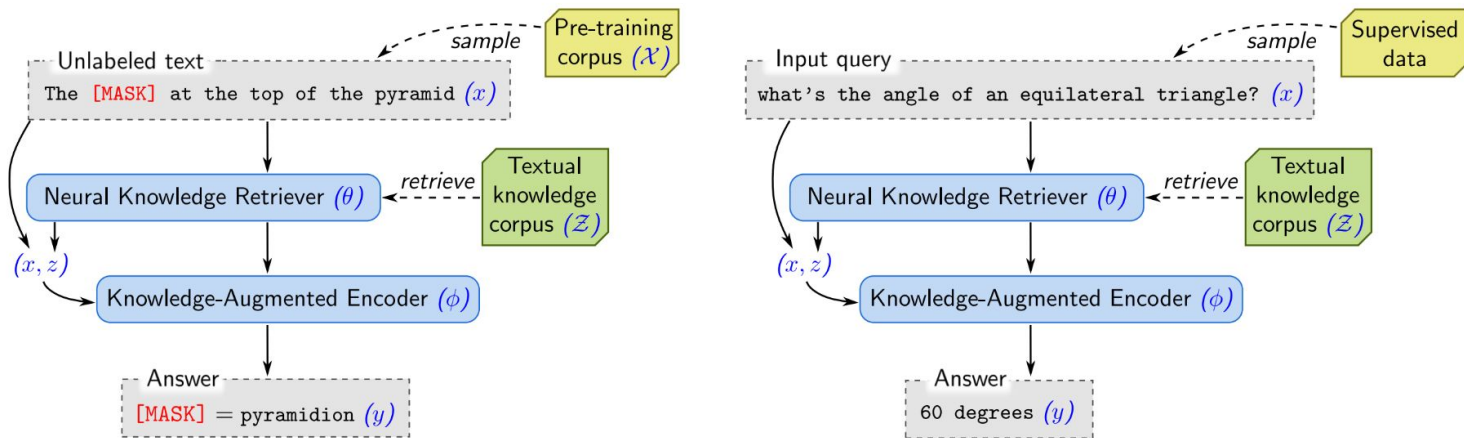


Figure 2. The overall framework of REALM. **Left:** *Unsupervised pre-training*. The knowledge retriever and knowledge-augmented encoder are jointly pre-trained on the unsupervised language modeling task. **Right:** *Supervised fine-tuning*. After the parameters of the retriever ( $\theta$ ) and encoder ( $\phi$ ) have been pre-trained, they are then fine-tuned on a task of primary interest, using supervised examples.

# Prior work: RAG (Retrieval Augmented Generation)

Combine **pre-trained neural retriever + sequence-to-sequence generator** so the model can fetch relevant documents

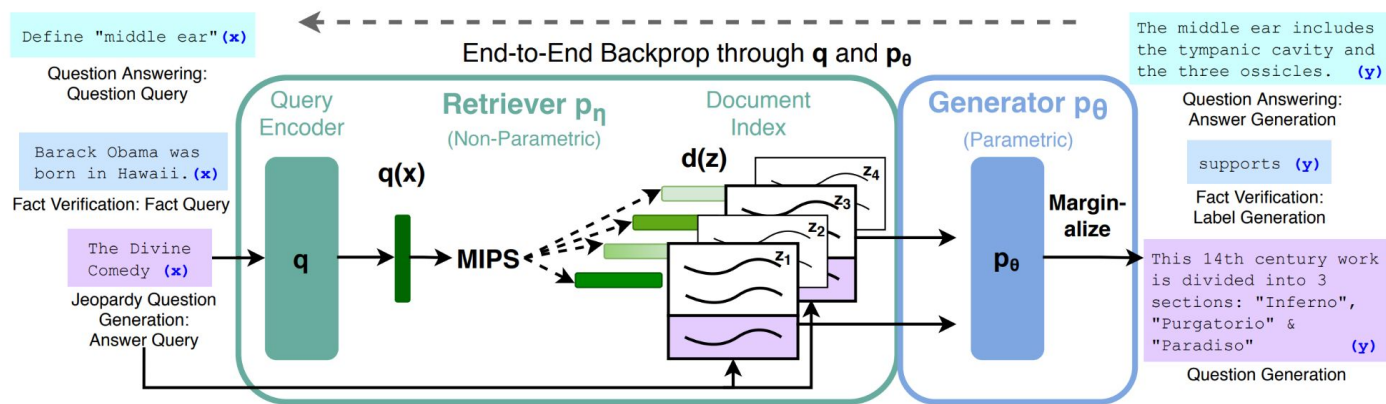
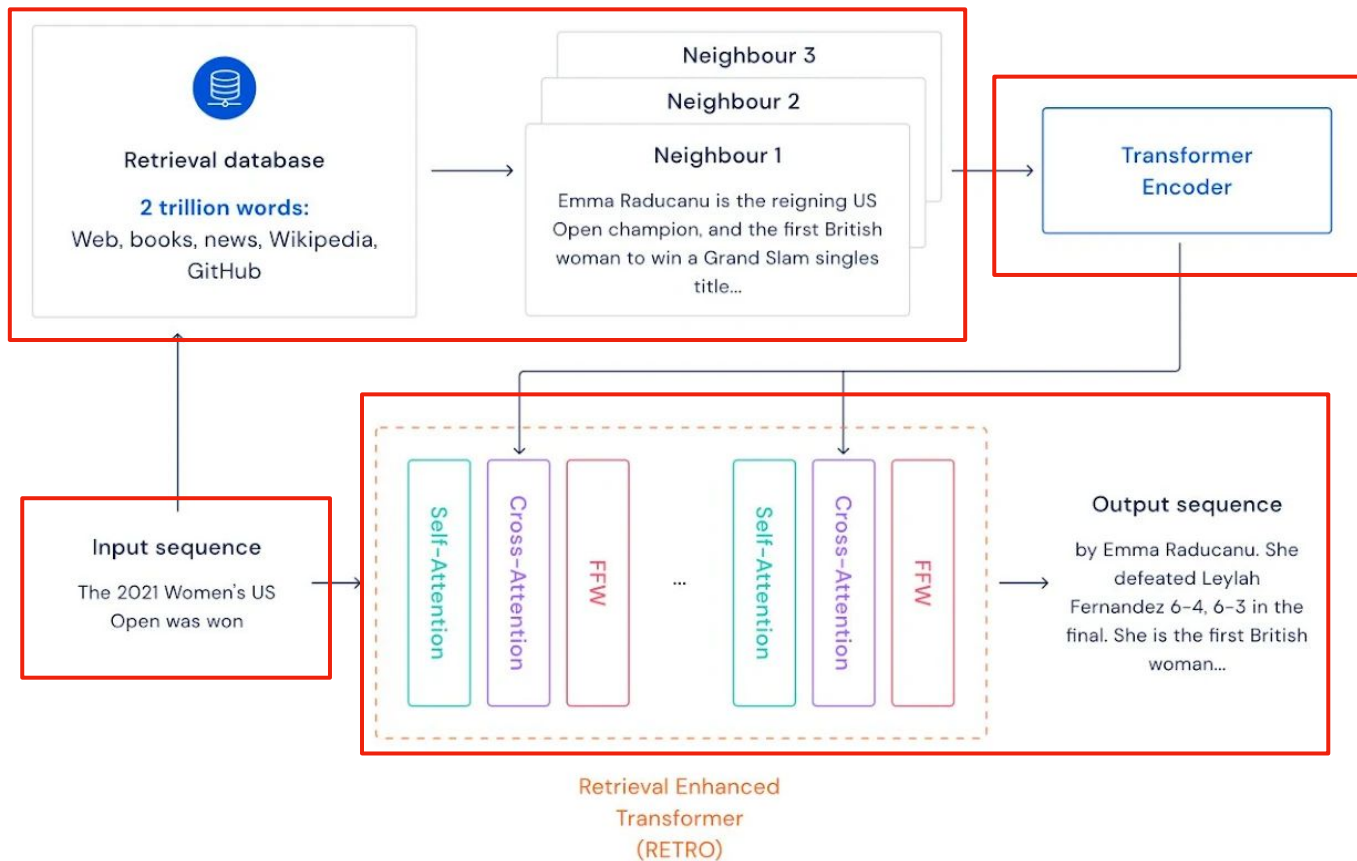


Figure 1: Overview of our approach. We combine a pre-trained retriever (*Query Encoder + Document Index*) with a pre-trained seq2seq model (*Generator*) and fine-tune end-to-end. For query  $x$ , we use Maximum Inner Product Search (MIPS) to find the top-K documents  $z_i$ . For final prediction  $y$ , we treat  $z$  as a latent variable and marginalize over seq2seq predictions given different documents.

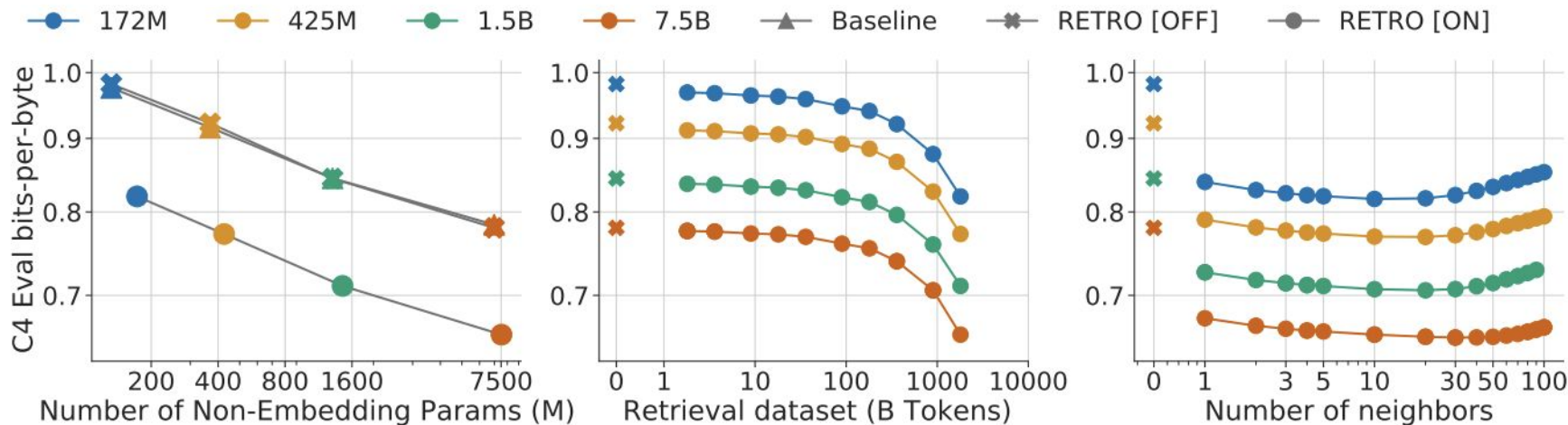
# Improving Language Models by Retrieving from Trillions of Tokens

**Presenter: Bhavya Chopra**

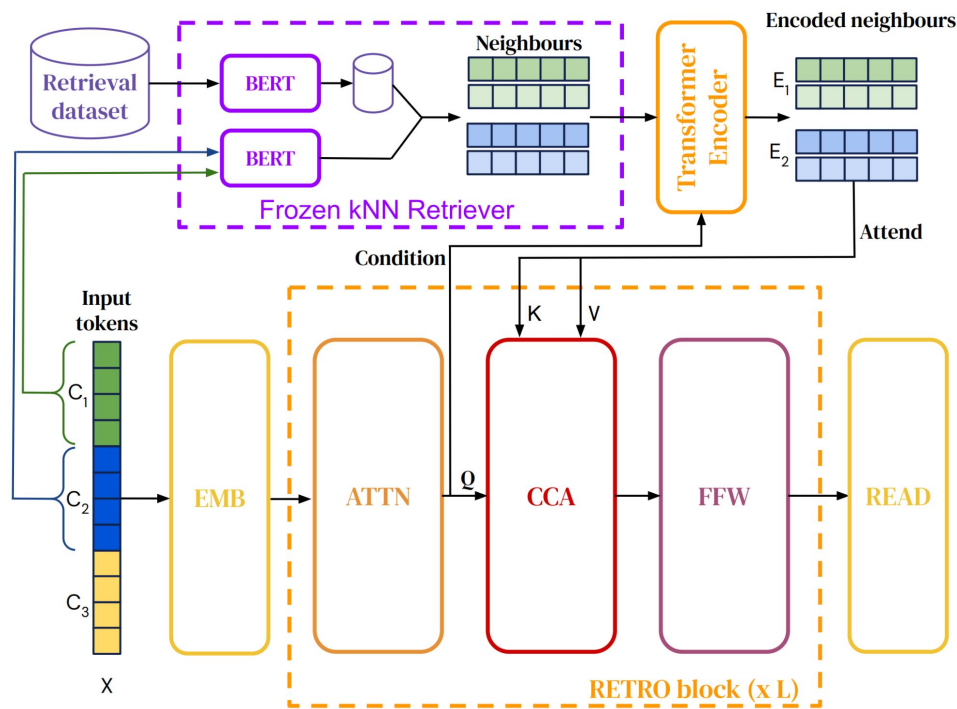
# Overview of RETRO



# Overview of RETRO



# Components of RETRO



**Chunking:** Split input text into chunks of 64 tokens

**Retriever:** Frozen BERT Encoder to embed chunks + retrieve nearest neighbors

**Encoder:** Re-encode retrieved chunks

**Cross-Attention:** LM attends to retrieved representations



# RETRO: Training Dataset

*MassiveText* (multilingual):

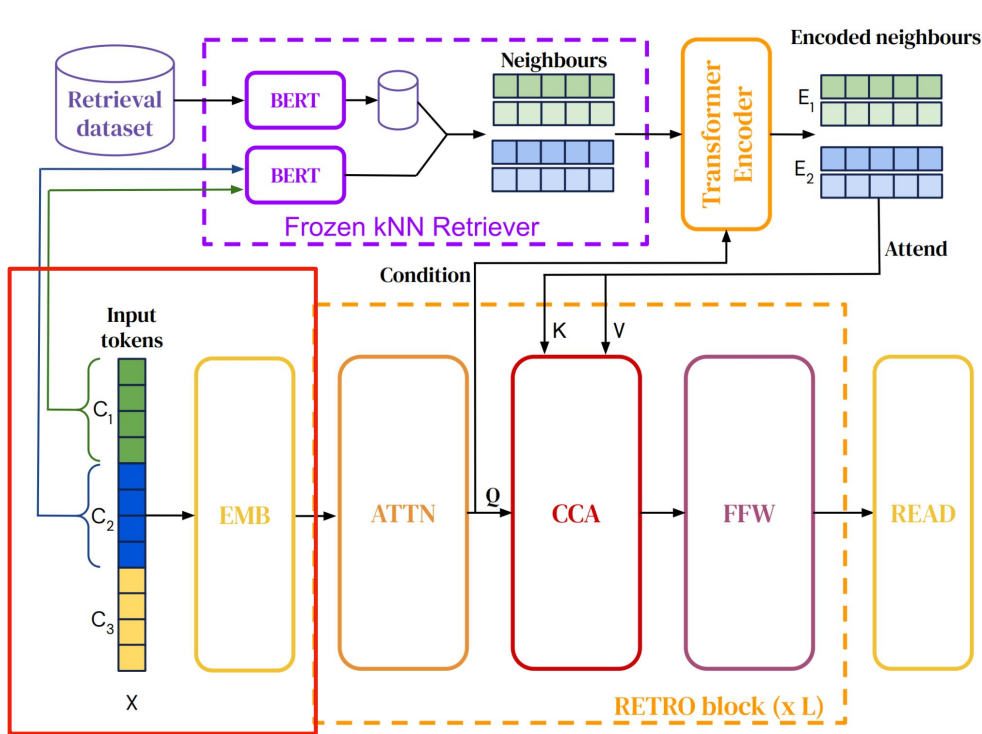
- 5 trillion tokens
- Web 55%, Books 25%, News 10% ...
- Vocabulary: 128K tokens

Retrieval DB:

- 600B tokens (training)
- 1.75T tokens (evaluation)

Source	Token count (M)	Documents (M)	Multilingual	Sampling frequency
Web	977,563	1,208	Yes	55%
Books	3,423,740	20	No	25%
News	236,918	398	No	10%
Wikipedia	13,288	23	Yes	5%
GitHub	374,952	143	No	5%

# RETRO: Retrieval-Enhanced Autoregressive Models



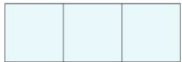
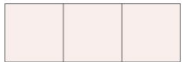
Sequence of 2048 tokens split into 64 token chunks

**Ret( $C_u$ )**: For a chunk  $C_u$  retrieve  $k$  nearest neighbors

**Likelihood**: Autoregressive log-likelihood conditions on previous chunks and retrieved sets.

**Sampling**: Retrieval via SCaNN at each chunk; preserves beam/sample efficiency

# RETRO: Nearest Neighbor Retrieval

Key (BERT sentence embedding)	Value (text. neighbor and completion chunks. Each up to 64 tokens in length)	
	Dune is a 2021 American epic science fiction film directed by Denis Villeneuve	NEIGHBOR
	It is the first of a planned two-part adaptation of the 1965 novel by Frank Herbert	COMPLETION
	Dune is a 1965 science fiction novel by American author Frank Herbert	NEIGHBOR
	originally published as two separate serials in Analog magazine	COMPLETION
...	...	

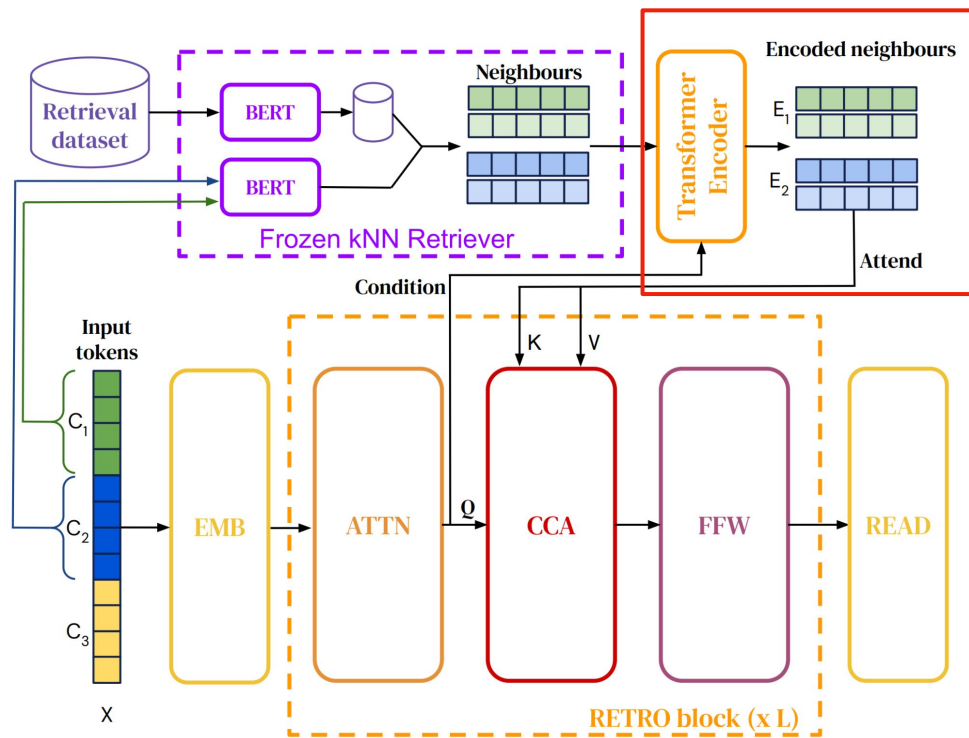
**Memory** is a key-value DB:

- Keys: Frozen BERT embeddings of neighbor chunks
- Values: [N, F] pairs (neighbor, continuation)

**Search:** Approx kNN via SCaNN in  $O(\log(T))$ : 10ms on 2T tokens

**Output:** Neighbor and its continuation (length 64 tokens)

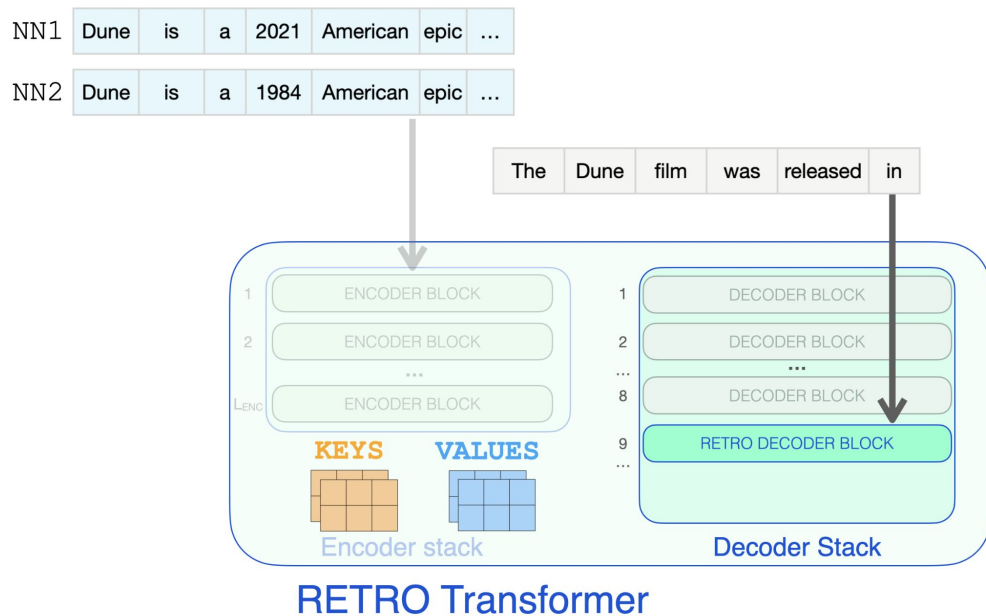
# RETRO: Model Architecture (1 of 2)



## The Retrieval Encoder:

- Bidirectional transformer
- Processes retrieved neighbor to create encoded neighbor ( $E$ )
- Conditioned on main model's activations

# RETRO: Model Architecture (2 of 2)



## Standard Transformer block:

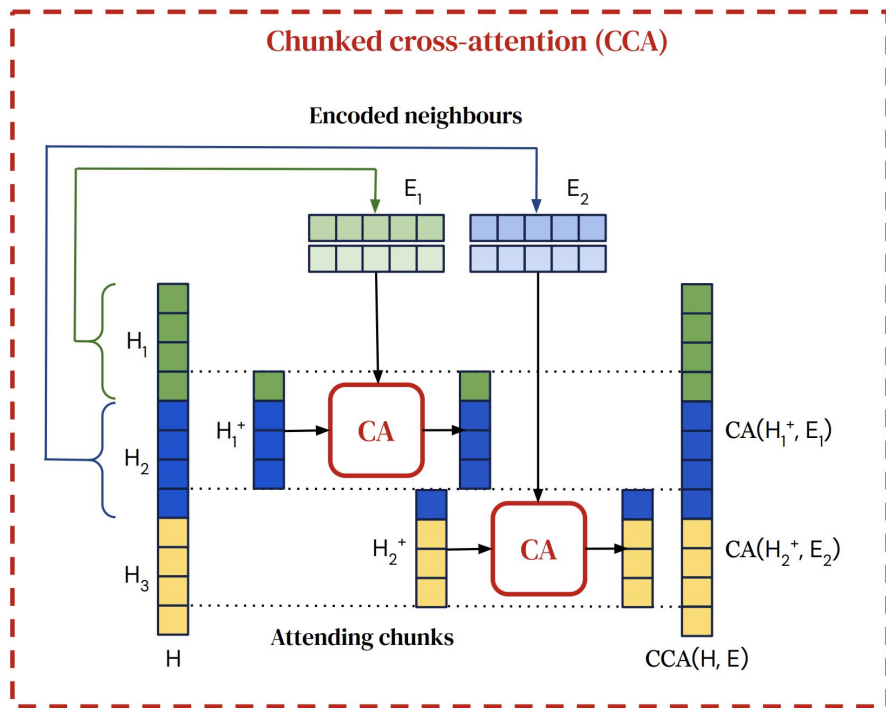
Input → Self-Attention → Feed-Forward → Output

## RETRO-block:

Input → Self-Attention → Chunked  
Cross-Attention (CCA) → Feed-Forward → Output

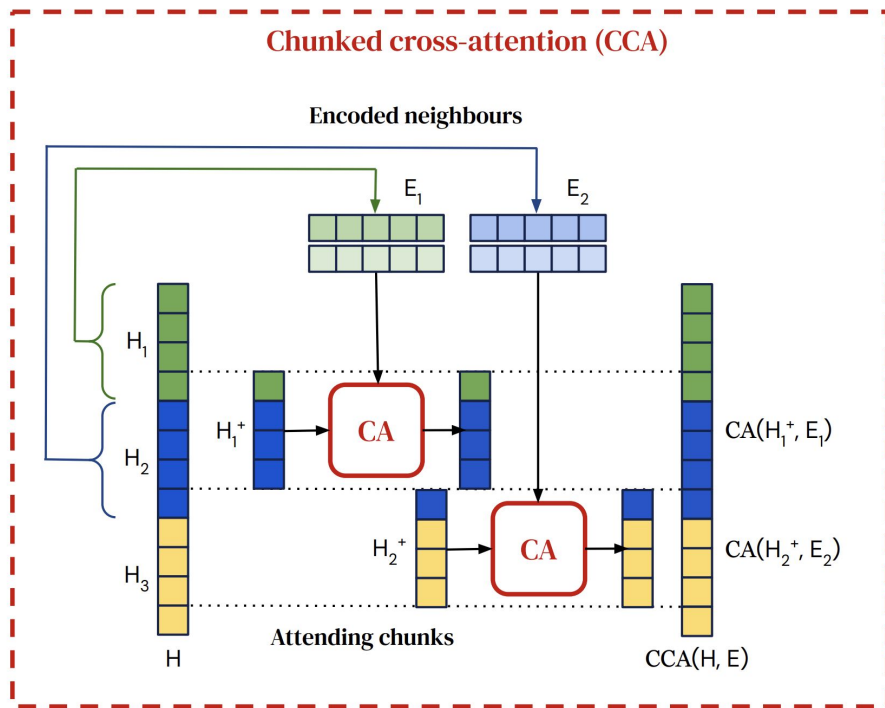
Every third block (starting from block 6 in smaller models) is a RETRO block: 9, 12, 15, ...32

# RETRO: Chunked Cross Attention



- Split activations into “attending chunks” (contain last token of 1 chunk, 63 of next chunk)
- Each attending chunk looks at (cross-attends to) encoded neighbors of PREV chunk
- Helps maintain causality (retrieve information from chunks already seen)

# RETRO: Chunked Cross Attention



Chunk 1 (C1): [1–64]

Chunk 2 (C2): [65–128]

...

Attending Chunk 1: [64–127]  
(looks at neighbors of chunk 1 ( $E_1$ ))

Attending Chunk 2: [128–191]  
(looks at neighbors of chunk 2 ( $E_2$ ))

# Experiments: Datasets and Metrics

## Datasets:

- **C4:** Web text corpus
- **Wikitext103:** Wikipedia articles
- **Curation Corpus:** Summaries of news articles
- **Lambada:** Predict last word of passage
- **The Pile:** Diverse collection of 22 text sources
- **Wikipedia Sept 2021:** Articles written AFTER training (ensures no data leakage)

## Baseline model:

- Decoder-only model with no retrieval
- 132M, 368M, 1.3B, 7B
- Same training data, schedule, hyperparameters, optimizations

## Metrics:

- **Bits-per-byte (bpb):** Lower is better - measures how well model compresses text
- **Perplexity:** Lower is better - measures how "surprised" the model is by the text
- **Accuracy:** For Lambada, percentage of correct final word predictions

## Compare with:

- **Baseline**
- **RETRO [OFF]**
- **RETRO [ON]**



# Performance v/s Dataset Leakage

Evaluating on data that is also part of the training set:

- Is the model just memorizing?

Solution: For each eval chunk:

- find its closest training neighbors
- measure the longest common substring

This gives a "leakage score" from 0 (completely new) to 1 (exact copy).

# Performance v/s Dataset Leakage

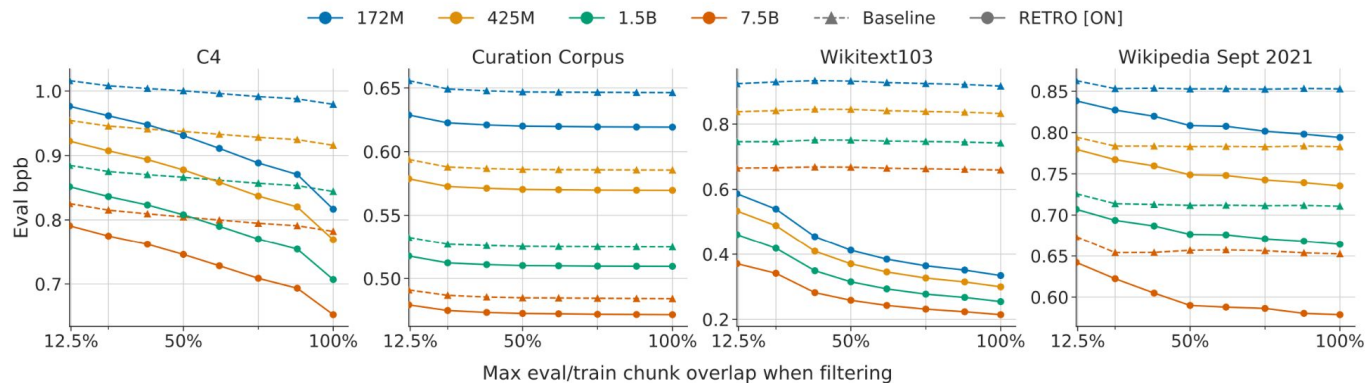


Figure 6 | **Performance vs. longest common retrieval substring.** Evaluation loss as a function of allowed longest common substring between evaluation data chunks and their nearest neighbours. Retrieval still helps when considering chunks with no more than 8 contiguous tokens overlapping with training dataset chunks.

# Experiments: Model Scaling

**Q: Does RETRO help small and large models scale equally?**

- (A) Improved perf across all model sizes**
- (B) Gains don't diminish as models get bigger**
- (C) RETRO performs at par with 10x larger baseline**

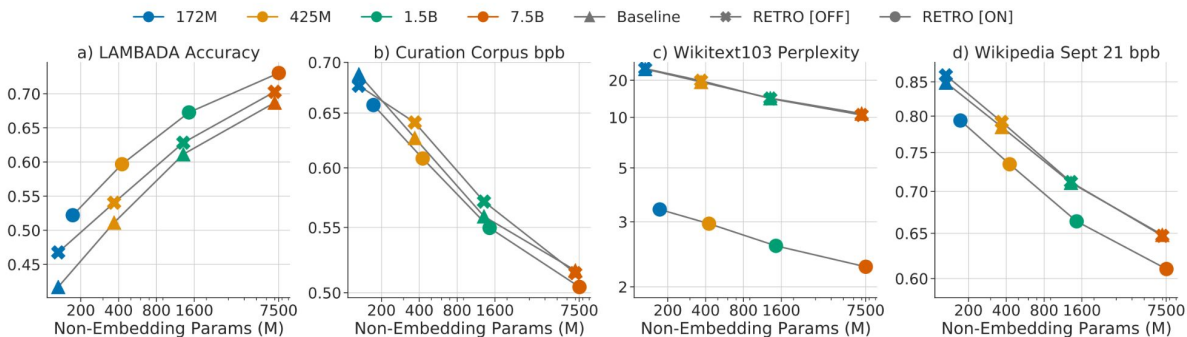


Figure 3 | **Scaling with respect to model size.** (a) LAMBADA top-1 accuracy. (b) Evaluation loss on curation corpus. (c) Perplexity on Wikitext103 valid. (d) Bits-per-byte on selected Wikipedia articles from September 2021.

# Experiments: Model Scaling

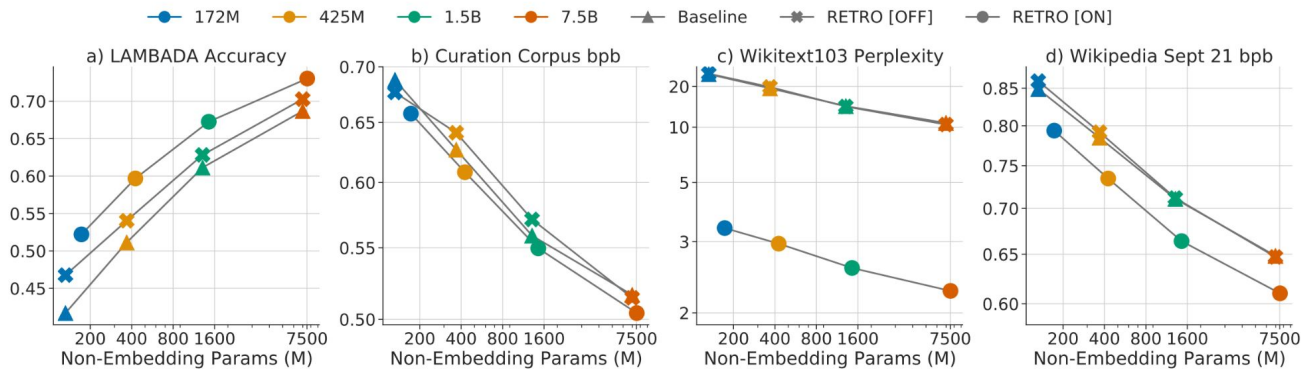


Figure 3 | **Scaling with respect to model size.** (a) LAMBADA top-1 accuracy. (b) Evaluation loss on curation corpus. (c) Perplexity on Wikitext103 valid. (d) Bits-per-byte on selected Wikipedia articles from September 2021.

Dataset-specific observations:

- Biggest gains: Wikitext103 (Wikipedia is in the retrieval DB)
- Smallest gains: Curation Corpus (news summaries not present in DB)
- Wikipedia Sept 2021: RETRO helps on articles written after training!

# Experiments: Model Scaling

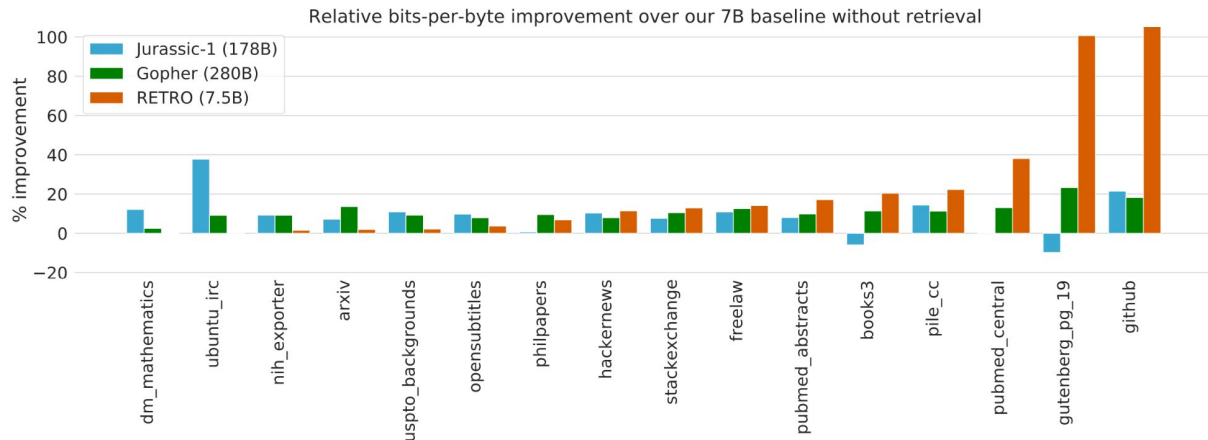


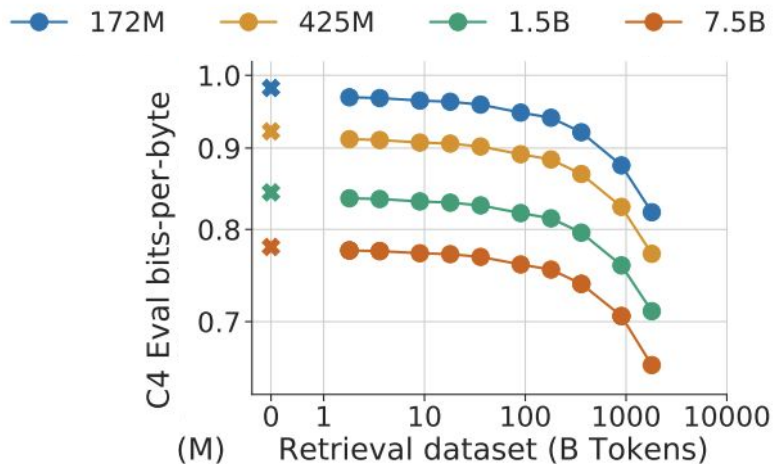
Figure 4 | **The Pile: Comparison of our 7B baseline against Jurassic-1, Gopher, and RETRO.** We observe that the retrieval model outperforms the baseline on all test sets and outperforms Jurassic-1 on a majority of them, despite being over an order of magnitude smaller.

- RETRO 7.5B **outperforms Jurassic-1** on most test sets despite being 25× smaller
- RETRO is competitive with Gopher (280B) on many datasets
- **Fails on:** dm\_mathematics and ubuntu\_irc (retrieved neighbors probably not helpful)

# Experiments: Data Scaling

**Q: Does having more data in the retrieval DB help?**

**Yes! Performance dramatically improves as DB size increases** from 4B (Wikipedia only) to 1.75T (MassiveText)



# Experiments: Retro-fitting Models

1. Take pre-trained transformer
2. Freeze all weights
3. Add new components:
  - a. Retrieval encoder
  - b. chunked cross-attention layers
4. Train only new weights  
( $< 10\%$  of total parameters)
5. Train on 3% of original training data

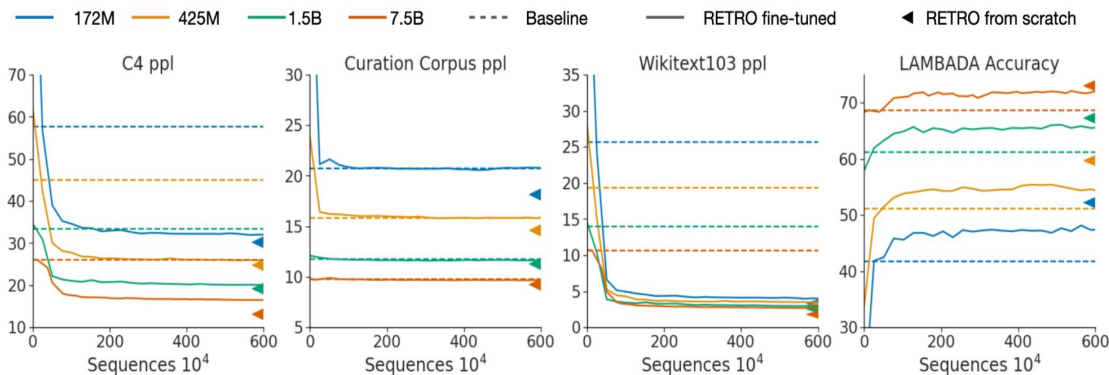


Figure 5 | **RETRO-fitting a baseline transformer.** Any transformer can be fine-tuned into a retrieval-enhanced transformer by randomly initializing and training only the chunked cross-attention and retrieval encoder weights. Fine-tuning in this way quickly recovers and surpasses the non-retrieval performance, and almost achieves the same performance as training a retrieval model from scratch (shown by the arrow on the right hand side of each plot). We find good performance RETRO-fitting our models training on only 3% the number of tokens seen during pre-training.

# Experiments: Question Answering

- RETRO is competitive with methods like DPR, RAG
- RETRO underperforms FiD (which uses T5 encoder-decoder architecture). Why?
  - RETRO uses frozen BERT retriever; not learning which documents are most useful for task
  - FiD: retrieval is jointly trained to align better with QA task
  - RETRO's retrieved neighbors may be semantically related but not always directly useful for answering a specific question

Model	Test Accuracy
REALM (Gua et al., 2020)	40.4
DPR (Karpukhin et al., 2020)	41.5
RAG (Lewis et al., 2020)	44.5
EMDR <sup>2</sup> (Sachan et al., 2021)	52.5
FiD (Izacard and Grave, 2021)	51.4
FiD + Distill. (Izacard et al., 2020)	<b>54.7</b>
Baseline 7B (closed book)	30.4
RETRO 7.5B (DPR retrieval)	45.5



# RETRO: Summary

## Input prompt

Pi=3.14159265358979323846264338327950288419716939937510582097494459230781  
64062862089986280348253421170679

## Baseline 7.1B sample

829404960289884960699858349

## RETRO 7.5B sample

82148086513282306647093844609

## Input prompt

Beavers are interesting animals that live near rivers. They build

## Baseline 7.1B sample

dams to create ponds. Frogs are amphibians so they can live in both land and water. They have great camouflage to hide from predators. The Golden Retriever, or Golden, is a very big dog...

## RETRO 7.5B sample

their houses called beaver dams in the riverbeds. They also live on land. Beavers use their strong teeth and strong jaws to cut down trees and branches to build their homes. They also use their teeth and jaws to chew up food.

# Scaling Retrieval-Based Language Models with a Trillion-Token Datastore

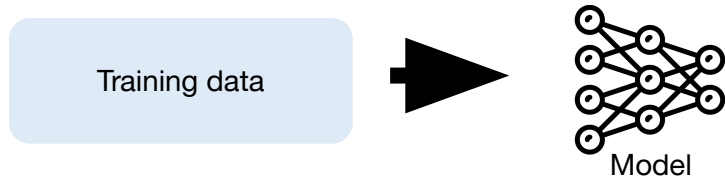
**Presenter: Yichuan Wang**

Credit to Rulin for providing the draft slides

# What is test-time data?

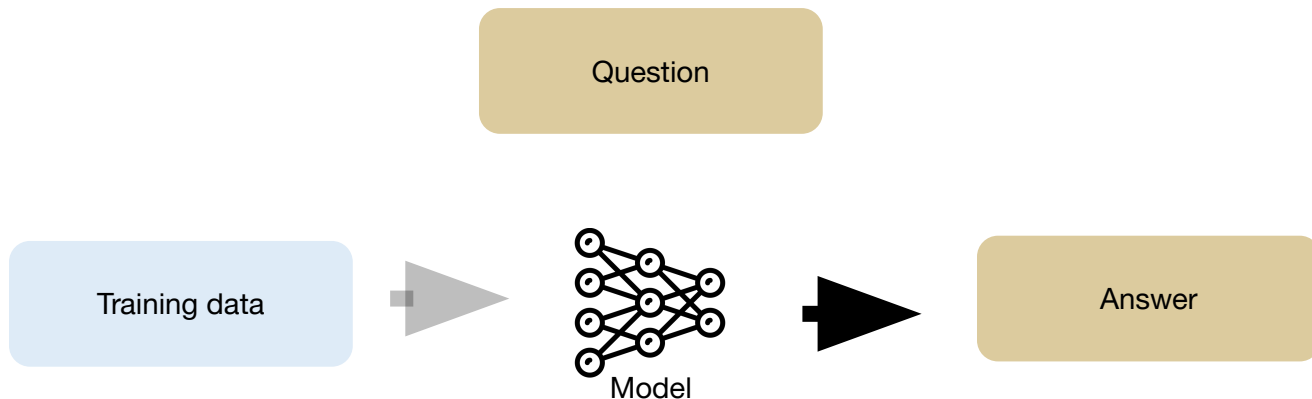
# What is test-time data?

Training:



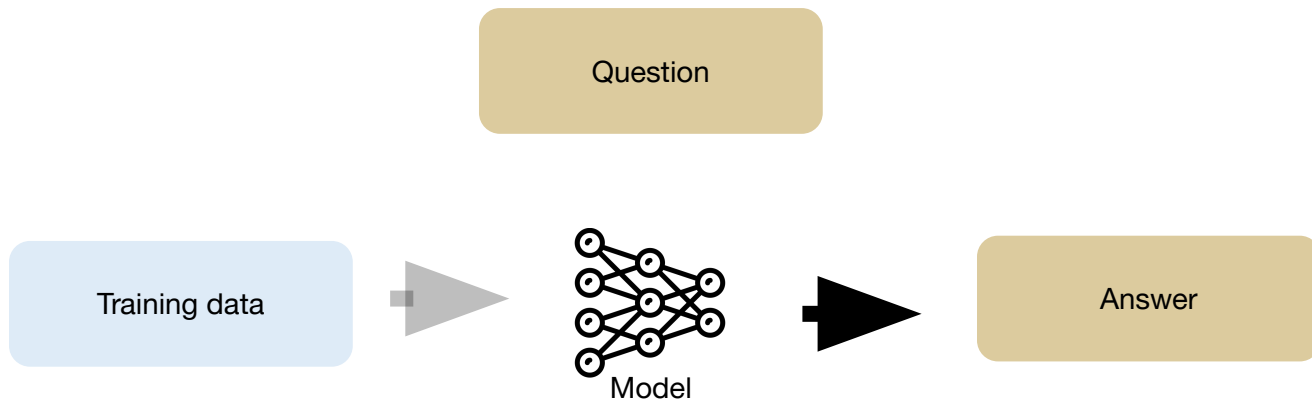
# What is test-time data?

Inference (w/o test-time data):



# What is test-time data?

Inference (w/o test-time data):

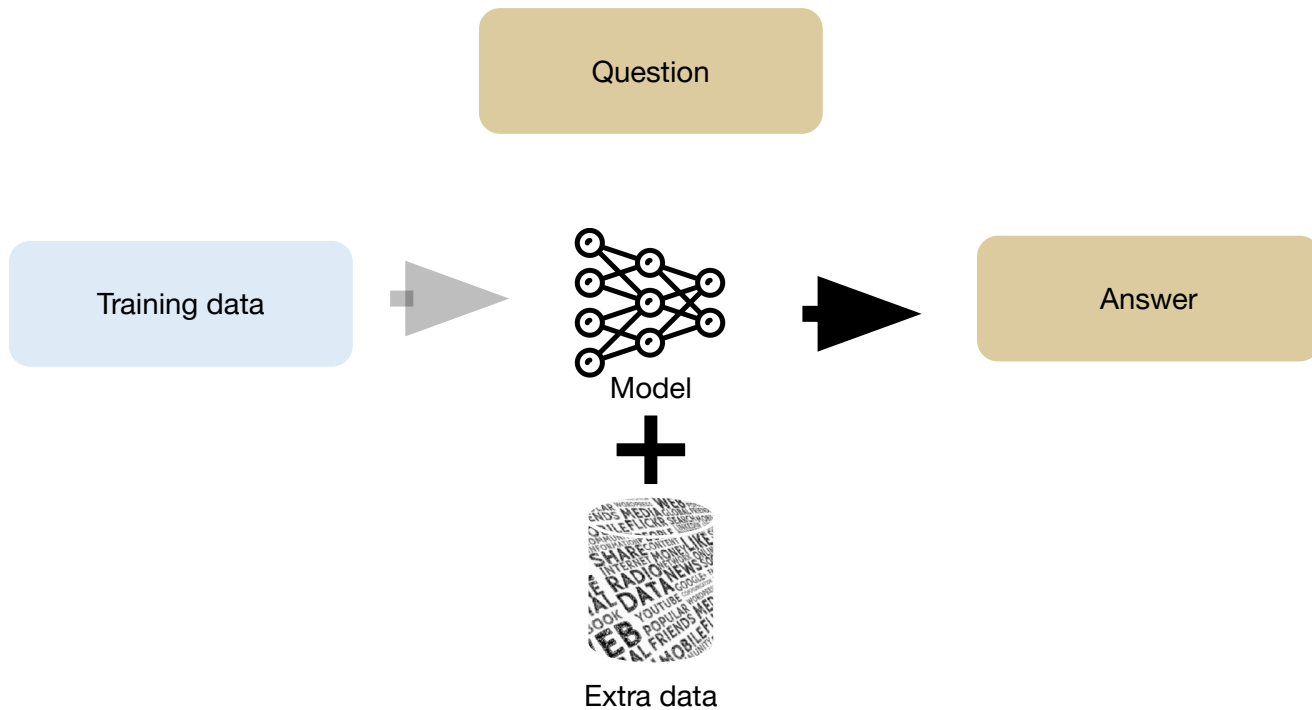


## Limitations:

1. Difficult to attribute/verify the data source;
2. The model can hallucinate a lot on long-tail knowledge;
3. Difficult to update with new knowledge.

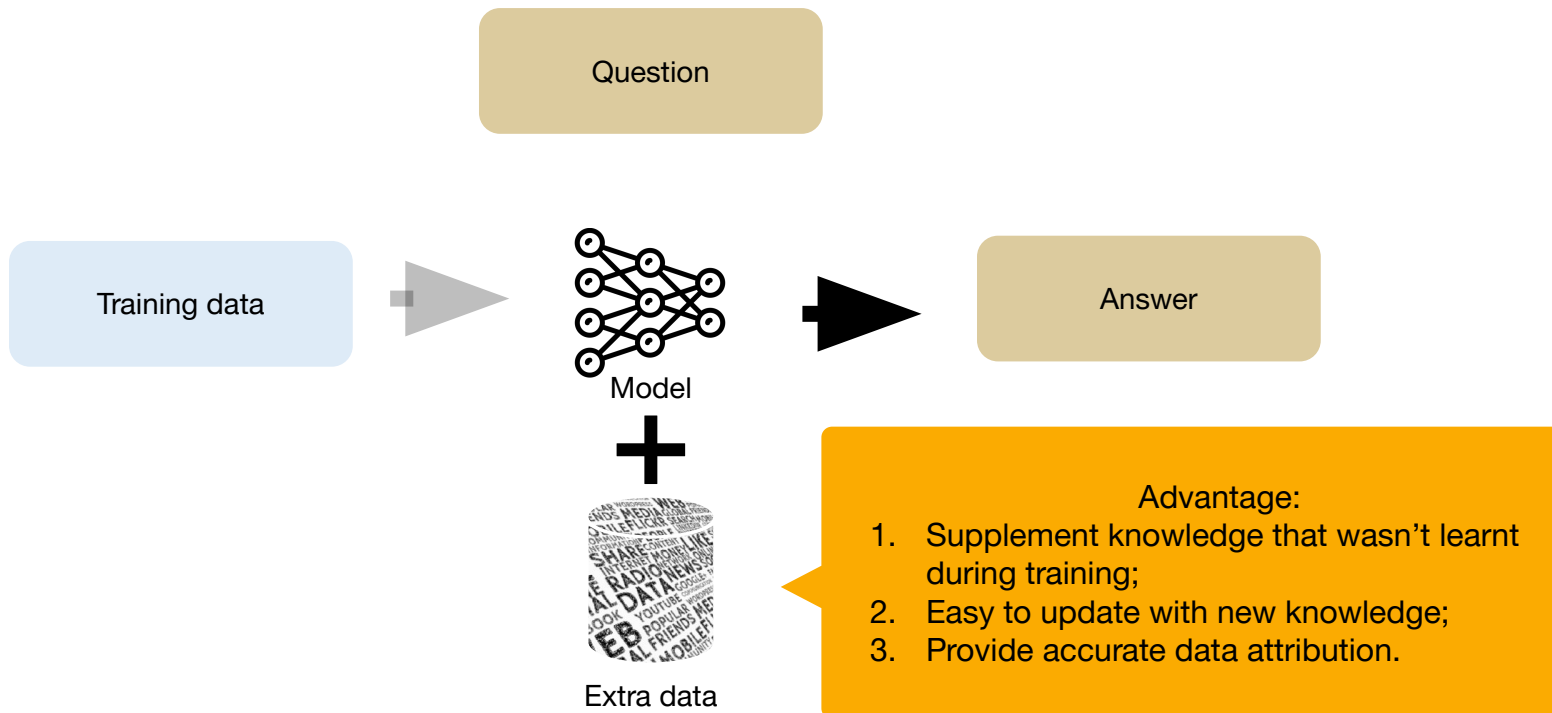
# What is test-time data?

Inference (w/ test-time data):



# What is test-time data?

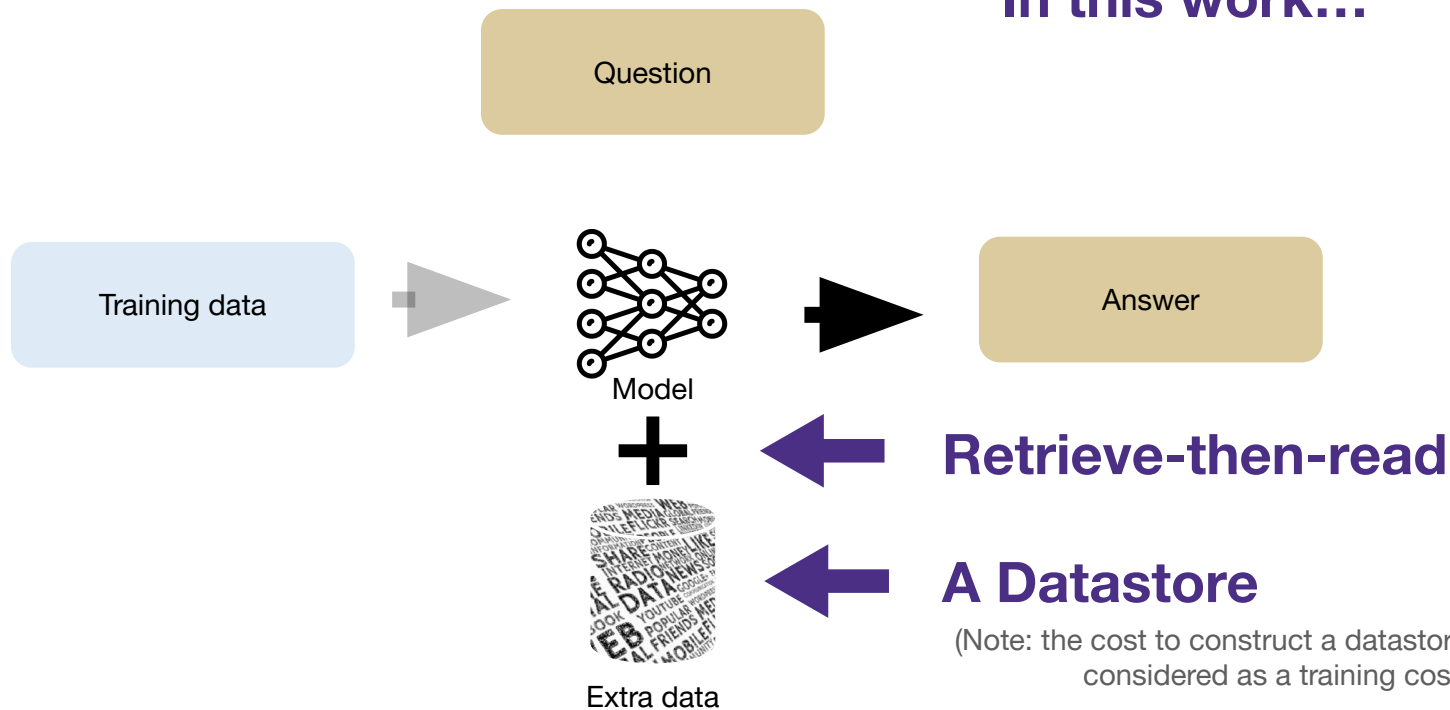
## Inference (w/ test-time data):





# What is test-time data?

Inference (w/ test-time data):

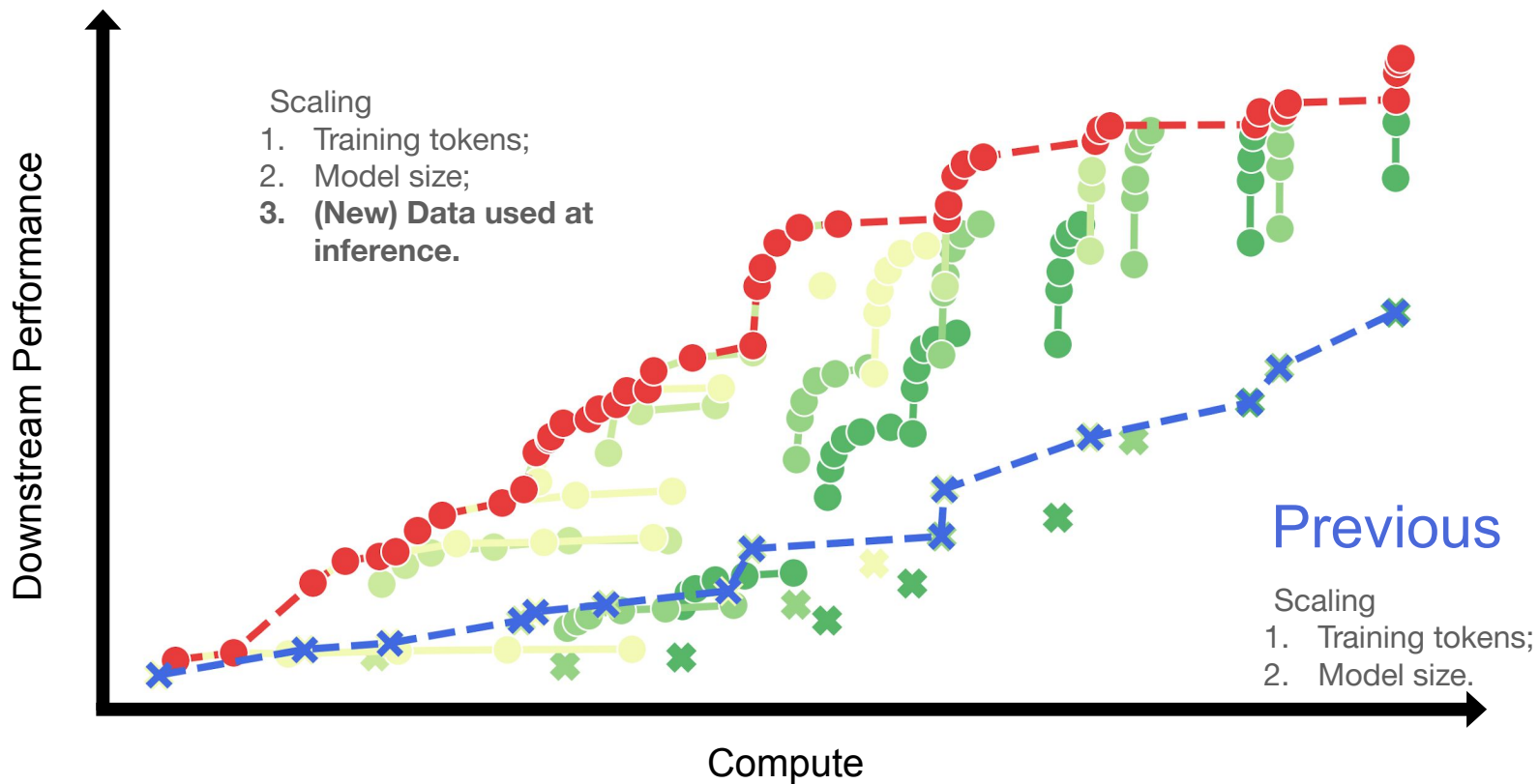


# Why scaling test-time data?

# Why scaling test-time data?

- Standard scaling dimensions:
  - Num. Parameters
  - Num. Training Tokens

# Why scaling test-time data?



# Scaling retrieval-based Language Models

## MassiveDS: a 1.4 trillion-token datastore

- Prior work on retrieval-based models focus on small-scale datastores

Reference	# Tokens	Data Sources	Open sourced
ATLAS (Izacard et al., 2023)	<5B	Wikipedia	✗
REALM (Guu et al., 2020)	<5B	Wikipedia	✗
RALM (Ram et al., 2023)	<5B	Wikipedia	✓
SELF-RAG (Asai et al., 2024a)	<5B	Wikipedia	✓
REPLUG (Shi et al., 2023)	47B	The Pile	✓
RA-DIT (Lin et al., 2024)	79B	Wikipedia, CommonCrawl	✗
SPHERE (Piktus et al., 2022)	90B	CCNet	✓

# Scaling retrieval-based Language Models

## MassiveDS: a 1.4 trillion-token datastore

- RETRO only studied datastore scaling performance on perplexity; the datastore wasn't open-sourced.

Reference	# Tokens	Data Sources	Open sourced
ATLAS (Izacard et al., 2023)	<5B	Wikipedia	✗
REALM (Guu et al., 2020)	<5B	Wikipedia	✗
RALM (Ram et al., 2023)	<5B	Wikipedia	✓
SELF-RAG (Asai et al., 2024a)	<5B	Wikipedia	✓
REPLUG (Shi et al., 2023)	47B	The Pile	✓
RA-DIT (Lin et al., 2024)	79B	Wikipedia, CommonCrawl	✗
SPHERE (Piktus et al., 2022)	90B	CCNet	✓
RETRO++ (Wang et al., 2024)	330B*	The Pile, CommonCrawl, RealNews, CC-Stories	✗
INSTRUCTRETRO (Wang et al., 2024)	1.2T*	Wikipedia, CommonCrawl, RealNews, CC-Stories, Books	✗
RETRO (Borgeaud et al., 2022)	1.7T*	MassiveText (Rae et al., 2022)	✗

# Scaling retrieval-based Language Models

## MassiveDS: a 1.4 trillion-token datastore

- This work: fully open-sourced; study datastore scaling on both perplexity and downstream tasks.

Reference	# Tokens	Data Sources	Open sourced
ATLAS (Izacard et al., 2023)	<5B	Wikipedia	✗
REALM (Guu et al., 2020)	<5B	Wikipedia	✗
RALM (Ram et al., 2023)	<5B	Wikipedia	✓
SELF-RAG (Asai et al., 2024a)	<5B	Wikipedia	✓
REPLUG (Shi et al., 2023)	47B	The Pile	✓
RA-DIT (Lin et al., 2024)	79B	Wikipedia, CommonCrawl	✗
SPHERE (Piktus et al., 2022)	90B	CCNet	✓
RETRO++ (Wang et al., 2024)	330B*	The Pile, CommonCrawl, RealNews, CC-Stories	✗
INSTRUCTRETRO (Wang et al., 2024)	1.2T*	Wikipedia, CommonCrawl, RealNews, CC-Stories, Books	✗
RETRO (Borgeaud et al., 2022)	1.7T*	MassiveText (Rae et al., 2022)	✗
MASSIVEDS (Ours)	<b>1.4T</b>	8 domains, listed in Table 2	✓

# Scaling retrieval-based Language Models

## MassiveDS: a 1.4 trillion-token datastore

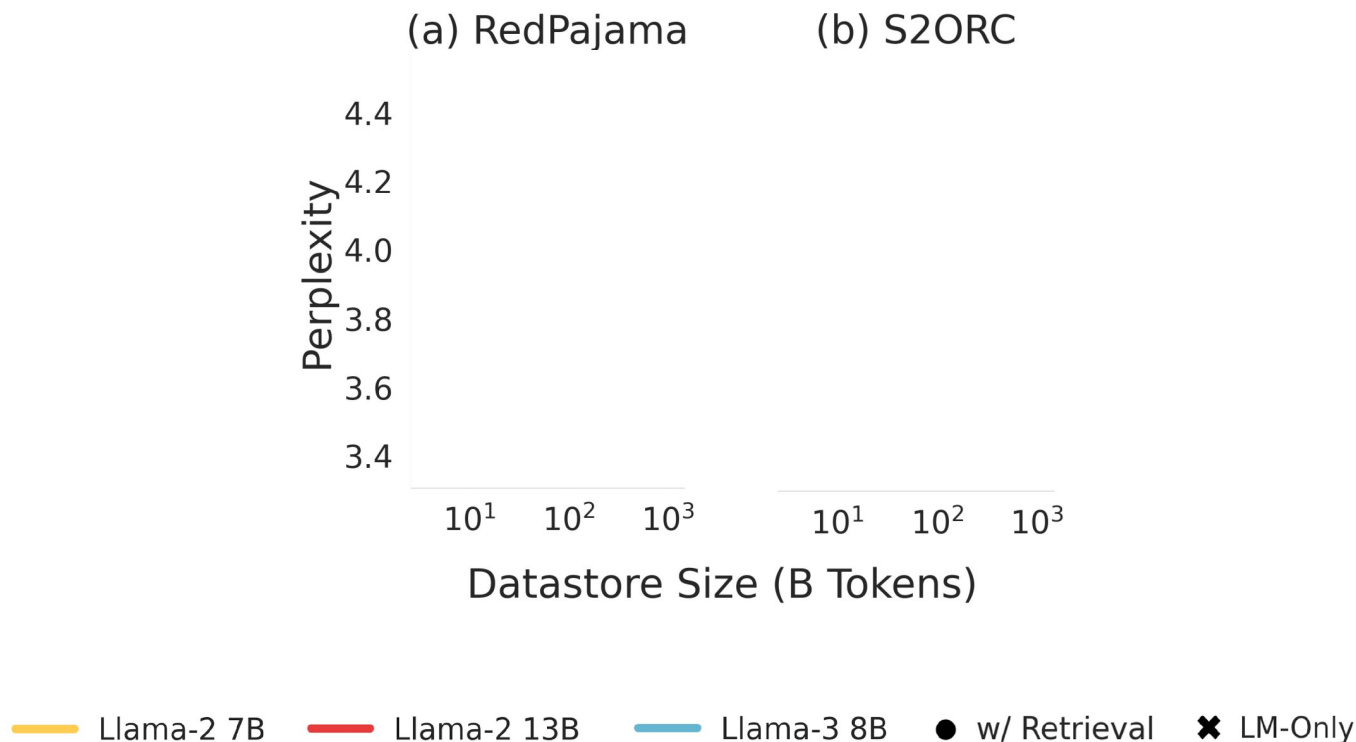
- The data composition of MassiveDS.

Domain	Datasets	Size (B)
BOOKS	RPJ Books	26.3
STEM	peS2o, RPJ ArXiv	97.7
ENCYCLOPEDIA	DPR 2018 Wiki, RPJ 2022 Wiki	31.9
FORUM (Q&A)	RPJ StackExchange	20.2
CODE	RPJ Github	52.8
MATH	OpenWebMath, NaturalProofs	14.1
BIOMEDICAL	PubMed	6.5
GENERAL WEB	RPJ CC (2019–2023), RPJ C4	1191.7
<b>Total</b>		1441.2



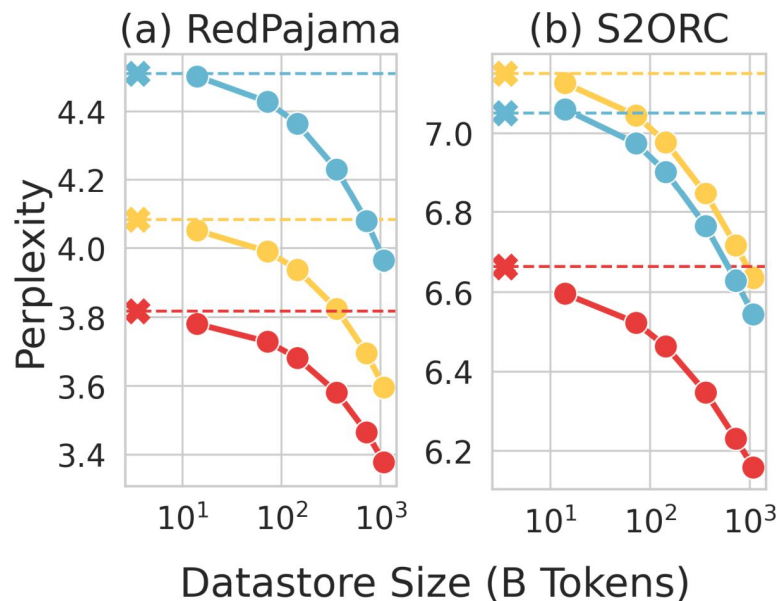
# Scaling retrieval-based Language Models

Results: datastore scaling on language modeling



# Scaling retrieval-based Language Models

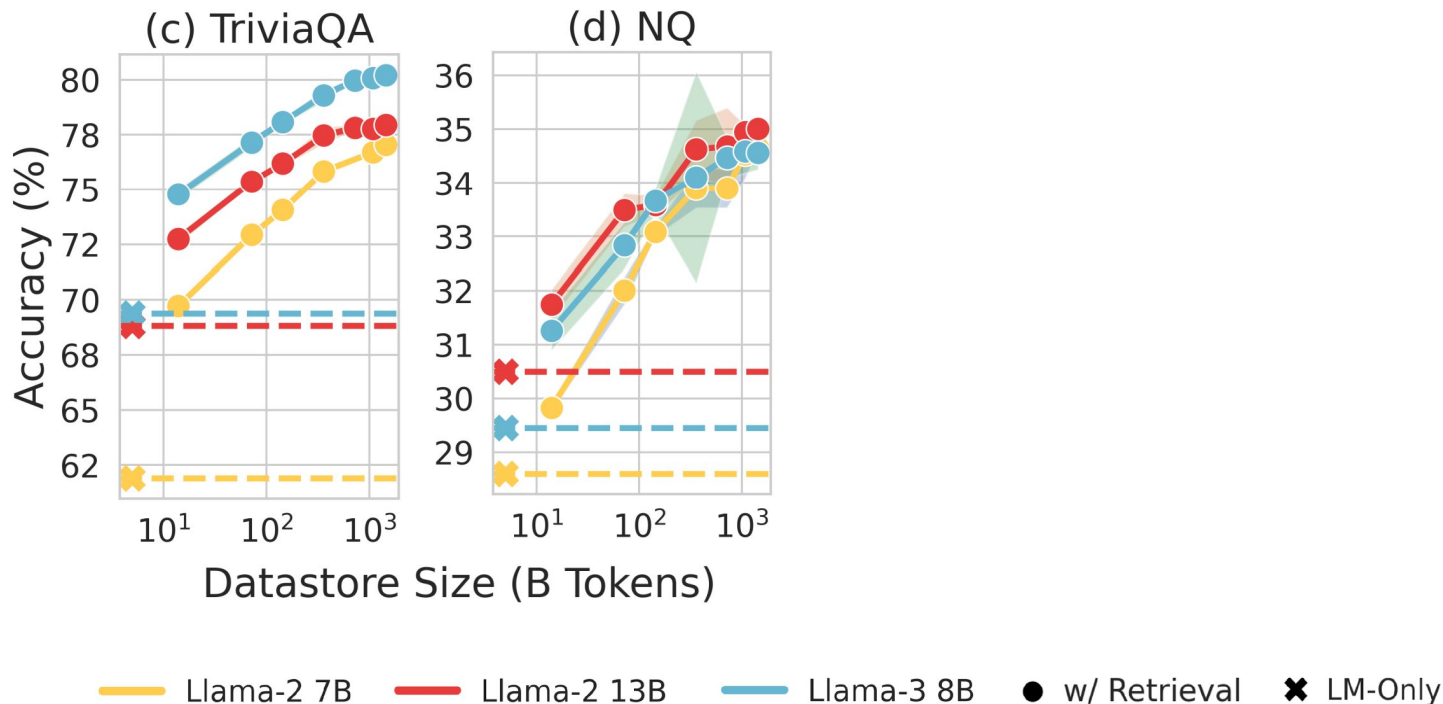
Results: datastore scaling on language modeling



— Llama-2 7B    — Llama-2 13B    — Llama-3 8B    ● w/ Retrieval    ✕ LM-Only

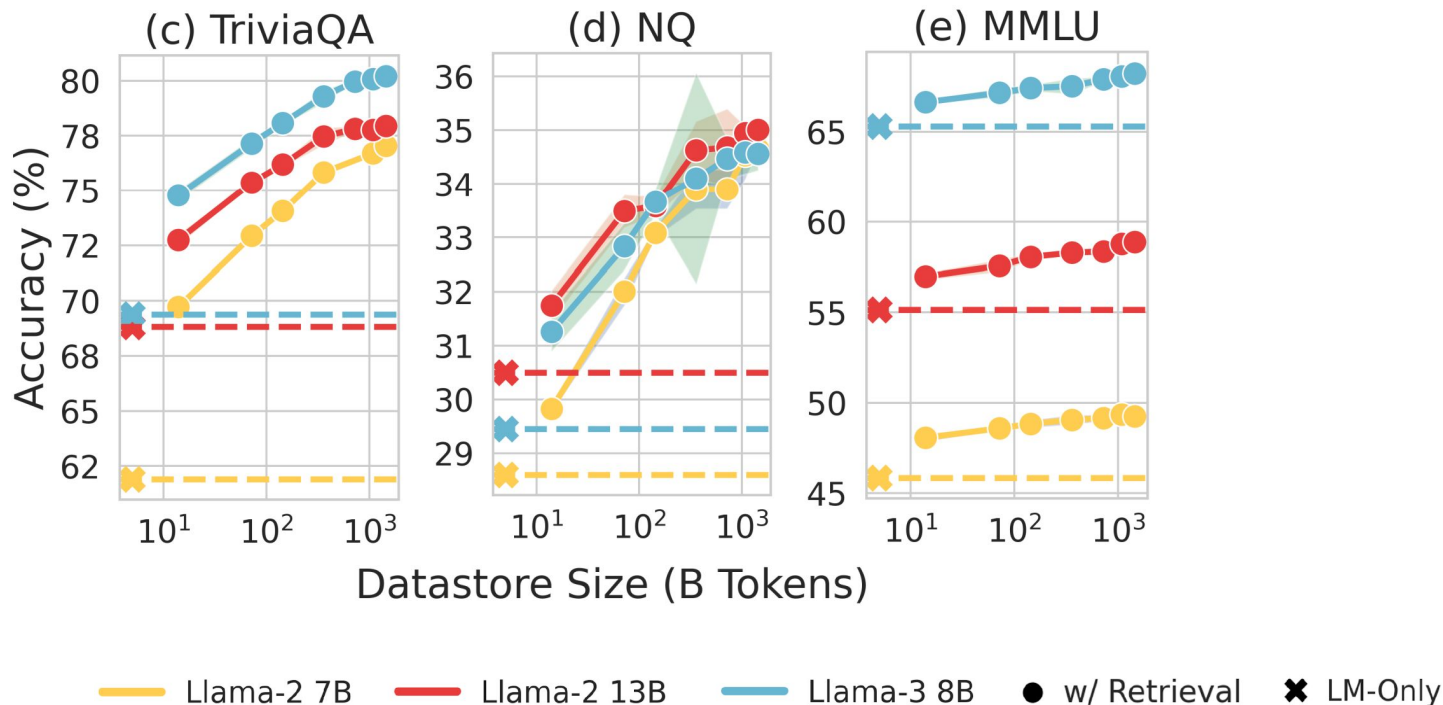
# Scaling retrieval-based Language Models

Results: datastore scaling on downstream tasks



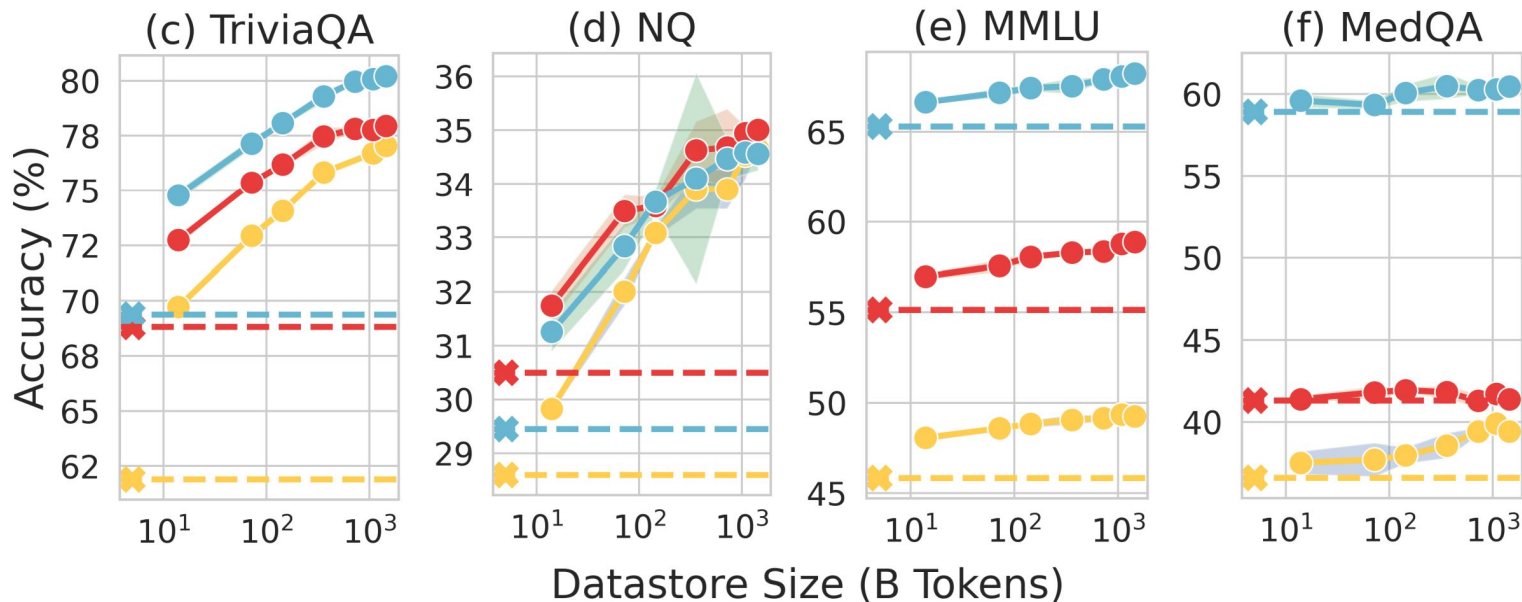
# Scaling retrieval-based Language Models

Results: datastore scaling on downstream tasks



# Scaling retrieval-based Language Models

Results: datastore scaling on downstream tasks

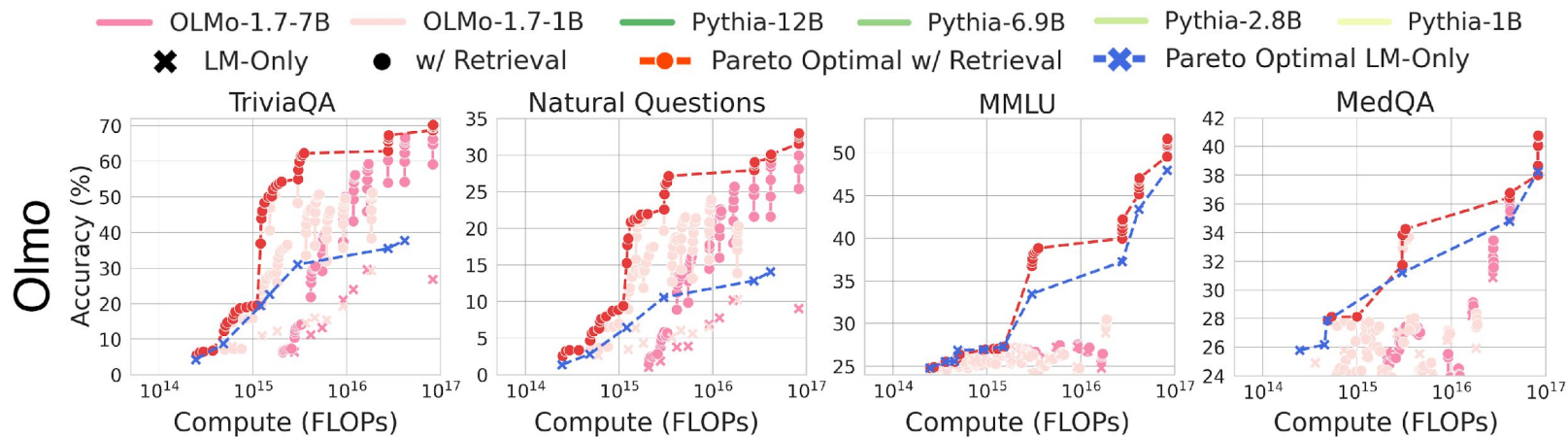


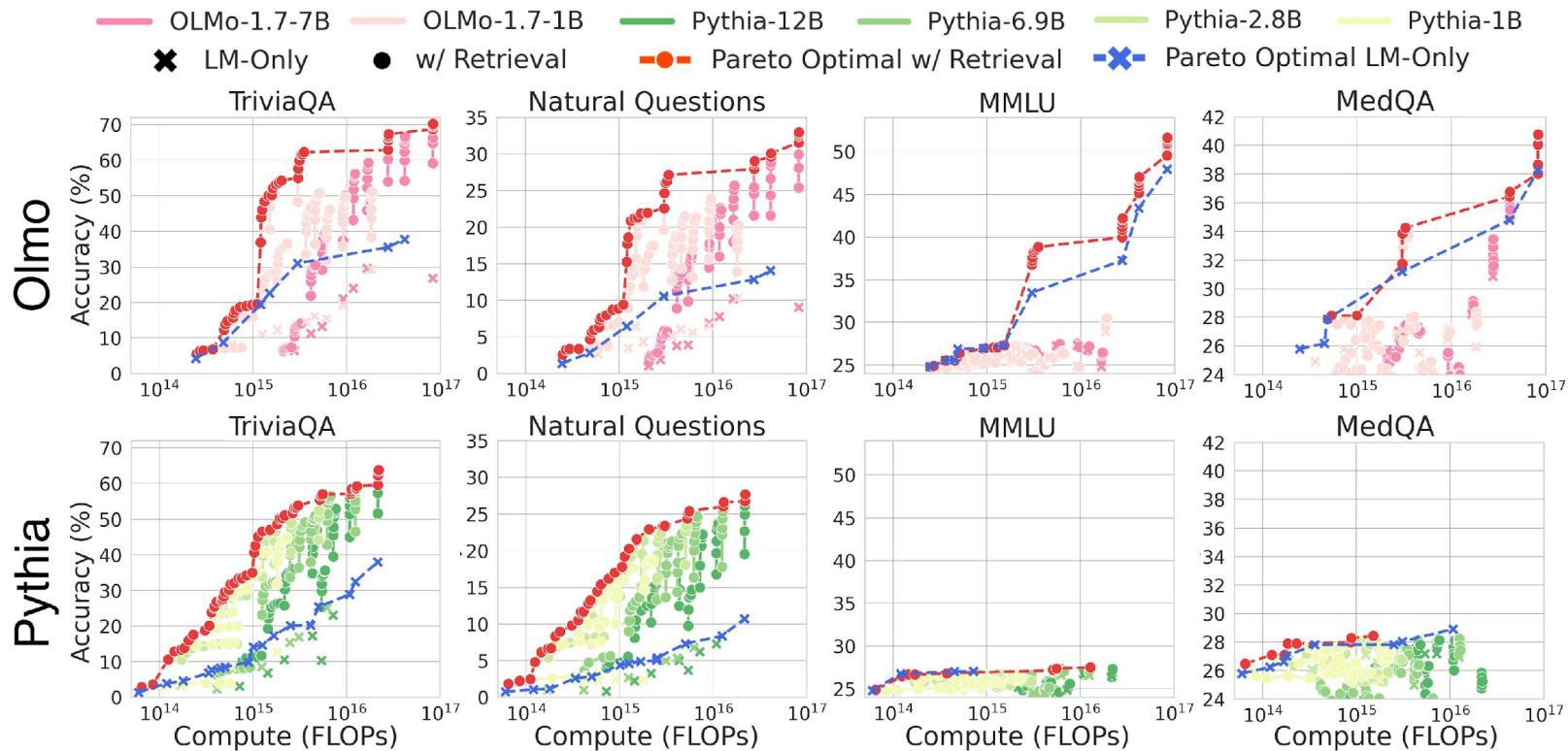
— Llama-2 7B    — Llama-2 13B    — Llama-3 8B    ● w/ Retrieval    ✕ LM-Only

# Scaling retrieval-based Language Models

**Results: compute-optimal scaling**

- Indexing the datastore takes compute at training time
- Is it worth spending FLOPs to index the datastore vs. do more pretraining?







# Scaling retrieval-based Language Models

Results: comparison with single-domain datastores

Tasks	LM-Only	PubMed	MATH	peS2o	DPR Wiki	RedPajama					MASSIVEDS
						Wiki	Books	ArXiv	SE	Github	
TQA ↑	64.1										
NQ ↑	26.6										
MedQA ↑	36.6										
MMLU ↑	45.8										
RedPajama (PPL) ↓	4.09										
S2ORC (PPL) ↓	7.18										

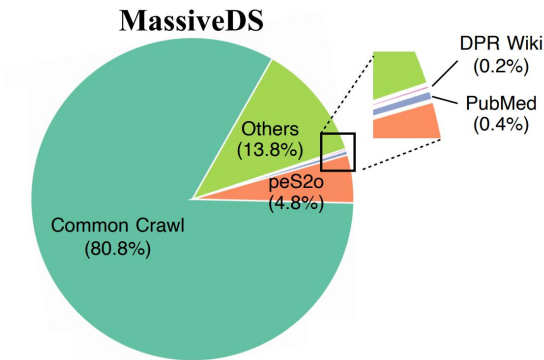
# Scaling retrieval-based Language Models

Results: comparison with single-domain datastores

Tasks	LM-Only	PubMed	MATH	peS2o	DPR Wiki	RedPajama					MASSIVEDS
						Wiki	Books	ArXiv	SE	Github	
TQA ↑	64.1	64.5	65.5	65.6	72.6	<u>72.9</u>	70.5	62.3	64.7	64.2	<b>77.0</b>
NQ ↑	26.6	26.7	26.4	26.9	<b>34.6</b>	33.8	28.0	26.4	27.0	26.4	<b>34.6</b>
MedQA ↑	36.6	37.8	36.5	38.1	38.5	38.4	<b>39.8</b>	36.9	35.4	36.1	<u>39.4</u>
MMLU ↑	45.8	46.8	47.5	47.4	48.3	48.1	<u>48.3</u>	45.6	46.2	45.9	<b>49.3</b>
RedPajama (PPL) ↓	4.09	4.06	4.08	4.08	4.06	3.99	4.01	<u>3.87</u>	4.01	3.95	<b>3.50</b>
S2ORC (PPL) ↓	7.18	7.05	7.10	6.71	7.08	7.11	7.14	<u>6.64</u>	7.08	7.11	<b>6.57</b>

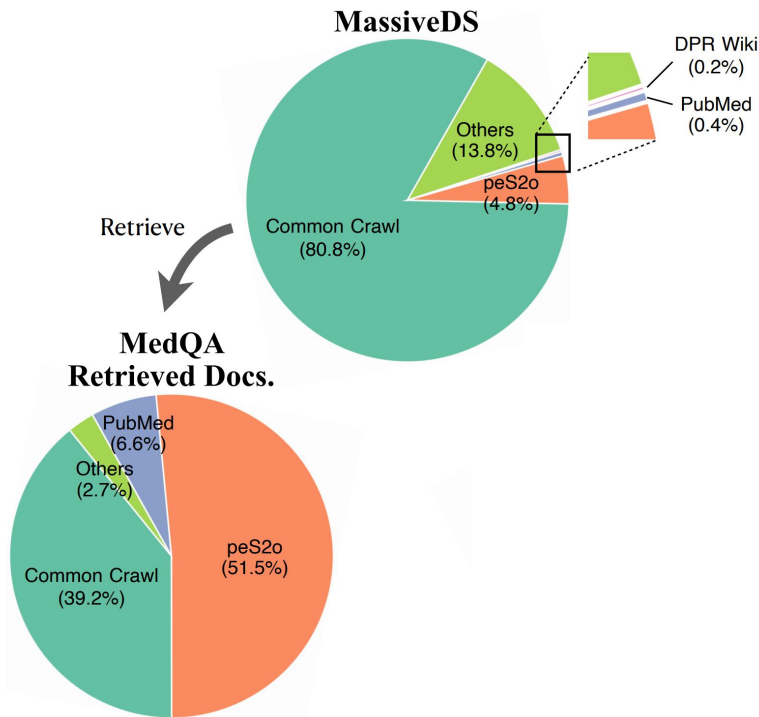
# Scaling retrieval-based Language Models

Results: comparison with single-domain datastores



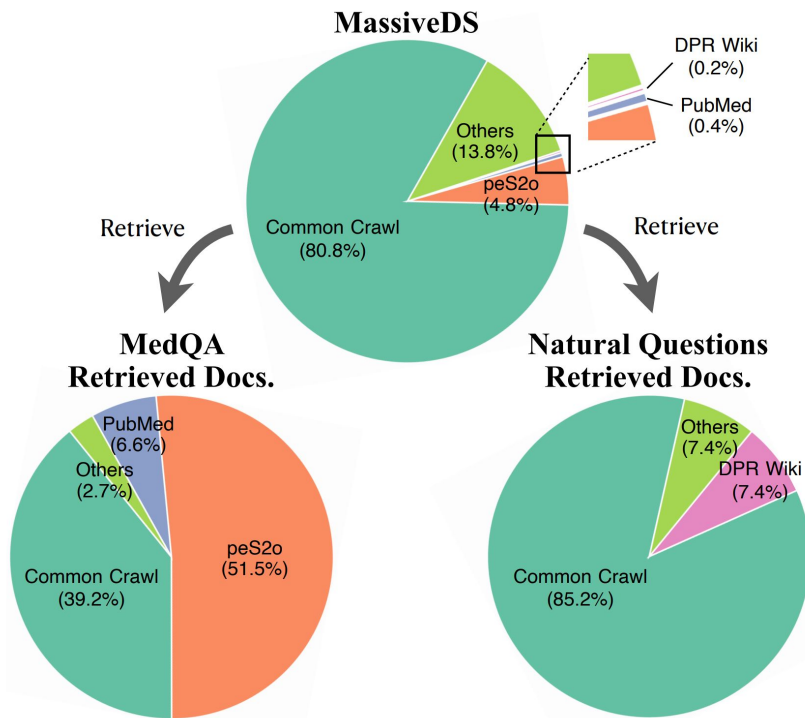
# Scaling retrieval-based Language Models

Results: comparison with single-domain datastores



# Scaling retrieval-based Language Models

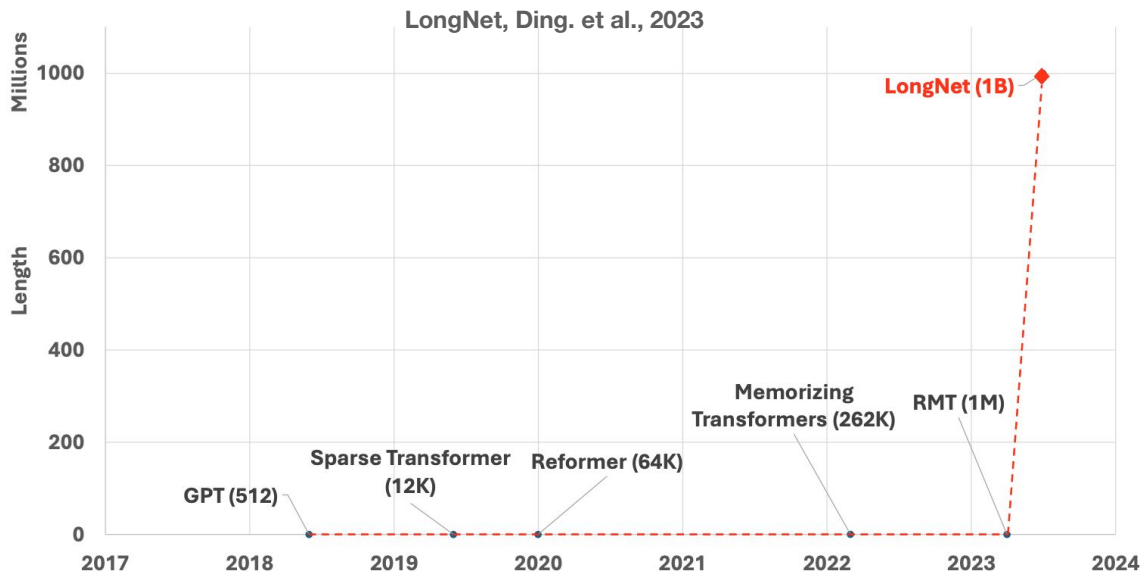
Results: comparison with single-domain datastores



# Scaling retrieval-based Language Models

## Discussion and future directions

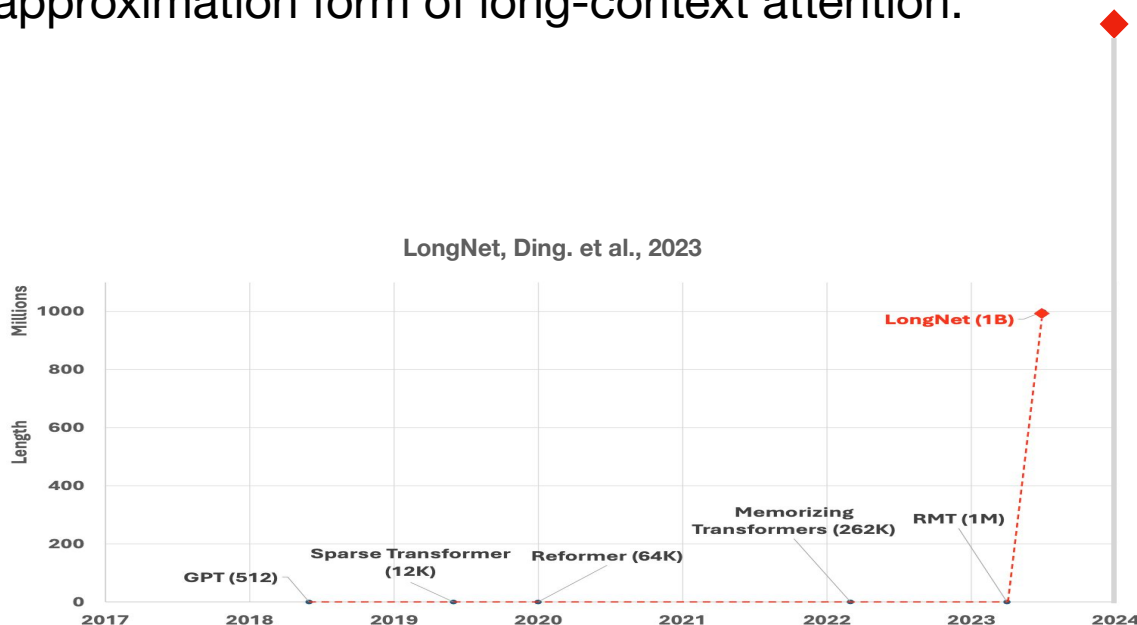
- Relationship with long-context modeling: consider retrieval as an extreme approximation form of long-context attention.



# Scaling retrieval-based Language Models

## Discussion and future directions

- Relationship with long-context modeling: consider retrieval as an extreme approximation form of long-context attention.



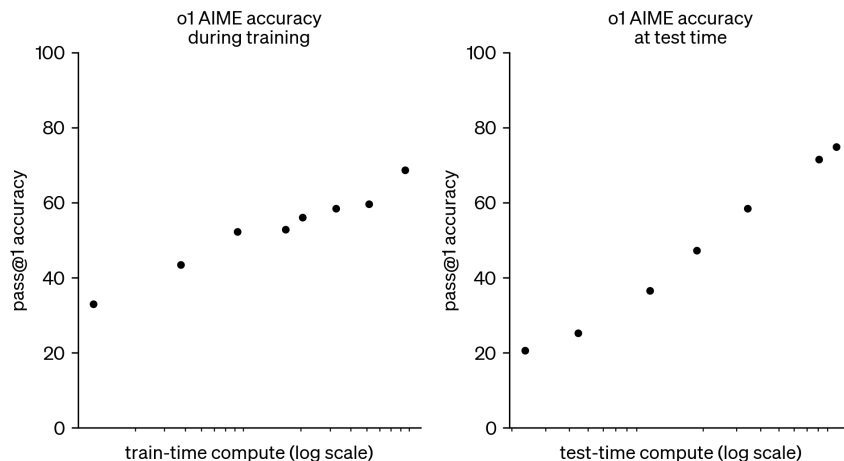
There are a few works discussing RAG v.s. long-context models:

- [Long-context LM wins] Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More? By Google
- [RAG wins] Long Context RAG Performance of LLMs. By Databricks

# Scaling retrieval-based Language Models

## Discussion and future directions

- Relationship with test-time scaling [1,2]: scaling datastore could be viewed as another way to do test-time scaling (with extra budget needed for datastore construction)



[1] Snell, Charlie, et al. "Scaling llm test-time compute optimally can be more effective than scaling model parameters." arXiv preprint arXiv:2408.03314 (2024).

[2] <https://openai.com/index/introducing-openai-o1-preview/>



# Scaling retrieval-based Language Models

## Discussion and future directions

- Efficiency aspects:
  - Datastore construction
  - Online serving

# Scaling retrieval-based Language Models

## Discussion and future directions

- How to make retrieval-based models helpful for reasoning-heavy tasks

# Critiquing Retrieval-based LMs

**Siddharth Gollapudi and Qiuyang Mang**

# Story so far

1. Retrieval/kNN distills information for LM in sublinear time
2. RBLMs make for a great example of test-time scaling
3. Most benchmarks see improved performance as the data store is scaled up

BUT

4. Reasoning performance?
5. Is retrieval the right tool for the problem?
6. Safety?

# Reasoning performance with/without datastores

	NQ	HotpotQA	Arc-Challenge	Arc-Easy	OBQA	MMLU
Llama2-7B	<b>23.18</b>	<b>22.72</b>	<b>41.81</b>	<b>57.49</b>	<b>57.00</b>	<b>39.22</b>
+Wiki	22.53	22.53	38.31	57.41	56.20	38.68
+Math	21.14	21.26	41.04	56.82	56.20	38.53
Llama3-8B	23.64	<b>25.14</b>	<b>44.88</b>	<b>58.83</b>	<b>55.80</b>	<b>42.67</b>
+Wiki	<b>24.00</b>	24.48	43.94	58.59	53.80	42.32
+Math	23.04	24.63	43.26	58.59	54.60	42.46
Mistral-7B	<b>20.63</b>	<b>20.96</b>	<b>46.42</b>	<b>60.94</b>	<b>58.80</b>	<b>41.91</b>
+Wiki	20.58	20.80	46.16	60.61	57.40	41.80
+Math	20.56	20.48	46.08	60.77	57.80	41.55

# Retrieval Quality is a problem

1. The actual retrieval data structure is essentially a black box
2. The encoder is not trained for retrieval wrt complex tasks
3. Blindly scaling the token store might not be everything

# Issue 1: ANN as a blackbox

1. Picking the right algorithm/data structure for the data distribution
2. Better worst case guarantees for tail-distributed embeddings

		Perplexity	Accuracy
OBQA	LM	255.76	55.80
	$k$ NN-LM	9.41	95.60
NQ	LM	112.56	23.64
	$k$ NN-LM	8.91	46.40
HotpotQA	LM	158.26	25.14
	$k$ NN-LM	8.15	49.85

Table 6: Results in an oracle setting where the  $k$ NN-LMs always include the correct answer as one of the  $k$  nearest neighbors.

## Issue 2: Encoder not trained for complex tasks

- Semantic similarity issues, e.g. supermarket giant vs largest supermarket

NQ Example	Label	LM Pred
who is the largest supermarket chain in the uk?	Tesco	Tesco
Retrieved context	Token	$k$ NN-LM Pred
• The majority of stores will open as normal across the UK, however Sainsbury's advise shoppers to check details of when your local branch as some may close earlier than normal using the online store locator tool.(Image: Bloomberg) Supermarket giant	Asda	
• Along with Lidl, Aldi has eaten away at the market share of the Big Four supermarkets:	Tesco	Asda
• buy one, get one free (BOGOF) offers have been criticised for encouraging customers to purchase food items that are eventually thrown away; as part of its own campaign on food waste, supermarket retailer	Morris	



# Embedding similarity is not good enough

Given a {product\_description}, find potential customs.

The text with highest similarity will be other products with the similar type / descriptions.

men's white puffer vest



Next  
White Slim Fit Ribbed Vest



Neiman Marcus  
Men's Treompan Quilted Zip Vest

# The Precision–Recall Tradeoff Is Dangerous (1)

Recall may be more important than precision in retrieval design!

1. LLM often can filter most of irrelevant chunks.
2. Context length goes longer and longer.
3. Finding useful chunks typically is a “needle-in-a-haystack” task

# The Precision–Recall Tradeoff Is Dangerous (2)

who is hanchen li's son



Show thinking ▼

Based on the information available, Hanchen Li's son is Huanzhi Mao.



Sources

# Issue 3: Scaling the token store isn't everything?

1. CompactDS keeps high quality data while maintaining diversity
2. Bonus: uses a second stage ENN to push end recall

	MMLU				MMLU Pro	AGI Eval	MATH	GPQA			AVG
	STEM	Human.	Social	Others				Phys	Bio	Chem	
No Retrieval	60.2	72.0	78.7	68.9	39.8	56.2	46.9	26.7	47.4	25.7	48.3
<b>Full MASSIVEDS (RAM use: 12.4TB)</b>											
ES Only (Contriever)	64.7	<b>81.7</b>	76.8	75.0	-	-	-	-	-	-	-
<b>COMPACTDS (RAM use: 0.5TB)</b>											
ANN Only (Contriever)	66.4	76.7	85.2	76.7	50.1	57.6	53.3	31.6	48.7	27.3	53.8
ANN (Contriever) + ES (Contriever)	64.8	75.8	83.6	75.5	50.0	59.0	52.9	32.1	<b>51.3</b>	24.6	53.6
ANN (Contriever) + ES (GRIT)	66.8	77.9	83.2	77.0	53.1	58.9	<b>55.9</b>	29.4	47.4	29.0	55.1
+ LM Reranking	<b>69.1</b>	77.8	<b>86.8</b>	<b>78.7</b>	<b>54.6</b>	<b>59.5</b>	53.0	<b>33.7</b>	47.4	<b>33.3</b>	<b>56.0</b>

# ReasonIR Reasoning Perf.

Retriever	Query Type	MMLU	GPQA
Closed-book	-	71.1	31.3
Contriever	Original question	72.0	36.4
GRIT-7B	Original question	74.1	32.3
Search Engine	Original question	-	33.8
<b>ReasonIR-8B</b>	Original question	75.0	<b>38.4</b>
Contriever	REASON-QUERY	72.8	31.3
GRIT-7B	REASON-QUERY	74.7	30.8
Search Engine	REASON-QUERY	-	36.4
<b>ReasonIR-8B</b>	REASON-QUERY	<b>75.6</b>	35.4

# Example ReasonIR Data

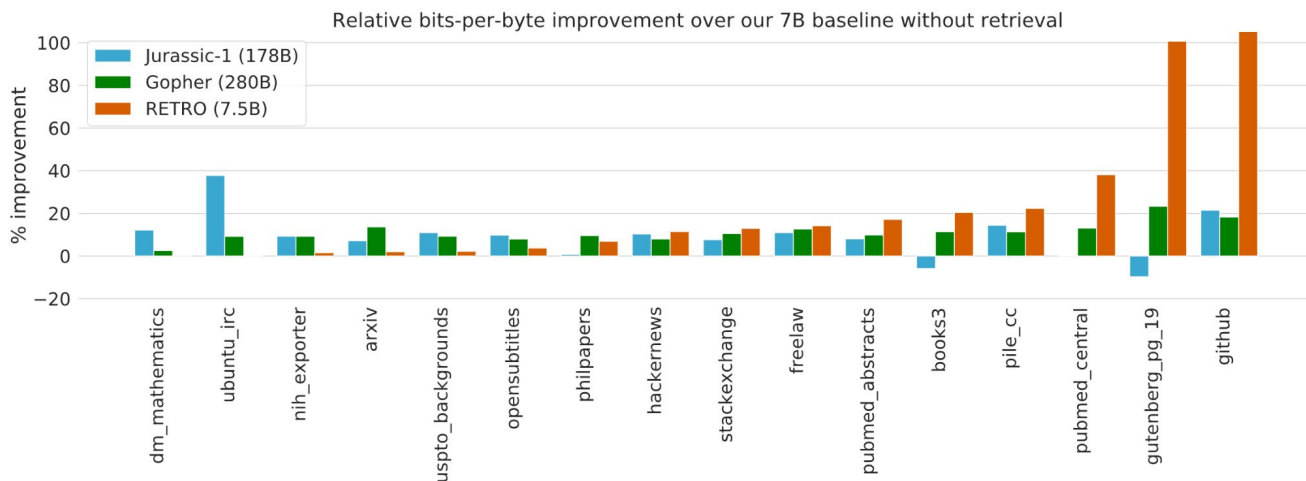
Query	A researcher is studying the sleep patterns of a group of individuals who work night shifts. The researcher notices that these individuals tend to have difficulty falling asleep during the day and experience fatigue during their work hours. What could be the primary factor contributing to this phenomenon, and how might it be related to the body's natural oscillations?
Positive Doc	A circadian rhythm (/səˈkeɪdiən/), or circadian cycle, is a natural oscillation that repeats roughly every 24 hours. Circadian rhythms can refer to any process that originates within an organism (i.e., endogenous) and responds to the environment (is entrained by the environment). Circadian rhythms are regulated by a circadian clock whose primary function is to rhythmically co-ordinate biological processes so they occur at the correct time to maximise the fitness of an individual. Circadian rhythms have been widely observed in animals, plants, fungi and cyanobacteria and there is evidence that they evolved independently in each of these kingdoms of life. The term circadian comes from the Latin circa, meaning "around", and dies, meaning "day". Processes with 24-hour cycles are more generally called diurnal rhythms; diurnal rhythms should not be called circadian rhythms unless they can be confirmed as endogenous, and not environmental. Although circadian rhythms are endogenous, they are adjusted to the local environment by external cues called zeitgebers (from German Zeitgeber (German: [ˈtsaɪt.ɡeːbɐ]; lit.'time giver'), which include light, temperature and redox cycles. In clinical settings, an abnormal circadian rhythm in humans is known as a circadian rhythm sleep disorder.
Hard Negative	During the night shift, individuals often experience an increase in body temperature, which can lead to discomfort and difficulty maintaining focus. This increase in temperature is usually highest in the late evening, around 10-11 pm, and gradually decreases as the night progresses. Body temperature is known to be controlled by the hypothalamus, which acts as the body's thermostat. The hypothalamus responds to changes in the body's core temperature to cool the body or warm it up through various mechanisms. These processes are an essential element of the body's natural responses, but understanding their relationship to the observed phenomenon of difficulty during the night shift is somewhat complex. Factors affecting the body's core temperature include the ambient temperature, intensity of workouts, and personal characteristics. The hypothalamus responds to the ambient temperature and helps maintain the body's core temperature. When the ambient temperature is high, sweating and other heat-loss mechanisms are activated, whereas low ambient temperatures result in the body conserving heat through the constriction of blood vessels near the skin. A higher intensity of workouts or engaging in activities that utilize more muscle mass increases body temperature. The thermoregulation response also varies across individuals based on characteristics including age, sex, and fitness level. However, the night shift affects body temperature, the same way regardless. It does not exhibit the same variations between individuals as natural temperature regulation might. Various studies have demonstrated the adverse effects of elevated body temperatures on sleep and vigilance. Revealing a correlation, rather than a causality between high body temperatures during night shifts and the subjective experience of discomfort with day-time sleep. These various factors interacting shows some overlaps with but also confusion with the issues experienced by people during the night shift study prompt.

# Proponent of Retrieval-based LMs

Sanjay Adhikesaven and Junyi Zhang

# The Breakthrough: A New, Efficient Scaling Paradigm

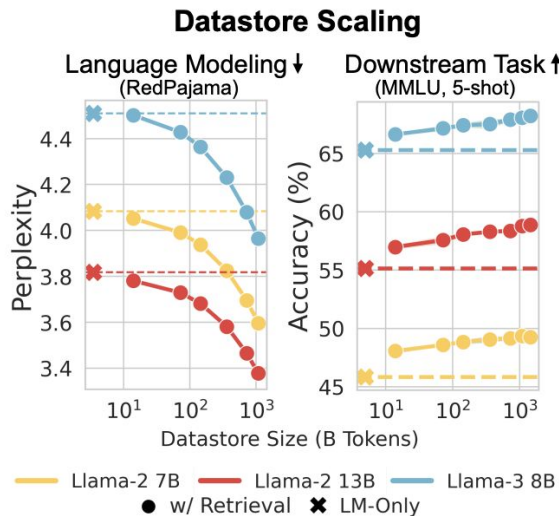
- **The Core Problem:** Scaling Language Models via parameters alone is computationally expensive, inefficient for storing knowledge, and difficult to update.
- **The RETRO Solution: Decouple knowledge from computation.** RETRO enhances a smaller language model by giving it access to a massive external memory—a 2 trillion token database.
- **Strong Result:** RETRO achieves performance comparable to models **25x larger** (like GPT-3 and Jurassic-1) on The Pile benchmark.



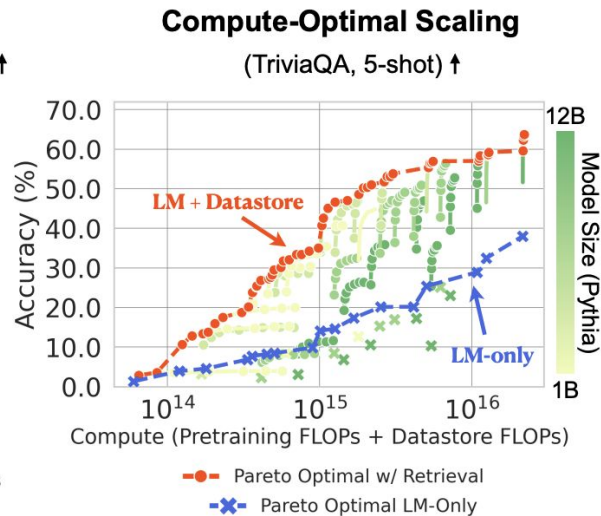


# Why Retrieval Scaling Matters

- We can scale knowledge at inference, not just params/train tokens
- MassiveDS (1.4T) is open and diverse



Bigger datastores →  
monotonic gains  
across tasks



For same training  
budget, retrieval-based  
LMs beat LM-only

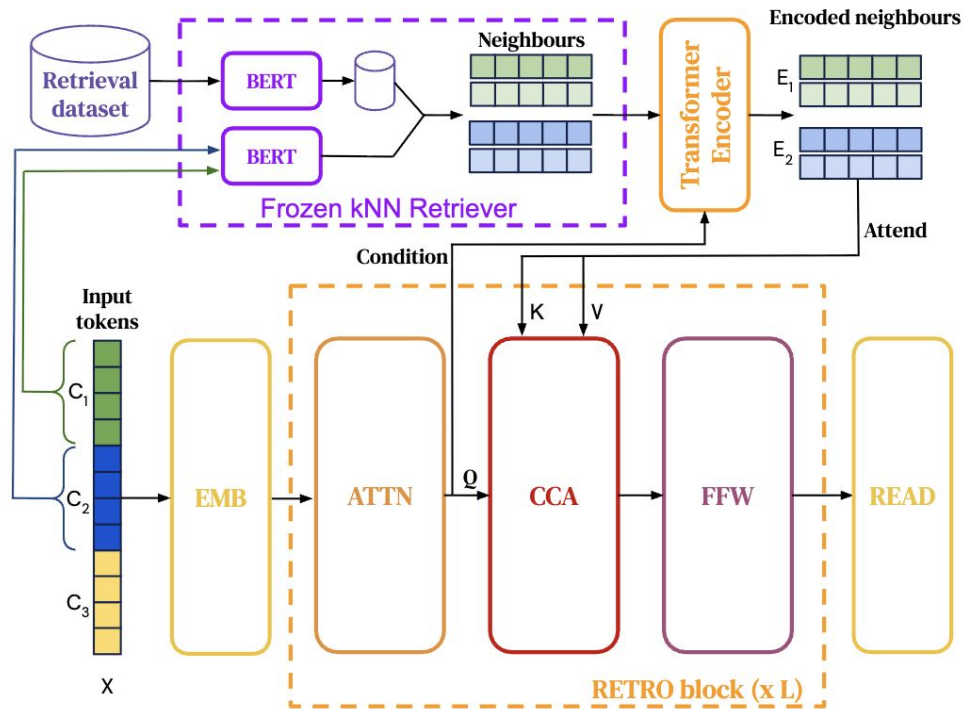
# Quality vs Quantity: Quantity is Required

- Table 3: **Downstream and upstream performance comparison between MASSIVEDS for retrieval versus single-domain datastores with LLAMA-2 7B.** “SE” is short for StackExchange. The best performance is highlighted in **bold** and the second best is underlined. We show the diverse domain coverage in MASSIVEDS consistently improve the performance across tasks.

Tasks	LM-Only	PubMed	MATH	peS2o	DPR Wiki	RedPajama					MASSIVEDS
						Wiki	Books	ArXiv	SE	Github	
TQA ↑	64.1	64.5	65.5	65.6	72.6	<u>72.9</u>	70.5	62.3	64.7	64.2	<b>77.0</b>
NQ ↑	26.6	26.7	26.4	26.9	<b>34.6</b>	<u>33.8</u>	28.0	26.4	27.0	26.4	<b>34.6</b>
MedQA ↑	36.6	37.8	36.5	38.1	38.5	38.4	<b>39.8</b>	36.9	35.4	36.1	<u>39.4</u>
MMLU ↑	45.8	46.8	47.5	47.4	48.3	48.1	<u>48.3</u>	45.6	46.2	45.9	<b>49.3</b>
RedPajama (PPL) ↓	4.09	4.06	4.08	4.08	4.06	3.99	4.01	<u>3.87</u>	4.01	3.95	<b>3.50</b>
S2ORC (PPL) ↓	7.18	7.05	7.10	6.71	7.08	7.11	7.14	<u>6.64</u>	7.08	7.11	<b>6.57</b>

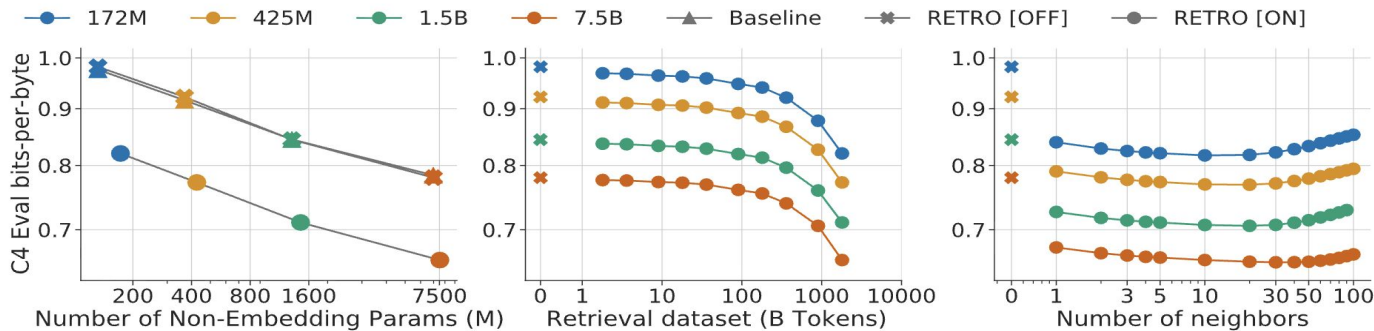
# Elegant Design for Trillion-Token Scale

- **Key to Scalability: Frozen kNN Retriever**  
**Retriever:** RETRO uses a *frozen* pre-trained BERT to create the database index.
- **Efficient Integration: Chunked Cross-Attention (CCA)**: Information from retrieved neighbors is integrated efficiently. To maintain causality, tokens in the current chunk attend to neighbors retrieved for the *previous* chunk.
- **Practical Innovation: "RETRO-fitting"**: Any pre-trained Transformer can be rapidly upgraded with retrieval capabilities.



# Empirical Proof and Lasting Impact

- **Clear Scaling Laws:** The RETRO paper demonstrates that performance consistently improves with:



Model size (from 172M to 7.5B parameters). Retrieval database size (up to ~2 trillion tokens). Number of retrieved neighbors at inference time.

- **Not Just Memorization:** A novel "leakage analysis" proves that gains are not just from copy-pasting. RETRO still outperforms baselines significantly even on evaluation chunks with minimal overlap (<8 continuous tokens) with the training data.
- **Proven Generalization:** RETRO shows consistent gains on a dataset of Wikipedia articles written in September 2021—months *after* the training data was collected—proving it can leverage its knowledge base for new, unseen topics.

# MassiveDS: Retrieval Scaling Works

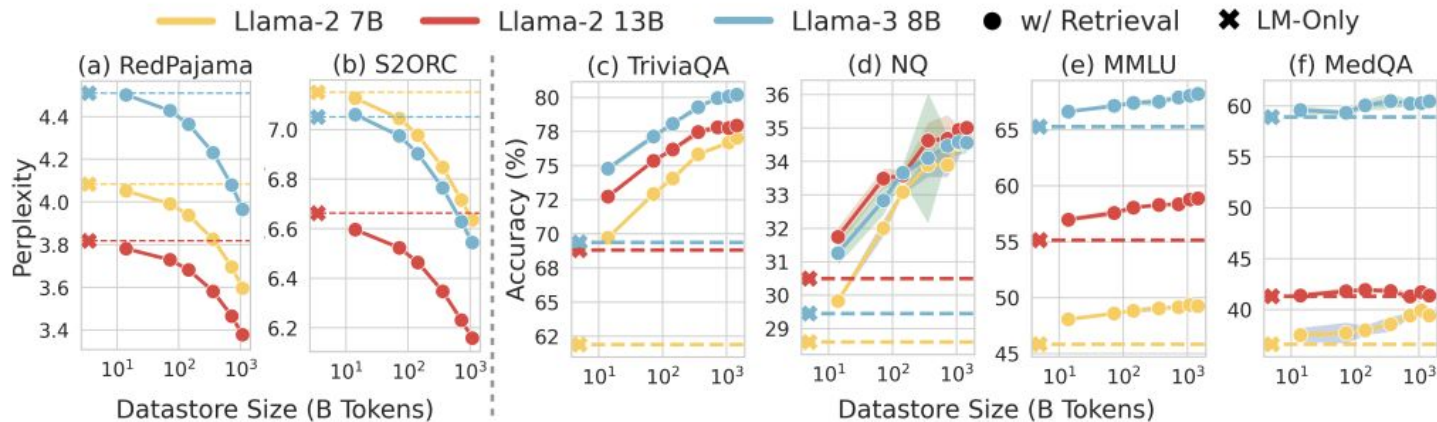
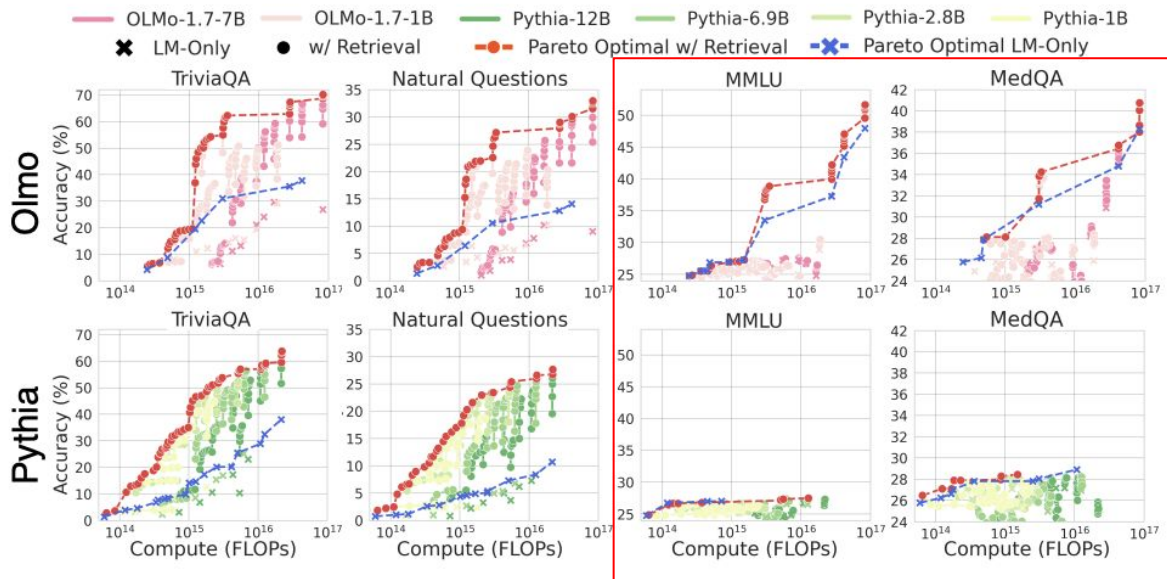


Figure 3: **Scaling performance on upstream and downstream tasks with MASSIVEDS, in comparison with LM-only performance.** *Left:* Perplexity (PPL) scaling performance on REDPAJAMA (multi-domain pretraining corpus) and S2ORC (scientific papers). *Right:* Downstream scaling performance on TriviaQA (TQA), Natural Questions (NQ), MMLU, and MedQA.

# Reasoning gains are model dependent



- Retrieval helps when the LM is capable and the datastore covers the domain
  - For instance, Pythia-12B on MMLU/MedQA shows little benefit → this is a capability/data issue and isn't an issue with retrieval