UNIVERSITY of WASHINGTON

Retrieval-augmented Language Models

Luke Zettlemoyer

NLP and LLMs

Slides adapted from ACL 2023 tutorial by Akari Asai, Sewon Min, Zexuan Zhong, Danqi Chen

https://acl2023-retrieval-lm.github.io/

How do LMs work?

The capital city of Ontario is **Toronto**

LM

Training time

The capital city of Ontario is _____

LM

Test time

Hallucinations in LM outputs

You

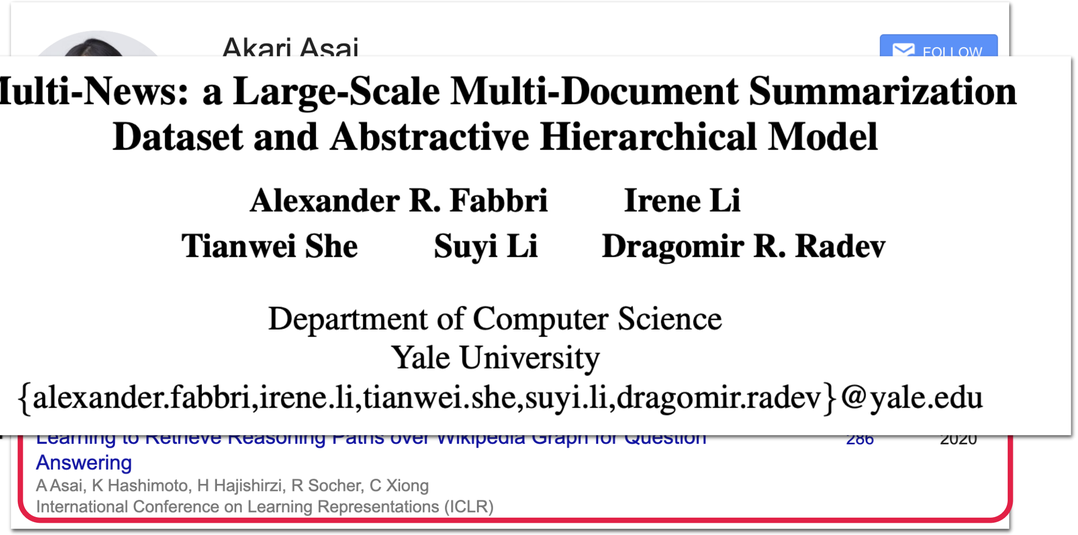
Explain Akari Asai, including the list of her famous papers. (*Febru

ChatGPT

Akari Asai is a prominent figure in the field of natural language processing
significant contributions to various aspects of NLP research. She has mad
particularly in the areas of question answering, dialogue systems, and mac

Here are some of her famous papers:

1. "Learning to Retrieve Reasoning Paths over Wikipedia Graph for Question Answering" - In this paper, Asai and her co-authors proposed a method for learning to retrieve reasoning paths over the Wikipedia graph to improve question answering systems.
2. "Large-Scale Multi-Document Summarization Dataset and Evaluation Benchmark" - This paper introduces a large-scale dataset for multi-document summarization, along with an evaluation benchmark, providing valuable resources for advancing research in this area.



Multi-News: a Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model

Alexander R. Fabbri **Irene Li**
Tianwei She **Suyi Li** **Dragomir R. Radev**

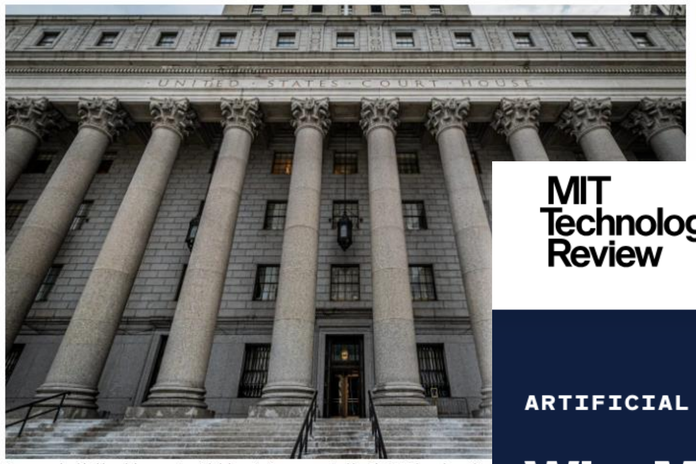
Department of Computer Science
Yale University
{alexander.fabbri,irene.li,tianwei.she,suyi.li,dragomir.radev}@yale.edu

Catastrophic Errors as Results of LM Hallucinations

TECH · LAW

Humiliated lawyers fined $5,000 for submitting ChatGPT hallucinations in court: 'I heard about this new site, which I falsely assumed was, like, a super search engine'

BY RACHEL SHIN
June 23, 2023 at 9:41 AM PDT



Lawyers who filed legal documents with false citations generated by ChatGPT have been fine

ERIK MCGREGOR—LIGHTROCKET/GETTY IMAGES

MIT Technology Review

Featured Topics Newsletters Events Podcasts SIGN IN

ARTIFICIAL INTELLIGENCE

Why Meta's latest large language model survived only three days online

Galactica was supposed to help scientists. Instead, it mindlessly spat out biased and incorrect nonsense.

By Will Douglas Heaven November 18, 2022

Air Canada must honor re… invented by airline's chatb…

Air Canada appears to have quietly killed its costly chatbot support.

ASHLEY BELANGER - 2/16/2024, 12:12 PM

Retrieval-augmented LMs

The capital city of Ontario is **Toronto**

LM

Training time

The capital city of Ontario is _____

LM

Test time

Inference

Datastore

Query

Index

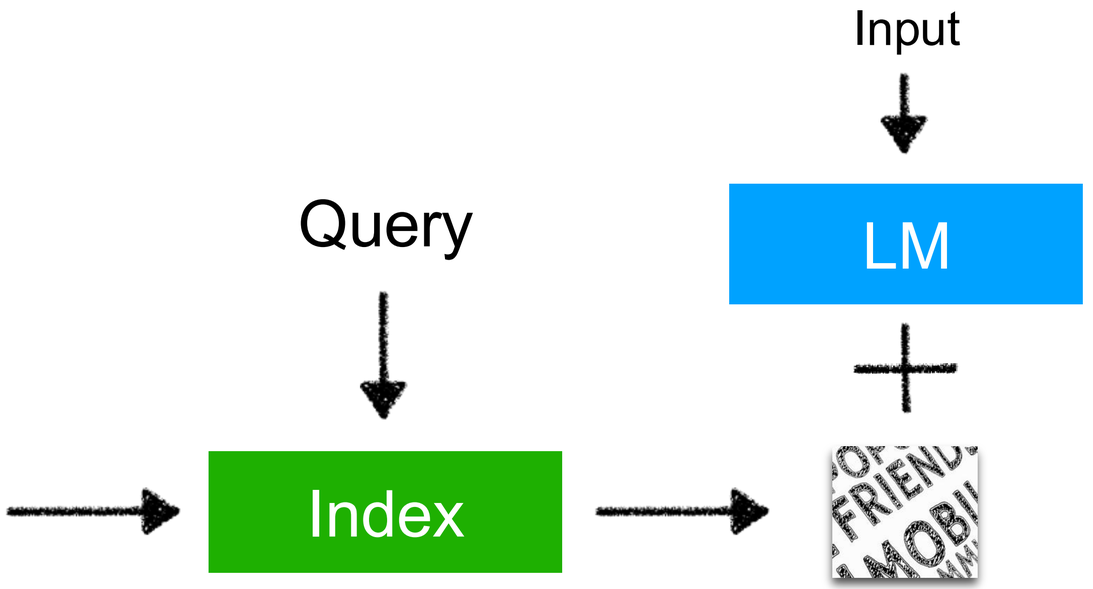
Input

LM

Inference: Datastore

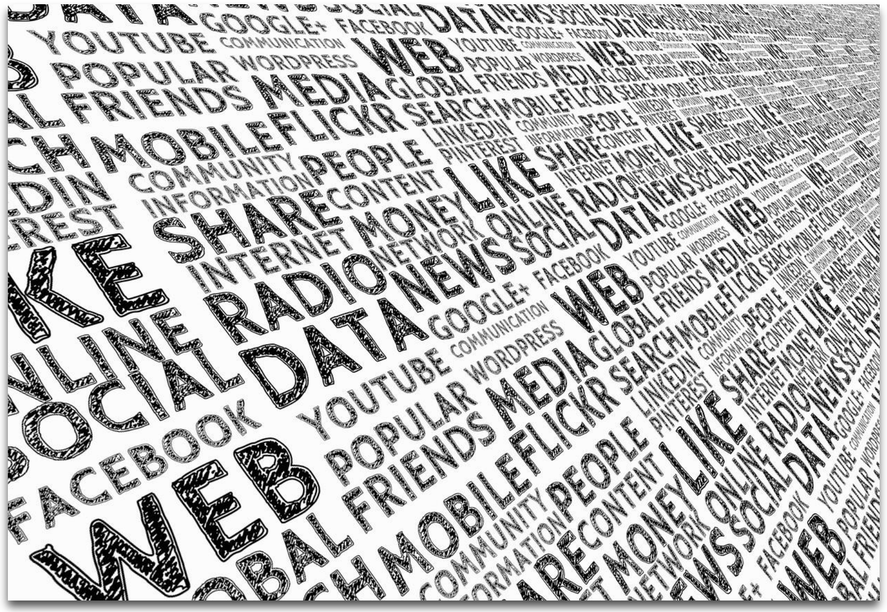
Datastore

Raw text corpus

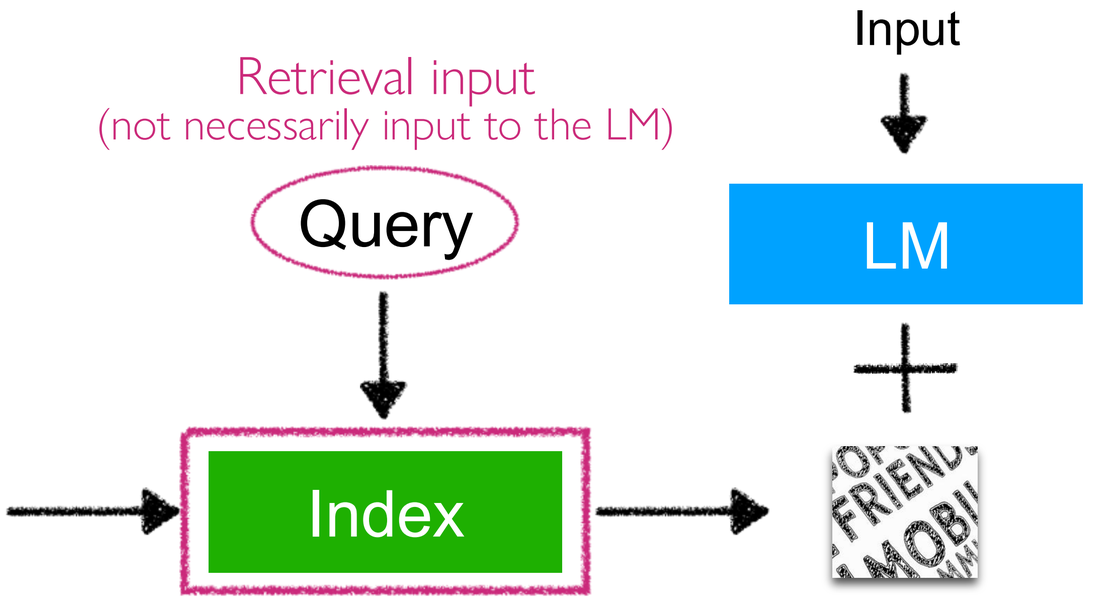


At least billions~trillions of tokens
Not labeled datasets
Not structured data (knowledge bases)

Inference: Index

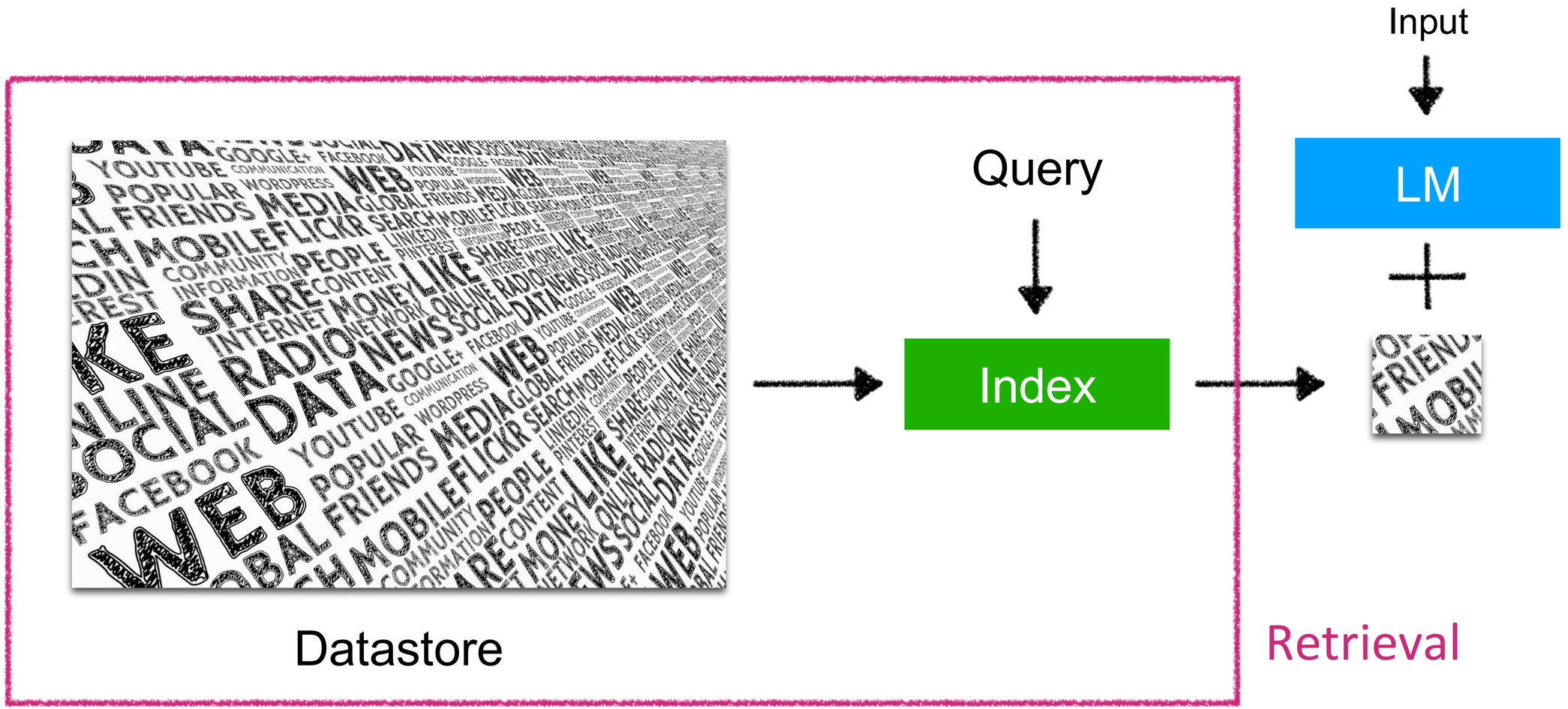


Datastore



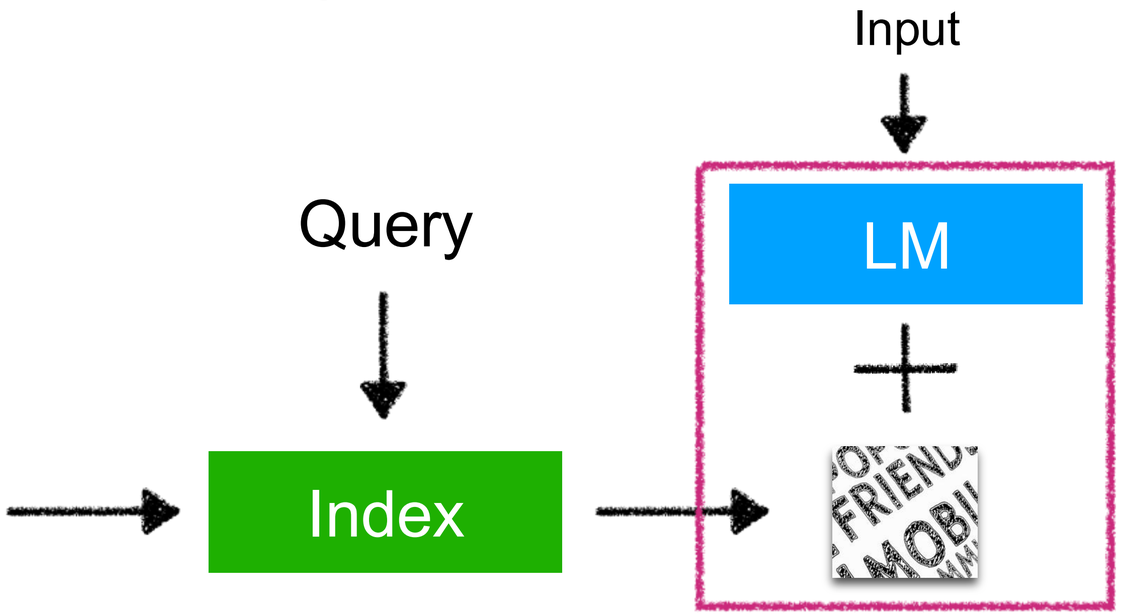
Find a small subset of elements in a datastore that are the most similar to the query

Inference: Search



Inference: Incorporation

Datastore



Retrieval-augmented LMs are now widely used!

YOU

Perplexity

Today's outline

Why do we need retrieval-augmented LMs?

Architectures of retrieval-augmented LMs (Inference)

Training of retrieval-augmented LMs

Limitations and future directions

Today's outline

Why do we need retrieval-augmented LMs?

Architectures of retrieval-augmented LMs (Inference)

Training of retrieval-augmented LMs

Limitations and future directions

A: Because retrieval-augmented LMs can solve many core limitations of parametric LMs!

Core limitations of parametric LMs

Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

AK **You**

Explain Akari Asai, including the list of her famous papers. (*February 18, 2024)

ChatGPT

Akari Asai is a prominent figure in the field of natural language processing (NLP), known for her significant contributions to various aspects of NLP research. She has made notable contributions particularly in the areas of question answering, dialogue systems, and machine learning.

Here are some of her famous papers:

2. "Large-Scale Multi-Document Summarization Dataset and Evaluation Benchmark" - This paper introduces a large-scale dataset for multi-document summarization, along with an evaluation benchmark, providing valuable resources for advancing research in this area.

Core limitations of parametric LMs

Hallucinations

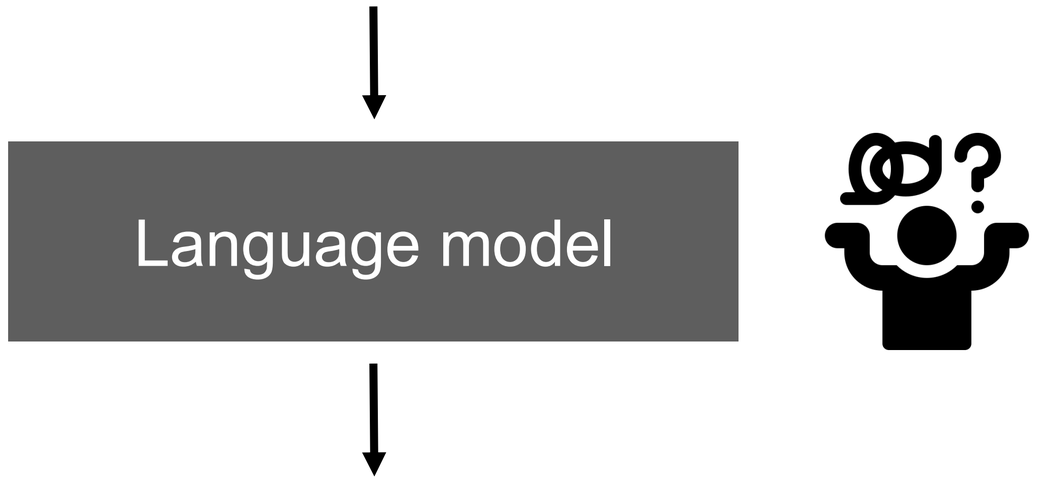
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Explain Akari Asai, including the list of her famous papers.



Her most famous paper is “*Large-Scale Multi-Document Summarization Dataset and Evaluation Benchmark*”

Core limitations of parametric LMs

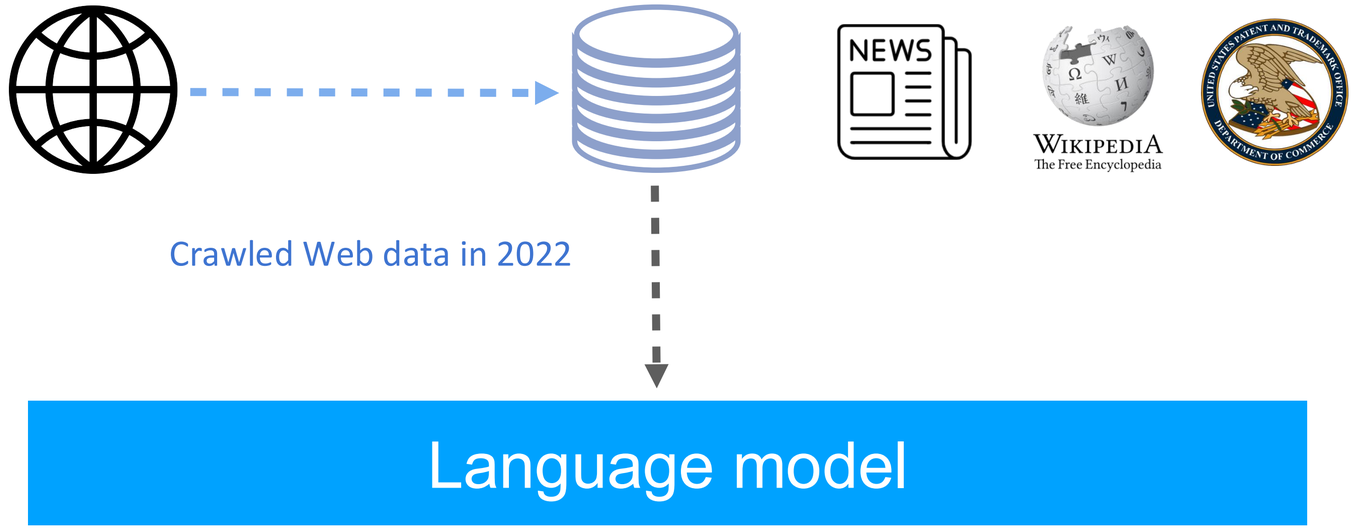
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



ChatGPT

I'm sorry, but I don't have access to real-time information including events beyond January 2022.

Core limitations of parametric LMs

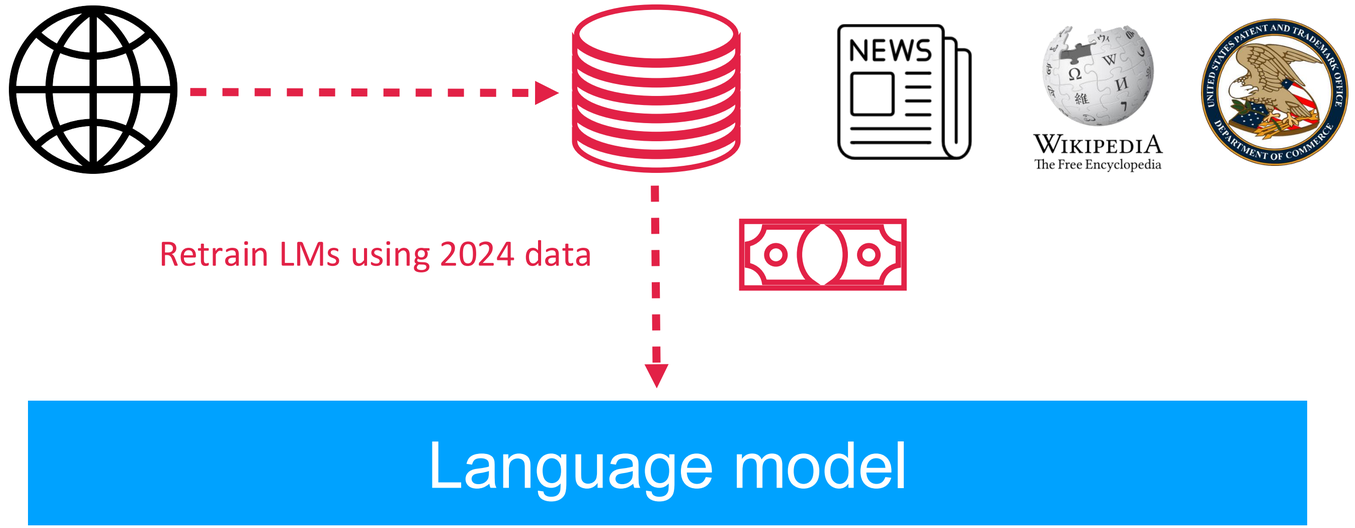
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



ChatGPT

I'm sorry, but I don't have access to real-time information including events beyond January 2022.

Core limitations of parametric LMs

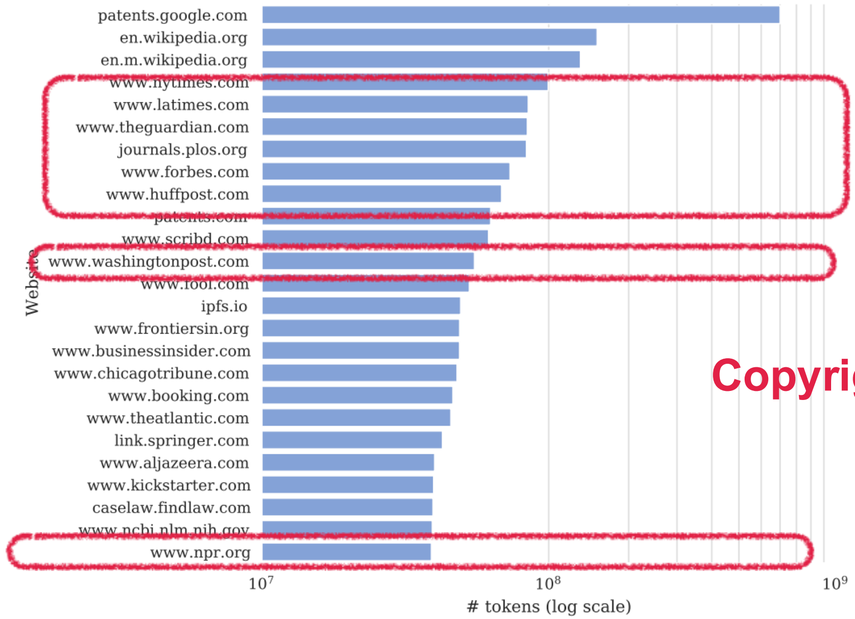
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



Copyright-protected data?

Dodge et al., Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus. EMNLP 2021.

Core limitations of parametric LMs

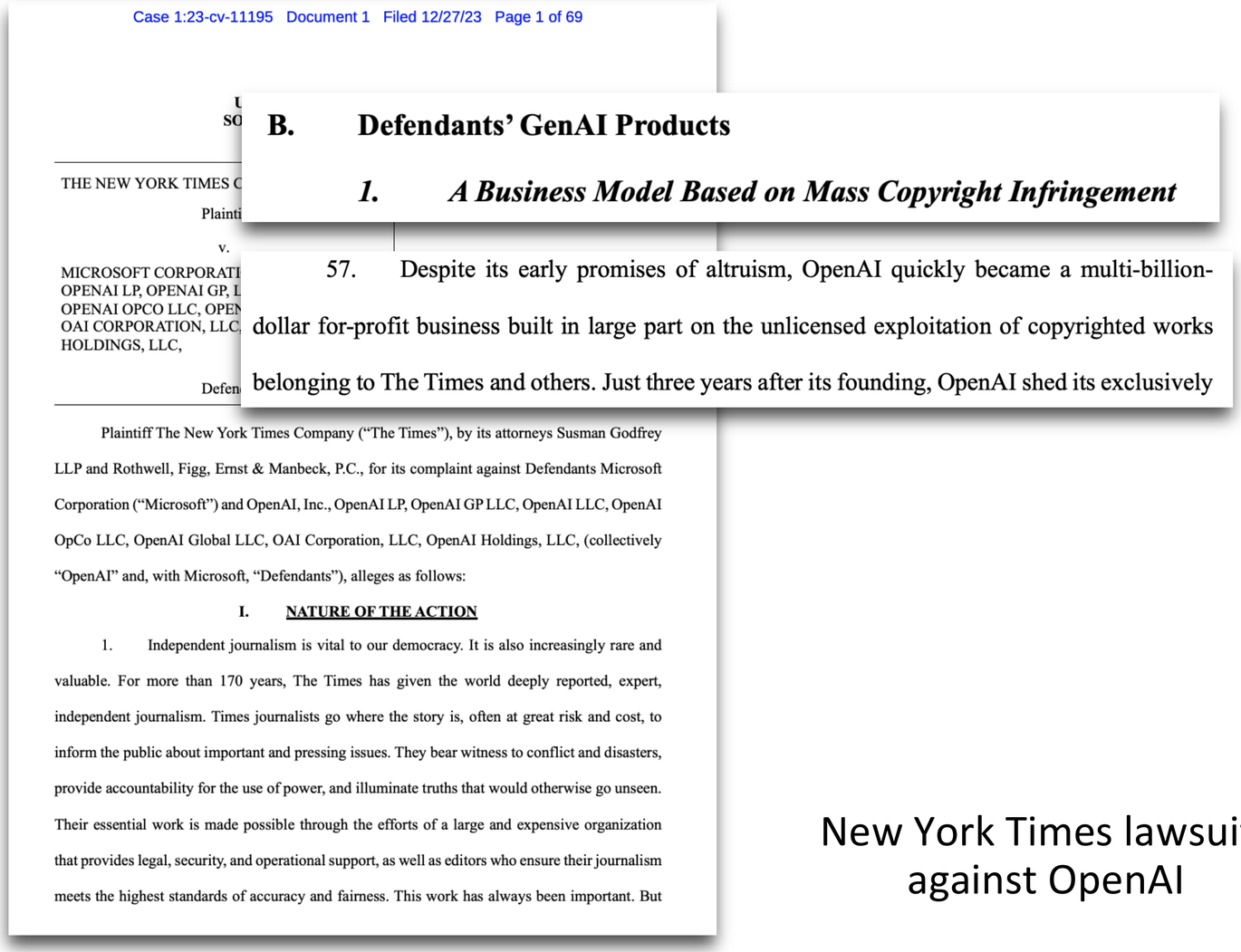
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



Case 1:23-cv-11195 Document 1 Filed 12/27/23 Page 1 of 69

THE NEW YORK TIMES C

Plainti

v.

MICROSOFT CORPORATI
OPENAI LP, OPENAI GP, L
OPENAI OPCO LLC, OPEN
OAI CORPORATION, LLC,
HOLDINGS, LLC,

Defen

Plaintiff The New York Times Company ("The Times"), by its attorneys Susman Godfrey LLP and Rothwell, Figg, Ernst & Manbeck, P.C., for its complaint against Defendants Microsoft Corporation ("Microsoft") and OpenAI, Inc., OpenAI LP, OpenAI GP LLC, OpenAI LLC, OpenAI OpCo LLC, OpenAI Global LLC, OAI Corporation, LLC, OpenAI Holdings, LLC, (collectively "OpenAI" and, with Microsoft, "Defendants"), alleges as follows:

I. NATURE OF THE ACTION

1. Independent journalism is vital to our democracy. It is also increasingly rare and valuable. For more than 170 years, The Times has given the world deeply reported, expert, independent journalism. Times journalists go where the story is, often at great risk and cost, to inform the public about important and pressing issues. They bear witness to conflict and disasters, provide accountability for the use of power, and illuminate truths that would otherwise go unseen. Their essential work is made possible through the efforts of a large and expensive organization that provides legal, security, and operational support, as well as editors who ensure their journalism meets the highest standards of accuracy and fairness. This work has always been important. But

B. Defendants' GenAI Products

1. A Business Model Based on Mass Copyright Infringement

57. Despite its early promises of altruism, OpenAI quickly became a multi-billion-dollar for-profit business built in large part on the unlicensed exploitation of copyrighted works belonging to The Times and others. Just three years after its founding, OpenAI shed its exclusively

New York Times lawsuits against OpenAI

Core limitations of parametric LMs

Hallucinations

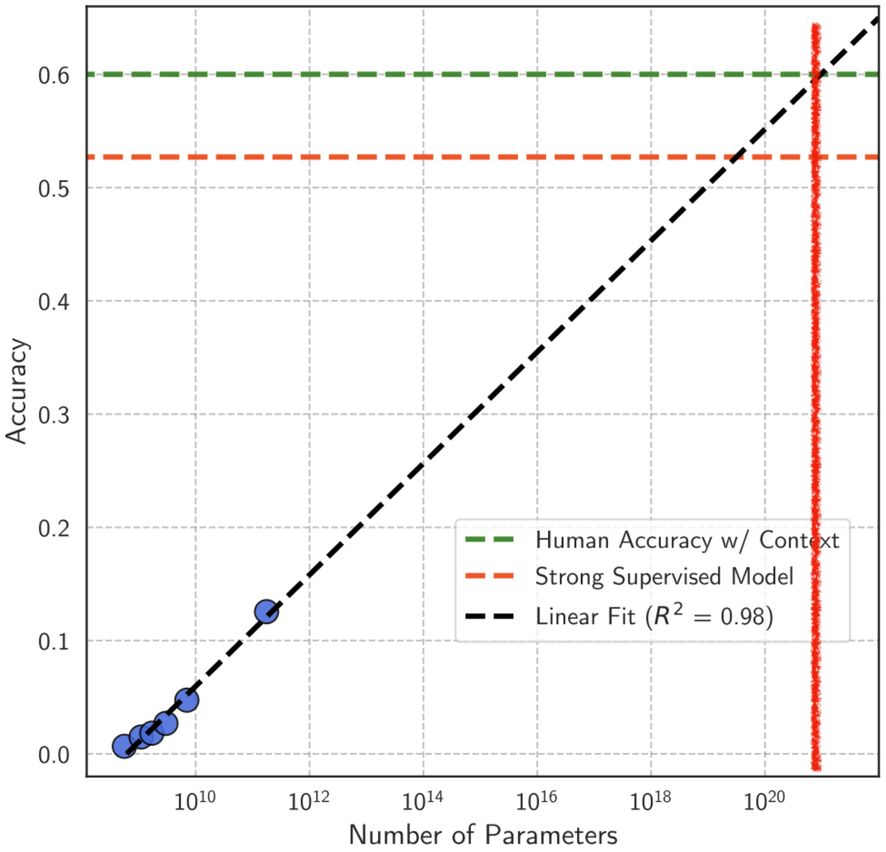
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Long-tail QA performance



100 quintillion parameters required to reach human performance

Kandpal et al., Large Language Models Struggle to Learn Long-Tail Knowledge. ICML 2023.

Q: So how can retrieval-augmented LMs solve those challenges?

How retrieval-augmented LMs solve the issues?

Hallucinations

Lack of attributions

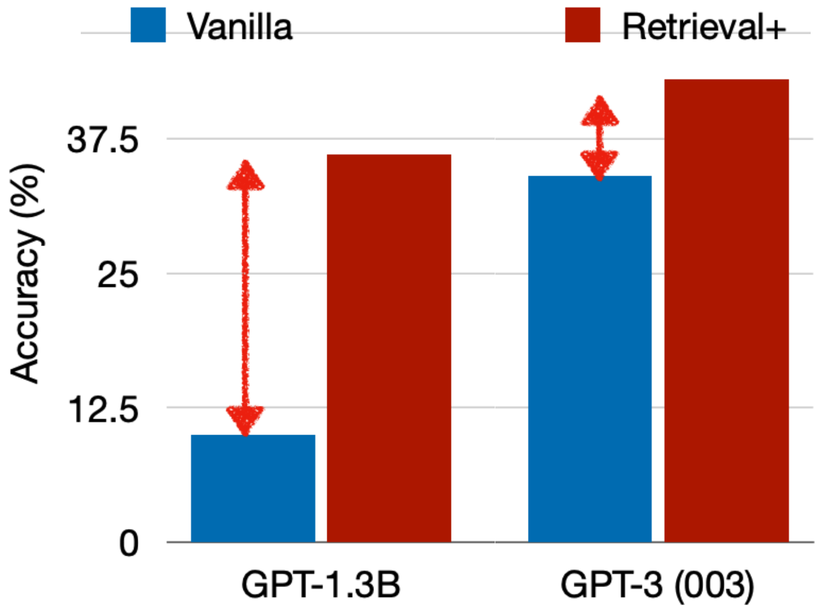
Costs of adaptations

Copyright / privacy

Large parameter size

Significant improvements across model scale, with larger gain with smaller LMs

QA



Mallen*, Asai* et al. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories (Best Video; Oral) 2023.

How retrieval-augmented LMs solve the issues?

Hallucinations

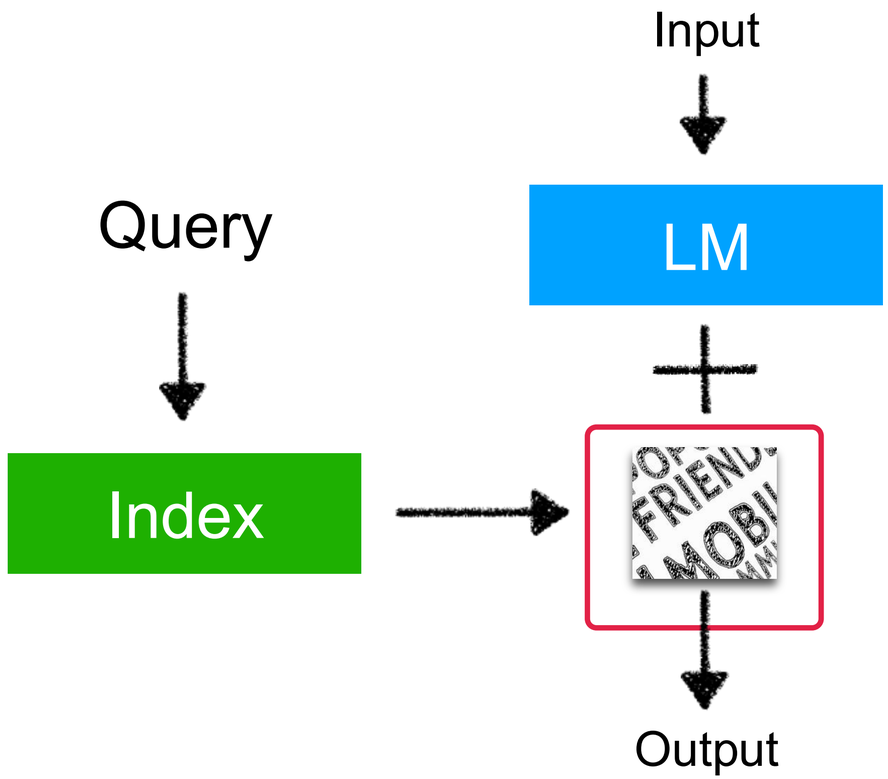
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Retrieved text can be used as attributions



How retrieval-augmented LMs solve the issues?

Hallucinations

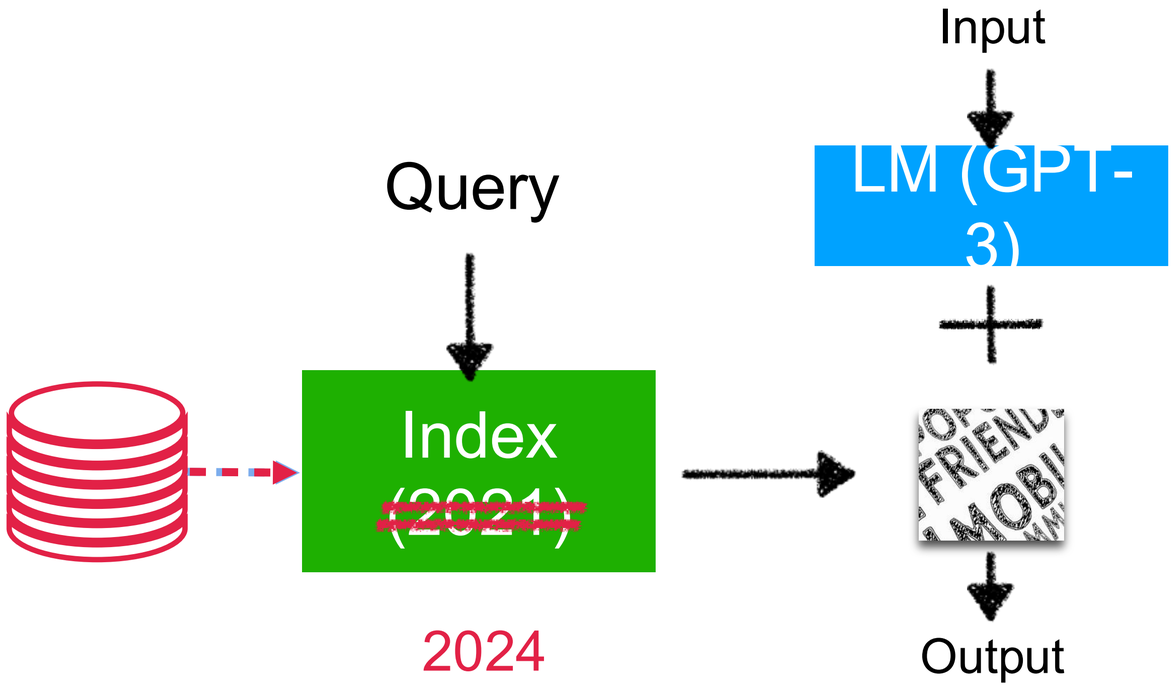
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Replacing datastores to catch up dynamically changing world without re-training



Kasai et al.,. REALTIME QA: What's the Answer Right Now. NeurIPS Dataset and Benchmark 2023.

How retrieval-augmented LMs solve the issues?

Hallucinations

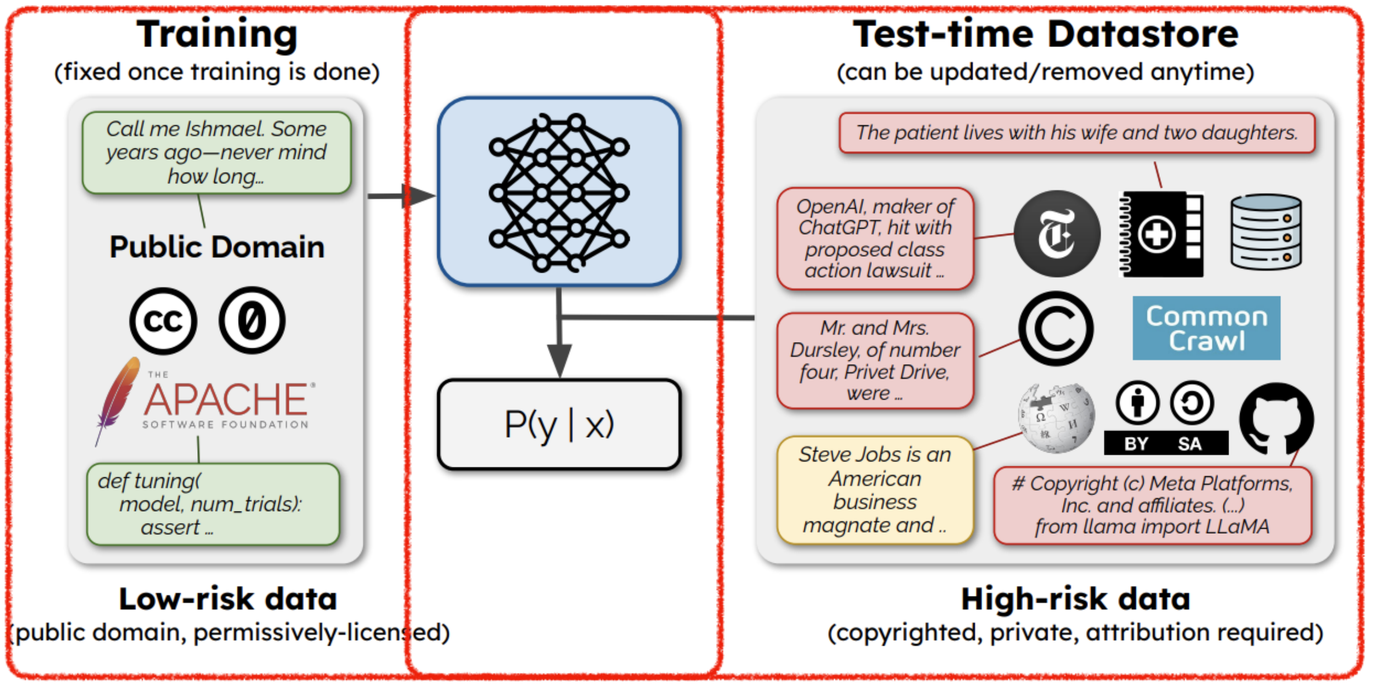
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Segregating copyright-sensitive data from pre-training data



Min* and Gururangan* et al., SILO Language Models: Isolating Legal Risk In a Nonparametric Datastore. ICLR 2024.

How retrieval-augmented LMs solve the issues?

Hallucinations

Lack of attributions

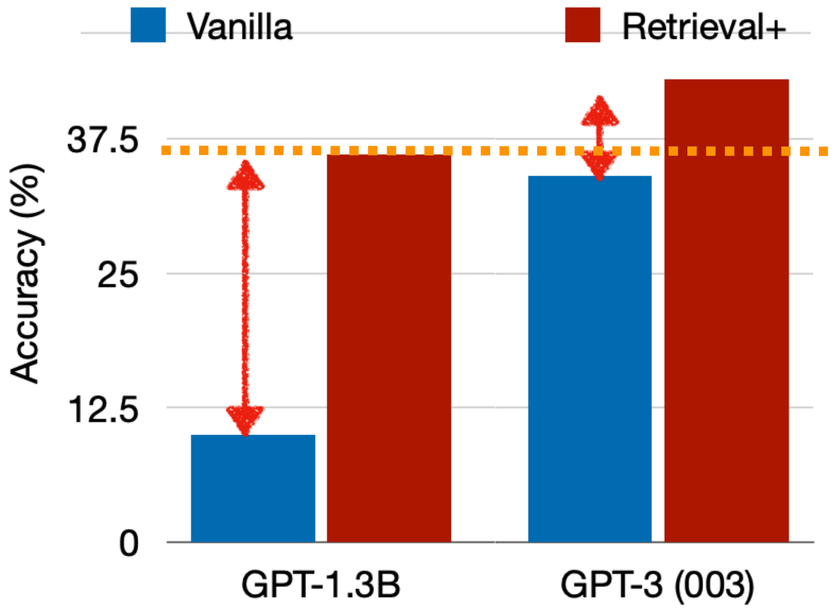
Costs of adaptations

Copyright / privacy

Large parameter size

Smaller LMs with retrieval outperform much larger LMs e.g., GPT-3

QA



Mallen*, Asai* et al., When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories (Best Video; Oral) 2023.

Today's outline

Why do we need retrieval-augmented LMs?

Architectures of retrieval-augmented LMs (Inference)

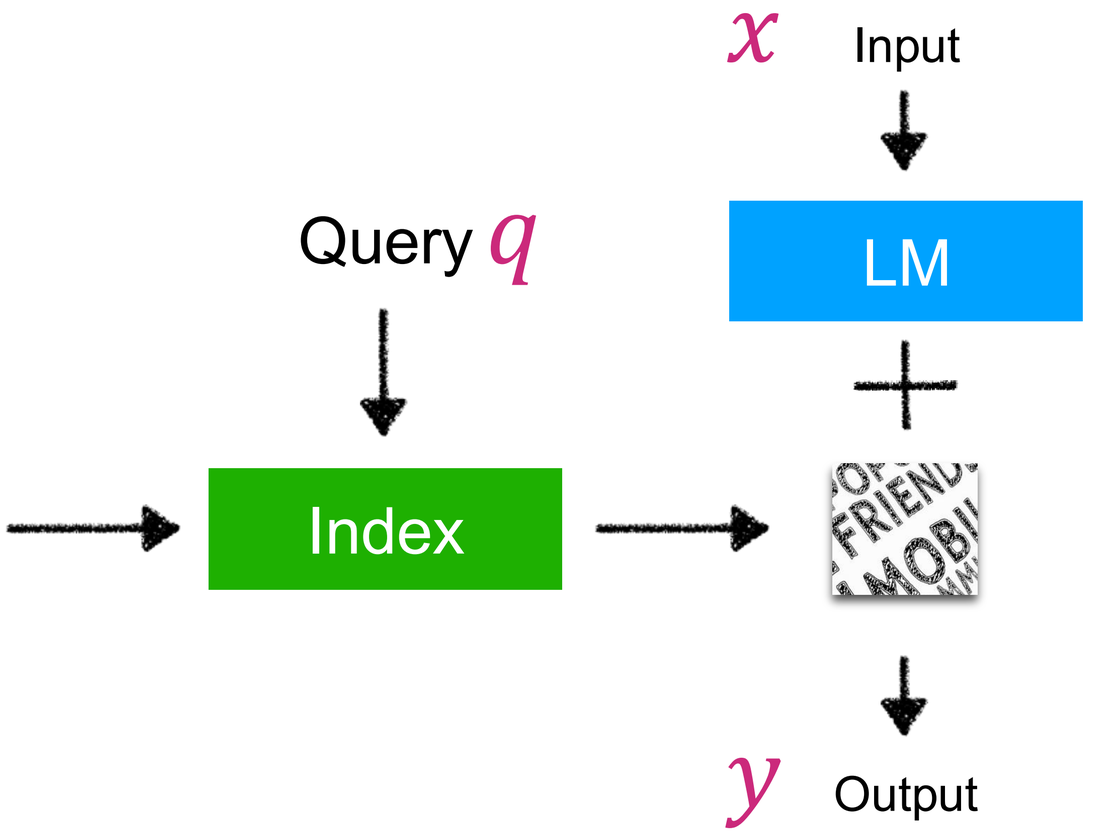
Training of retrieval-augmented LMs

Limitations and future directions

Notations

Datastore

$\mathcal{D}$



Inference: Index

Goal: find a small subset of elements in a datastore
that are the most similar to the query

sim: a similarity score between two pieces of text

Example $\text{sim}(i, j) = \text{tf}_{i,j} \times \log \frac{N}{\text{df}_i}$

$\text{tf}_{i,j}$: # of occurrences of i in j

N: # of total docs

df_i: # of docs containing i

Example $\text{sim}(i, j) = \text{Encoder}(i) \cdot \text{Encoder}(j)$

Maps the text into an h-dimensional vector

An entire field of study on how to get (or learn) the similarity function better
(We'll see some later!)

Inference: Index

Goal: find a small subset of elements in a datastore
that are the most similar to the query

$\mathbf{sim}$: a similarity score between two pieces of text

Separate research on how
to do this fast & accurate

Index: given q, return $\mathbf{argTop}\text{-}k_{d \in \mathcal{D}} \mathbf{sim}(q, d)$ through fast nearest neighbor search

k elements from a datastore

https://github.com/facebookresearch/faiss/wiki/

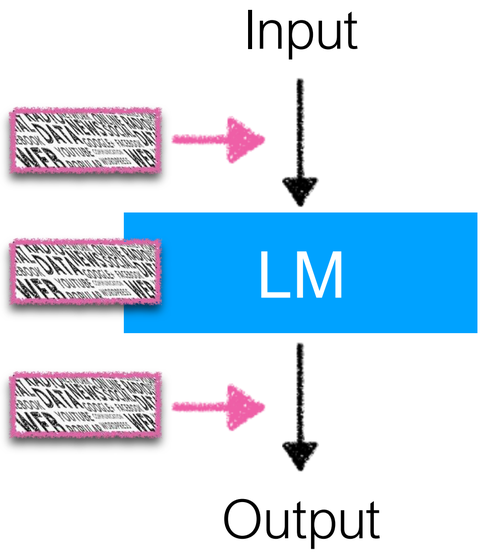
Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



When to retrieve?

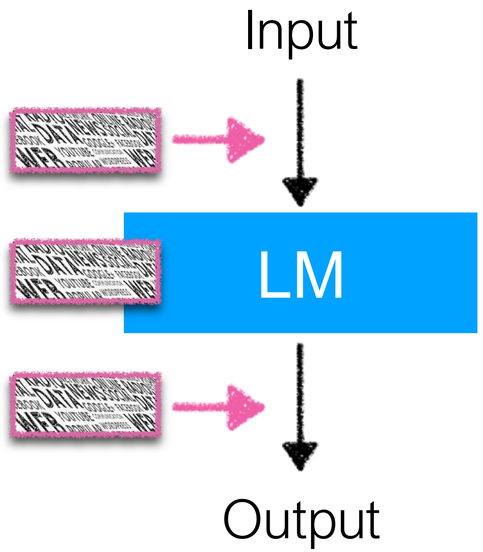
Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



When to retrieve?

Today we focus on
1. What to retrieve
2. How to use retrieval

Three representative architectures

What: Text chunks
How: Input

Input augmentation (RAG)

What: Text chunks
How: Intermediate

Intermediate fusion

What: Tokens
How: Output

Output interpolations

More details?

- Section 3 of our tutorial (https://acl2023-retrieval-lm.github.io/)
- Our position paper (Asai et al., 2024; https://akariasai.github.io/assets/pdf/ralm_position.pdf)

Three representative architectures

What: Text chunks
How: Input

REALM (Guu et al., 2020)

What: Text chunks
How: Intermediate

RETRO (Borgeaud et al., 2021)

What: Tokens
How: Output

kNN-LM (Khandelwal et al., 2020)

More details?

- Section 3 of tutorial (https://acl2023-retrieval-lm.github.io/)
- Position paper (Asai et al., 2024; https://akariasai.github.io/assets/pdf/ralm_position.pdf)

Three representative architectures

What: Text chunks
How: Input

REALM (Guu et al., 2020)

What: Text chunks
How: Intermediate

RETRO (Borgeaud et al., 2021)

What: Tokens
How: Output

kNN-LM (Khandelwal et al., 2020)

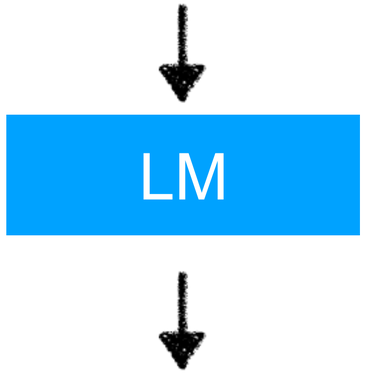
More details?

- Section 3 of our tutorial (https://acl2023-retrieval-lm.github.io/)
- Our position paper (Asai et al., 2024; https://akariasai.github.io/assets/pdf/ralm_position.pdf)

REALM (Guu et al 2020)

x = World Cup 2022 was the last with 32 teams before the increase to **[MASK]** in 2026.

World Cup 2022 was … the increase to **[MASK]** in 2026.



Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

REALM (Guu et al 2020)

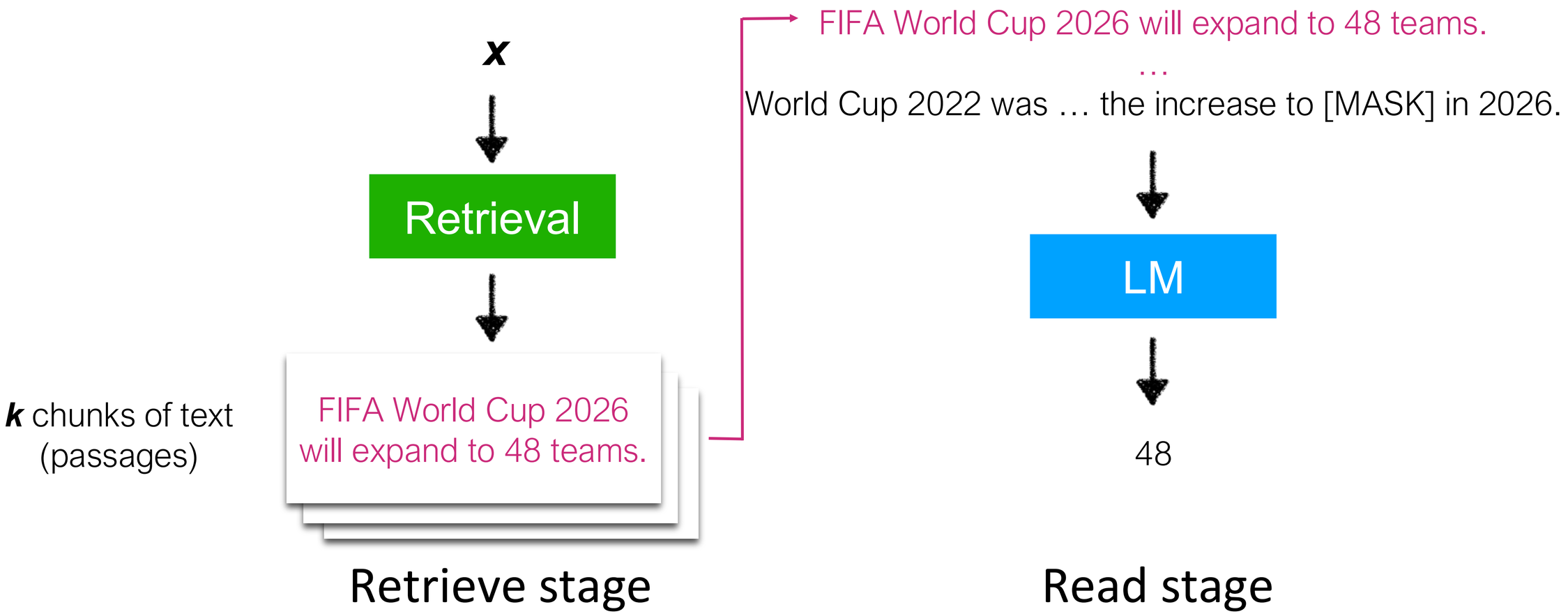
x = World Cup 2022 was the last with 32 teams before the increase to [MASK] in 2026.

k chunks of text (passages)

World Cup 2022 was … the increase to [MASK] in 2026.

REALM (Guu et al 2020)

x = World Cup 2022 was the last before the increase to [MASK] in the 2026 tournament.



Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

REALM: (1) Retrieve stage

$\boldsymbol{x}$ = World Cup 2022 was … the increase to [MASK] in 2026.

FIFA World Cup 2026 will expand to 48 teams.

In 2022, the 32 national teams involved in the tournament.

Team USA celebrated after winning its match against Iran …

Wikipedia
13M chunks (passages)
(called ***documents*** in the paper)

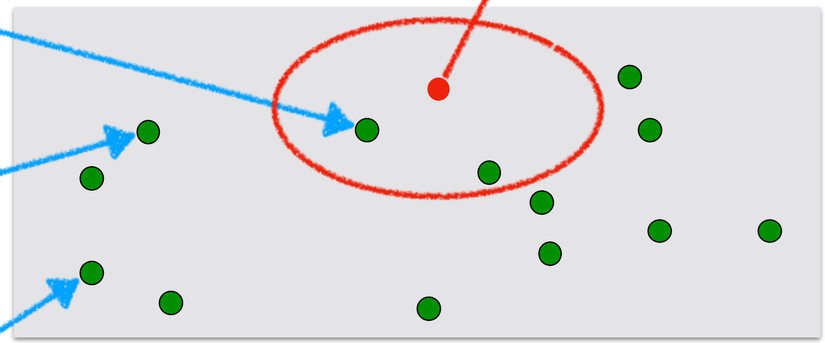
Encoder

Encoder

Encoder

Encoder

Fast nearest neighbor search



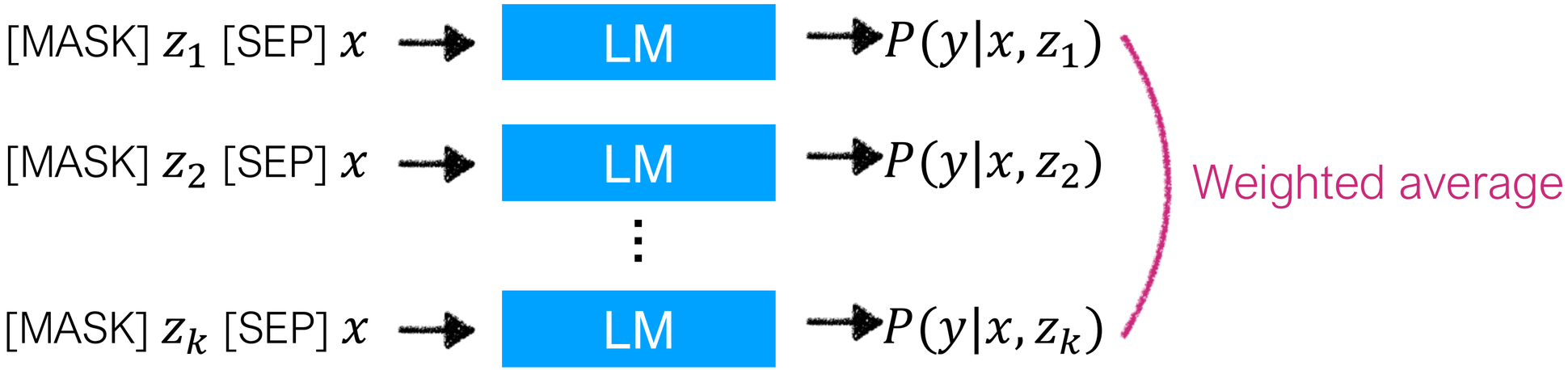
$$\mathbf{z} = \text{Encoder}(z)$$

$$\mathbf{x} = \text{Encoder}(x)$$

$$z_1, \ldots, z_k = \text{argTop}-k(\mathbf{x} \cdot \mathbf{z})$$

k retrieved chunks

REALM: (2) Read stage



$$\sum_{z \in \mathcal{D}} P(z|x) P(y|x, z)$$

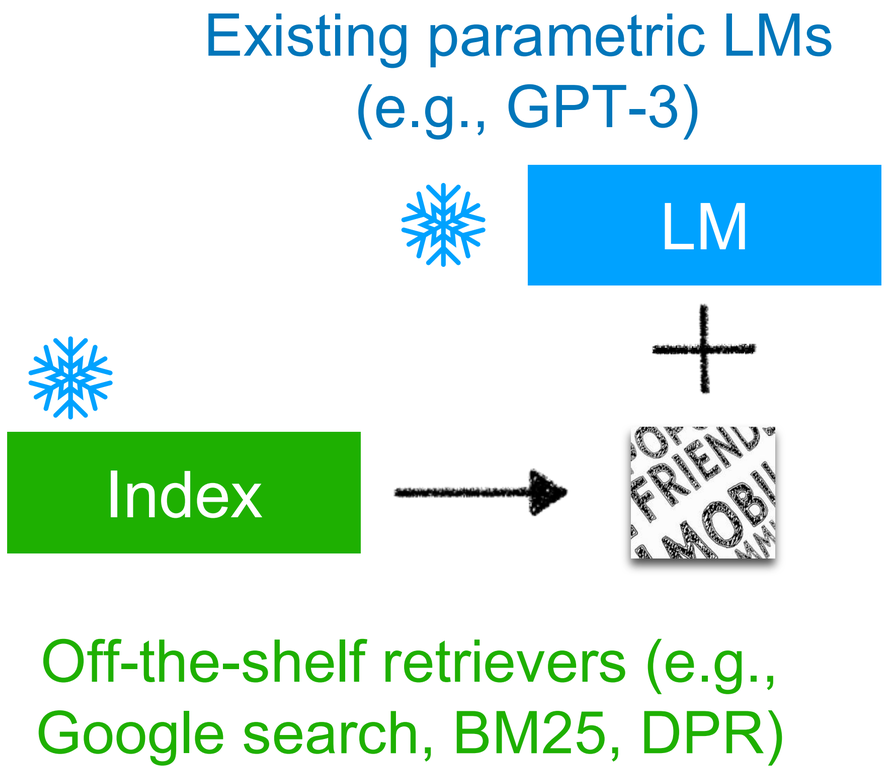
Need to approximate
→ Consider top *k* chunks only

from the retrieve stage

from the read stage

Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

Recent trend: RAG with LLMs



Simply combining existing models w/o training has shown to be successful!

Shi et al. REPLUG: Retrieval-Augmented Black-Box Language Models. Arxiv 2023.
Ram et al. In-Context Retrieval-Augmented Language Models. TACL 2023.

Three representative architectures

What: Text chunks
How: Input

REALM (Guu et al., 2020)

What: Text chunks
How: Intermediate

RETRO (Borgeaud et al., 2021)

What: Tokens
How: Output

kNN-LM (Khandelwal et al., 2020)

More details?

- Section 3 of our tutorial (https://acl2023-retrieval-lm.github.io/)
- Our position paper (Asai et al., 2024; https://akariasai.github.io/assets/pdf/ralm_position.pdf)

RETRO (Borgeaud et al. 2022)

✓ Incorporation in the “intermediate layer” instead of the “input” layer
→ designed for ***many*** chunks, ***frequently***, more ***efficiently***

✓ Scale the datastore (1.8T tokens)

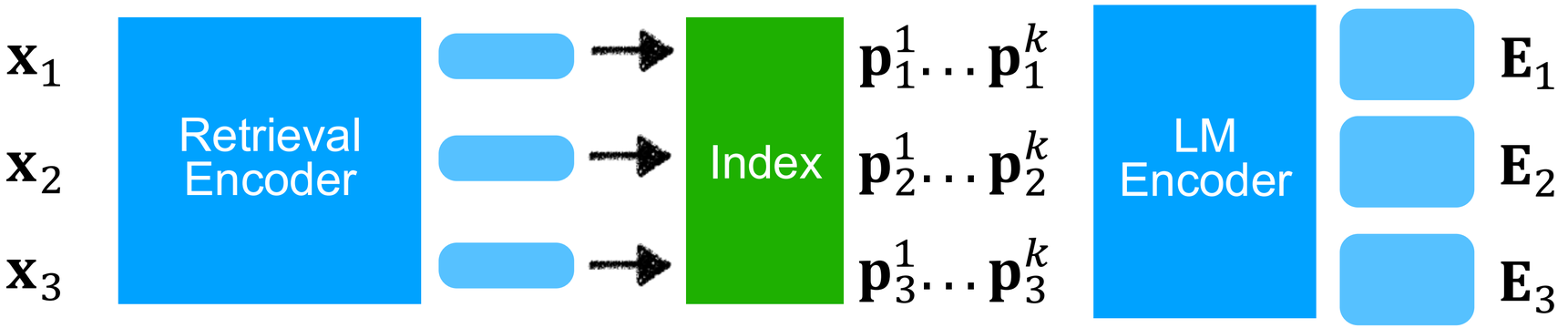
Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

RETRO (Borgeaud et al. 2021)

$\boldsymbol{x}$ = World Cup 2022 was the last with 32 teams, before the increase to

$\mathbf{x}_1$ $\mathbf{x}_2$ $\mathbf{x}_3$

(k chunks of text per split)



(A $r \times k \times d$ matrix)

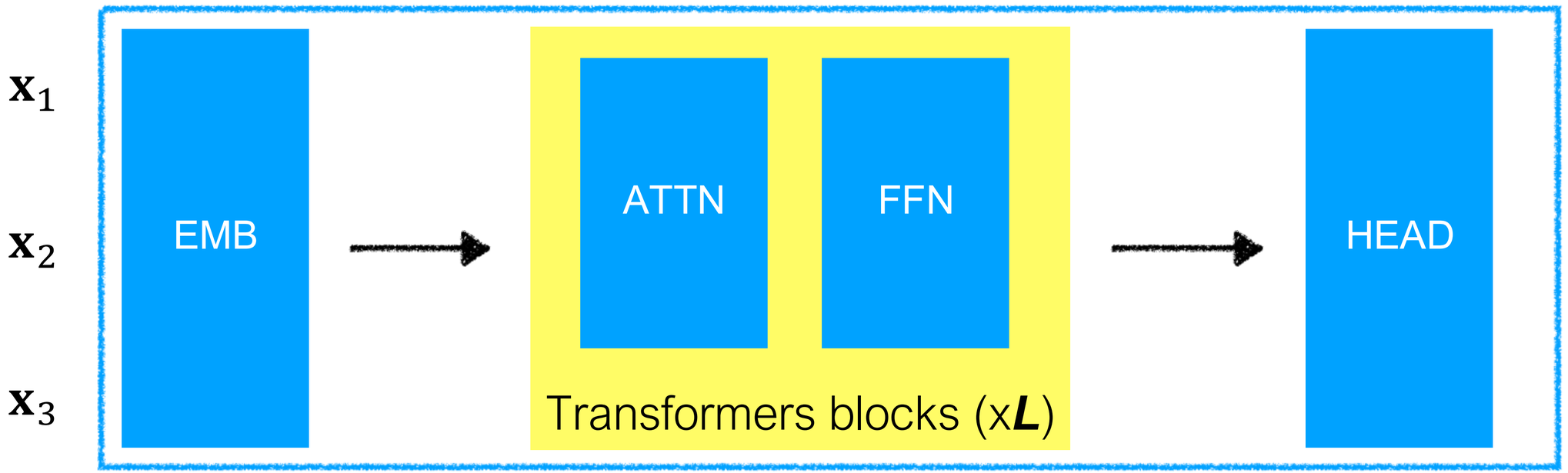
(r =# tokens per text chunk)

(d =hidden dimension)

(k =# retrieved chunks per split)

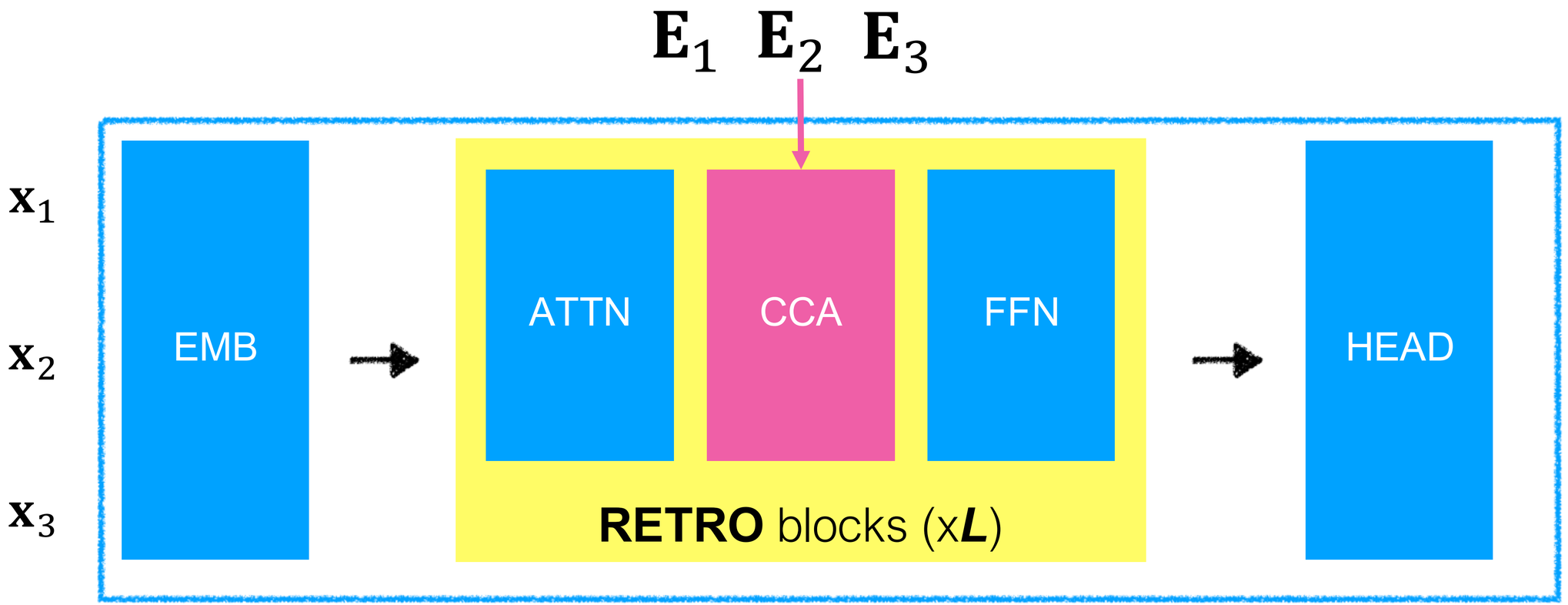
Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

Regular decoder



Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

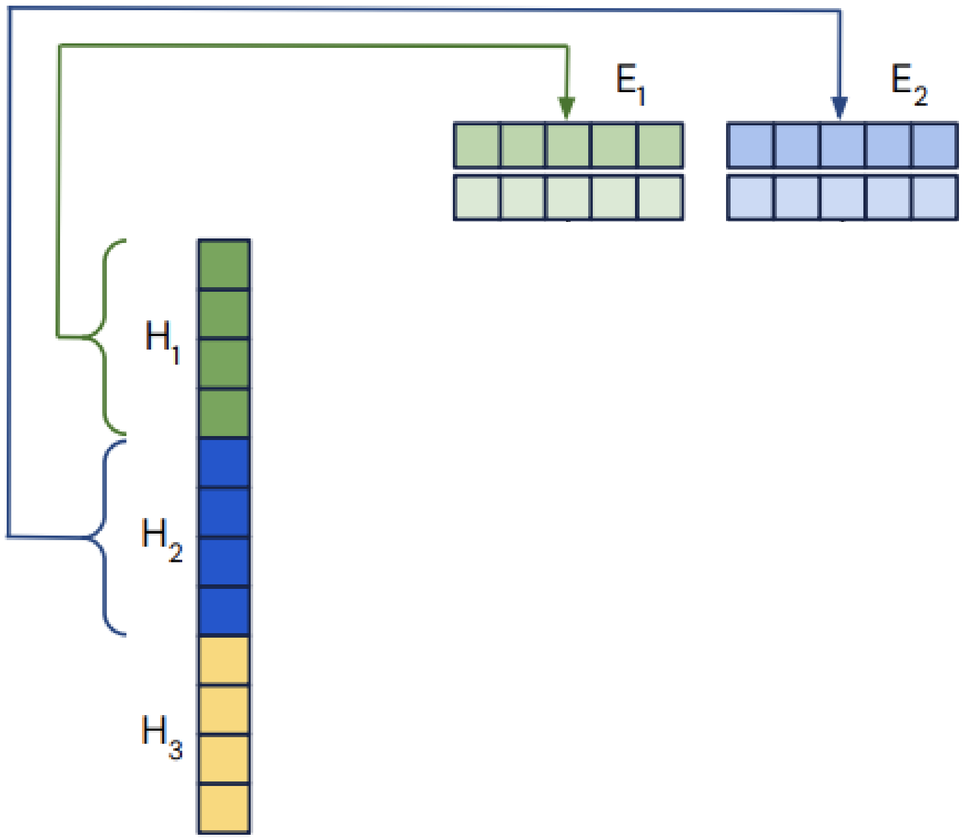
Decoder in RETRO



Chunked Cross Attention (CCA)

Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

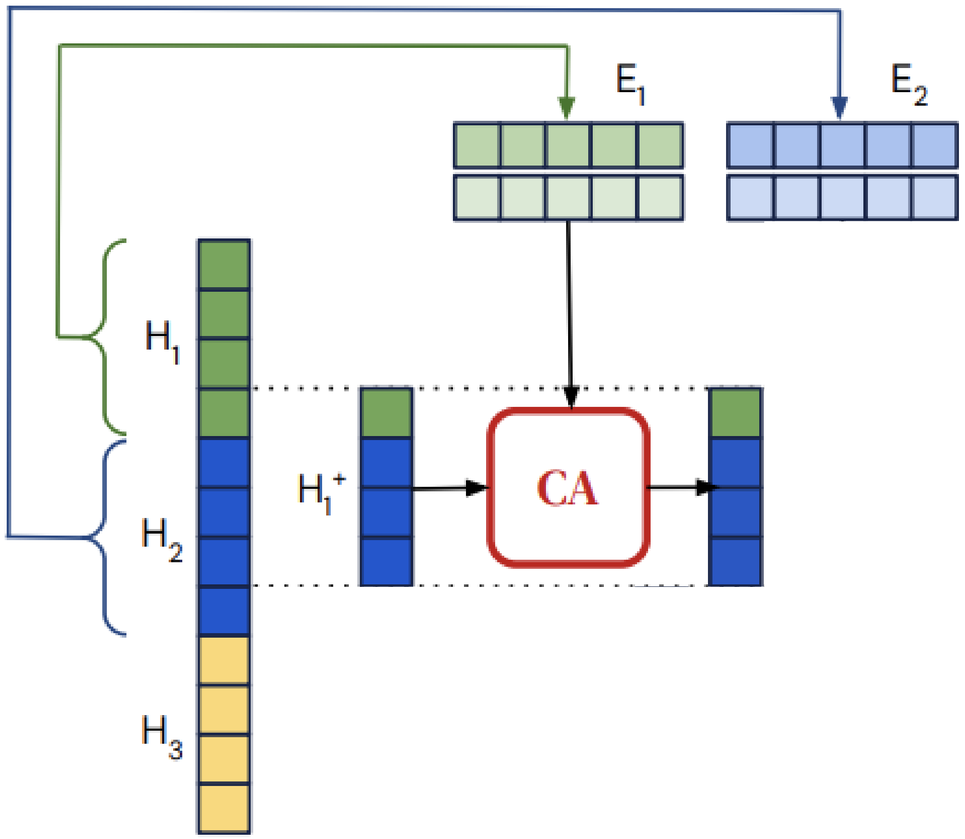
Chunked Cross Attention



Outputs from the previous layer H

Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

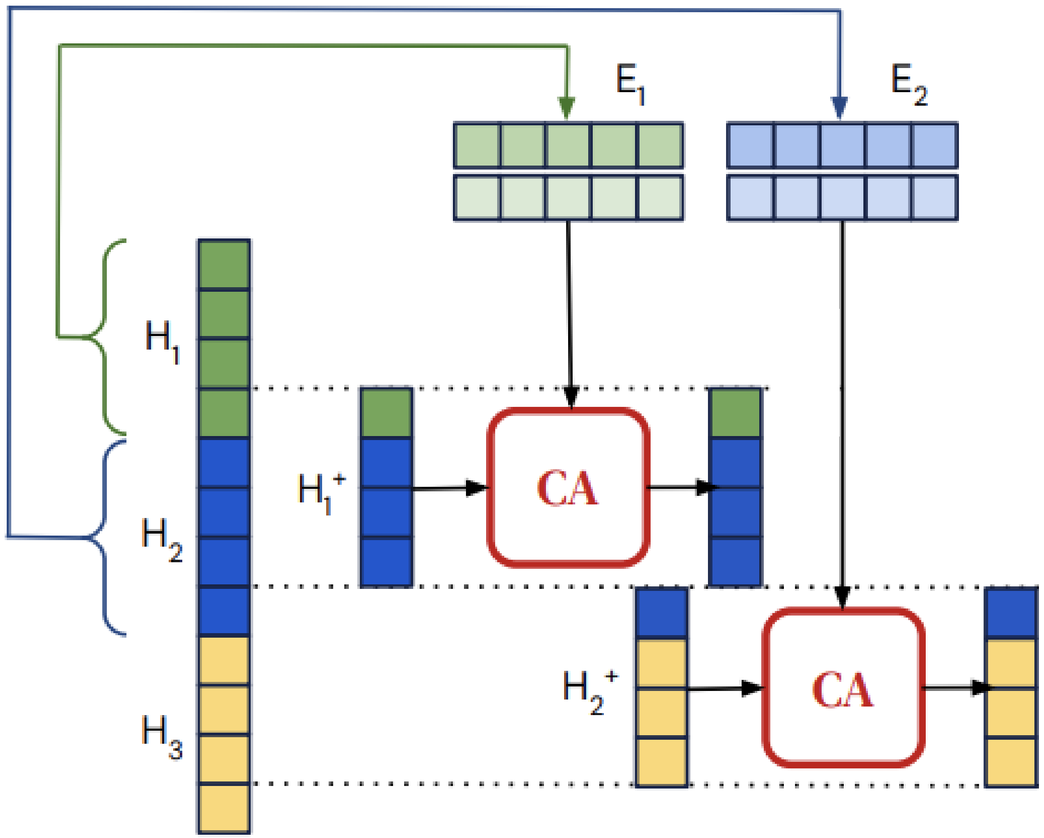
Chunked Cross Attention



Outputs from the previous layer H

Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

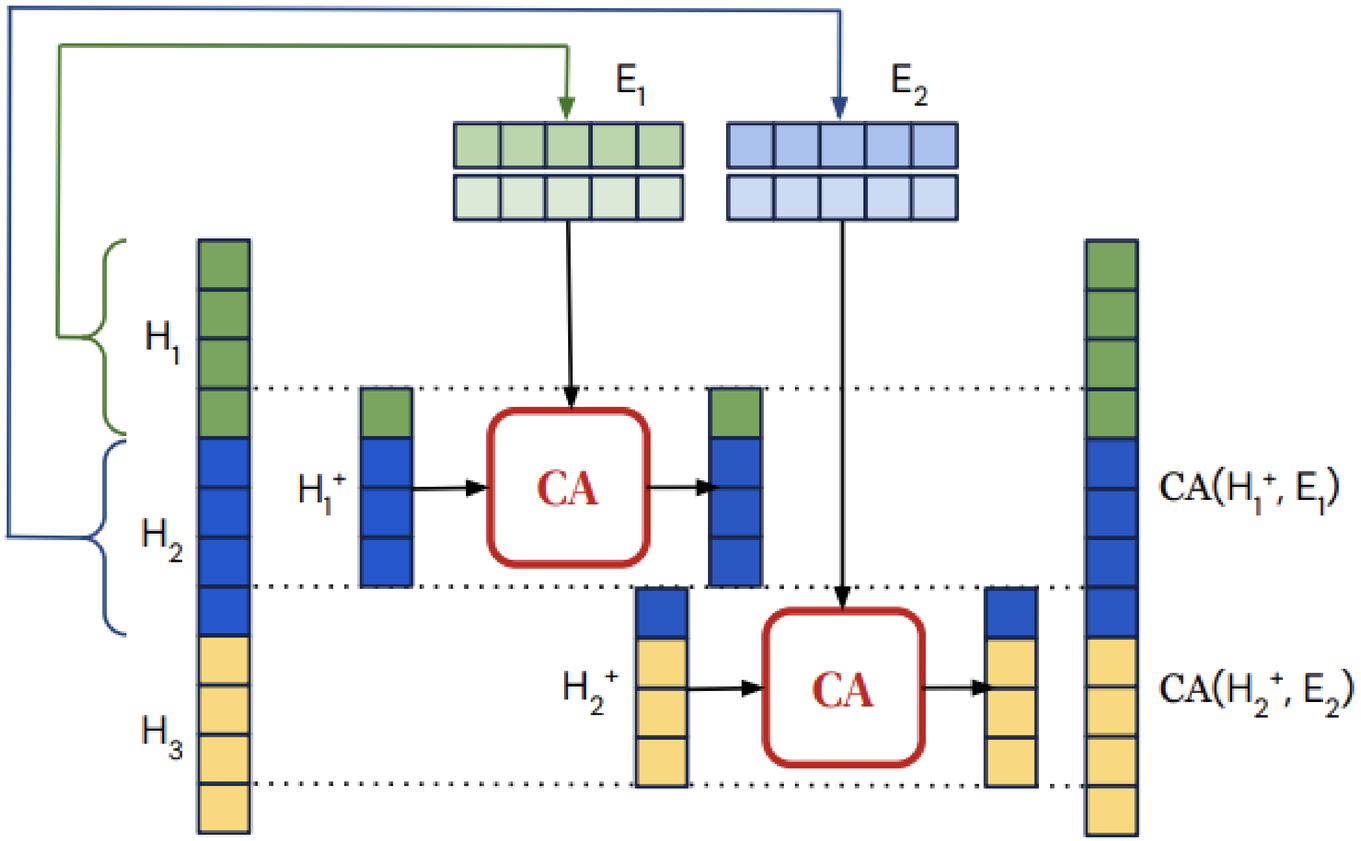
Chunked Cross Attention



Outputs from the previous layer H

Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

Chunked Cross Attention



Outputs from the previous layer H

Inputs to the next layer

Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

Results

Perplexity: The lower the better

Model	Retrieval Set	#Database tokens	#Database keys	Valid	Test
Baseline transformer (ours)	-	-	-	21.53	22.96
*k*NN-LM (ours)	Wikipedia	4B	4B	18.52	19.54
RETRO	Wikipedia	4B	0.06B	18.46	18.97
RETRO	C4	174B	2.9B	12.87	10.23
RETRO	MassiveText (1%)	18B	0.8B	18.92	20.33
RETRO	MassiveText (10%)	179B	4B	13.54	14.95
RETRO	MassiveText (100%)	1792B	28B	**3.21**	**3.92**

RETRO (w/ Wikipedia) outperforms its parametric counterpart

Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

Results

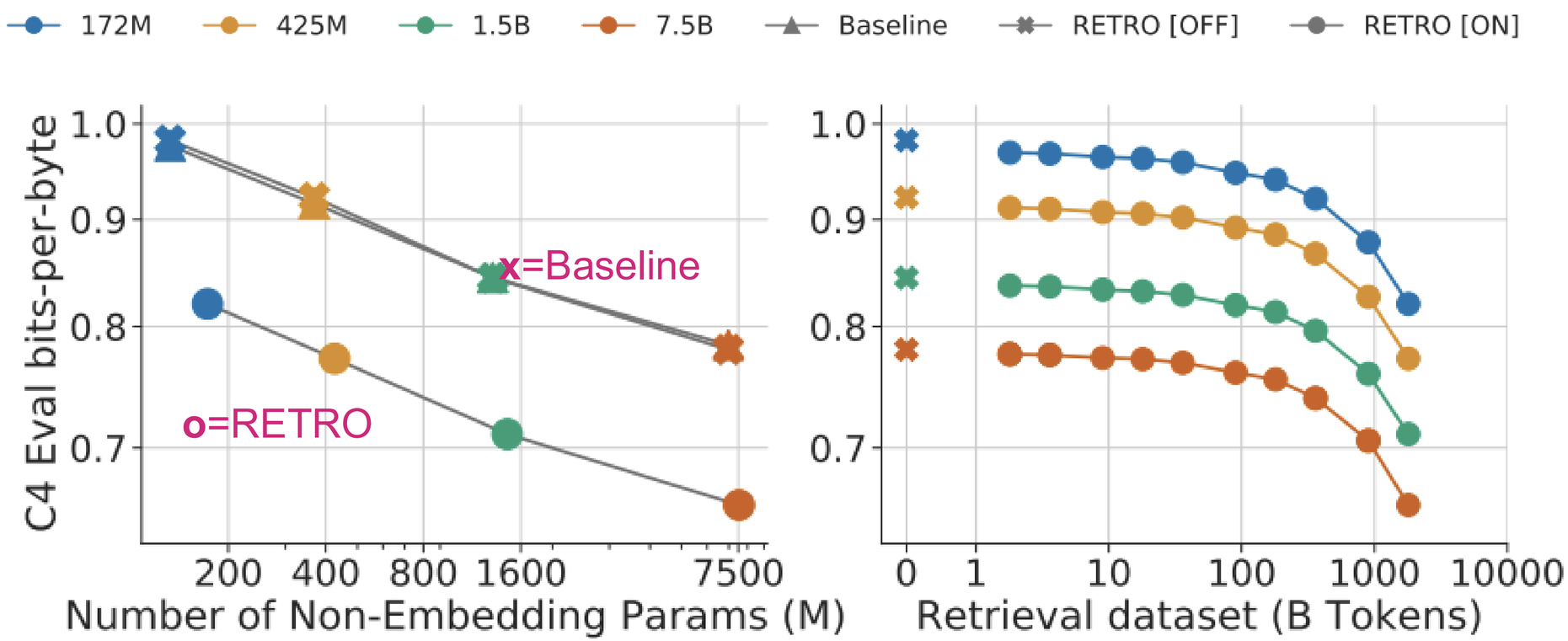
Perplexity: The lower the better

Model	Retrieval Set	#Database tokens	#Database keys	Valid	Test
Adaptive Inputs (Baevski and Auli, 2019)	-	-	-	17.96	18.65
SPALM (Yogatama et al., 2021)	Wikipedia	3B	3B	17.20	17.60
*k*NN-LM (Khandelwal et al., 2020)	Wikipedia	3B	3B	16.06	16.12
Megatron (Shoeybi et al., 2019)	-	-	-	-	10.81
Baseline transformer (ours)	-	-	-	21.53	22.96
*k*NN-LM (ours)	Wikipedia	4B	4B	18.52	19.54
RETRO	Wikipedia	4B	0.06B	18.46	18.97
RETRO	C4	174B	2.9B	12.87	10.23
RETRO	MassiveText (1%)	18B	0.8B	18.92	20.33
RETRO	MassiveText (10%)	179B	4B	13.54	14.95
RETRO	MassiveText (100%)	1792B	28B	**3.21**	**3.92**

RETRO w/ 1.8T datastores achieves SOTA

Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

Results



Gains are constant with model scale

The larger datastore is, the better

Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

Three representative architectures

What: Text chunks
How: Input

REALM (Guu et al., 2020)

What: Text chunks
How: Intermediate

RETRO (Borgeaud et al., 2021)

What: Tokens
How: Output

kNN-LM (Khandelwal et al., 2020)

More details?

- Section 3 of our tutorial (https://acl2023-retrieval-lm.github.io/)
- Our position paper (Asai et al., 2024; https://akariasai.github.io/assets/pdf/ralm_position.pdf)

kNN-LM (Khandelwal et al. 2020)

✓ A different way of using retrieval, where the LM outputs a nonparametric distribution over every token in the data.

✓ Can be seen as an incorporation in the “output” layer

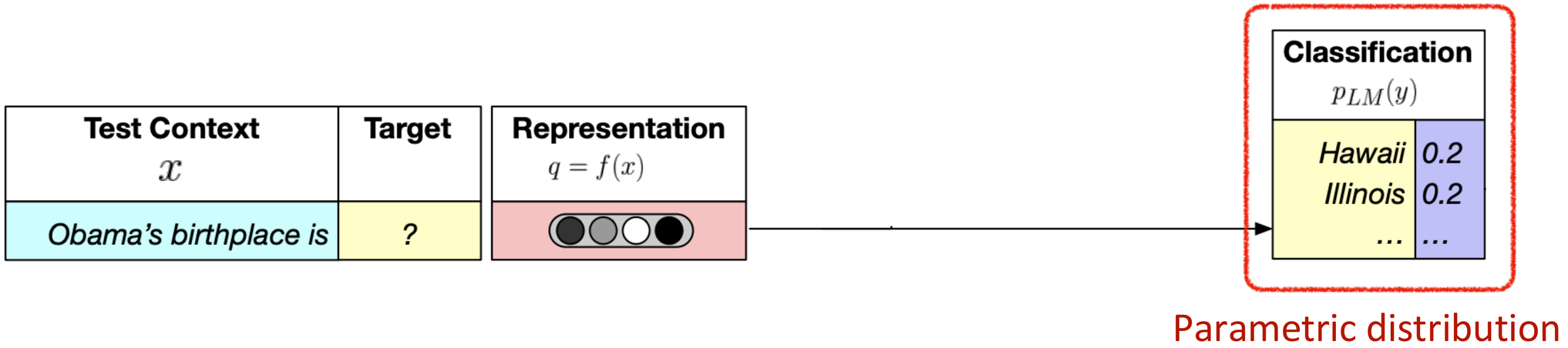
Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

kNN-LM (Khandelwal et al. 2020)

Test Context x	Target
Obama's birthplace is	*?*

Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

kNN-LM (Khandelwal et al. 2020)

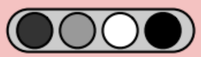


Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

kNN-LM (Khandelwal et al. 2020)

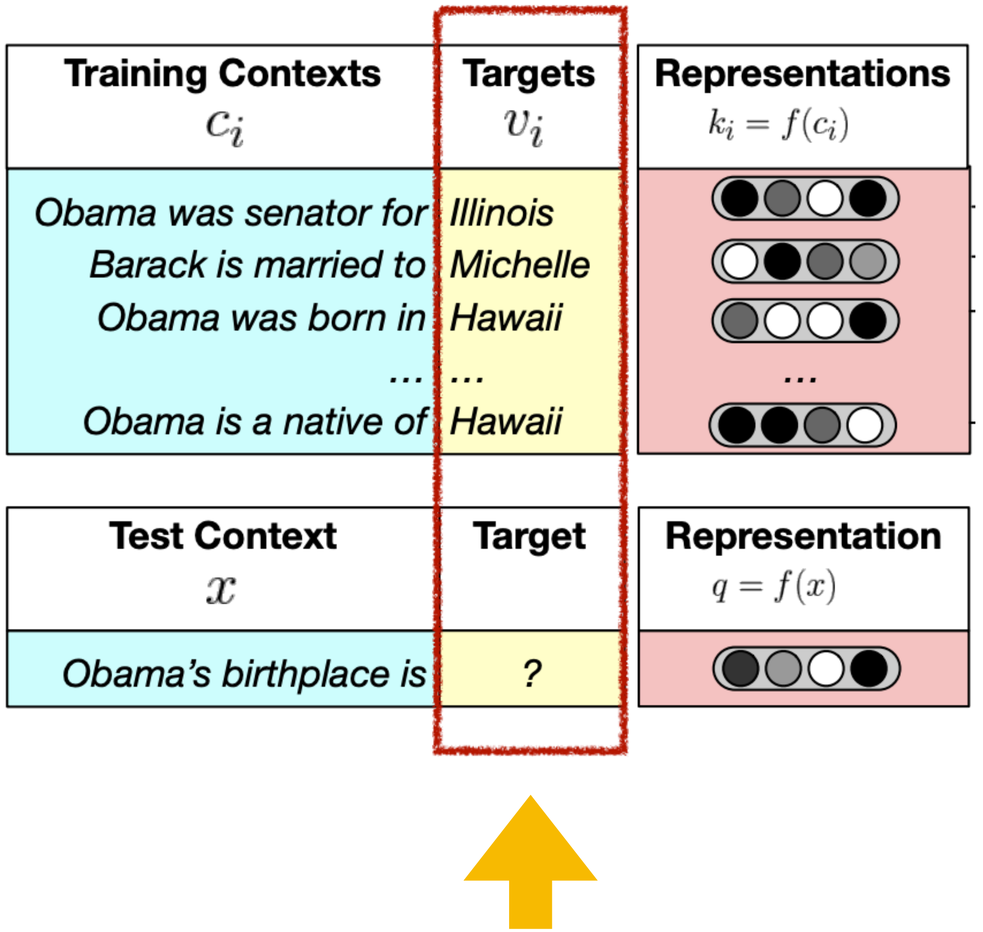
Training Contexts c_i	Targets v_i
Obama was senator for	*Illinois*
Barack is married to	*Michelle*
Obama was born in	*Hawaii*
...	*...*
Obama is a native of	*Hawaii*

… Obama was senator for Illinois from 1997 to 2005, ….
Barack is Married to Michelle and their first daughter, …
Obama was born in Hawaii, and graduated from Columbia University. … Obama is a native of Hawaii, ….

Test Context x	Target	Representation $q = f(x)$
Obama's birthplace is	*?*	

Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

kNN-LM (Khandelwal et al. 2020)

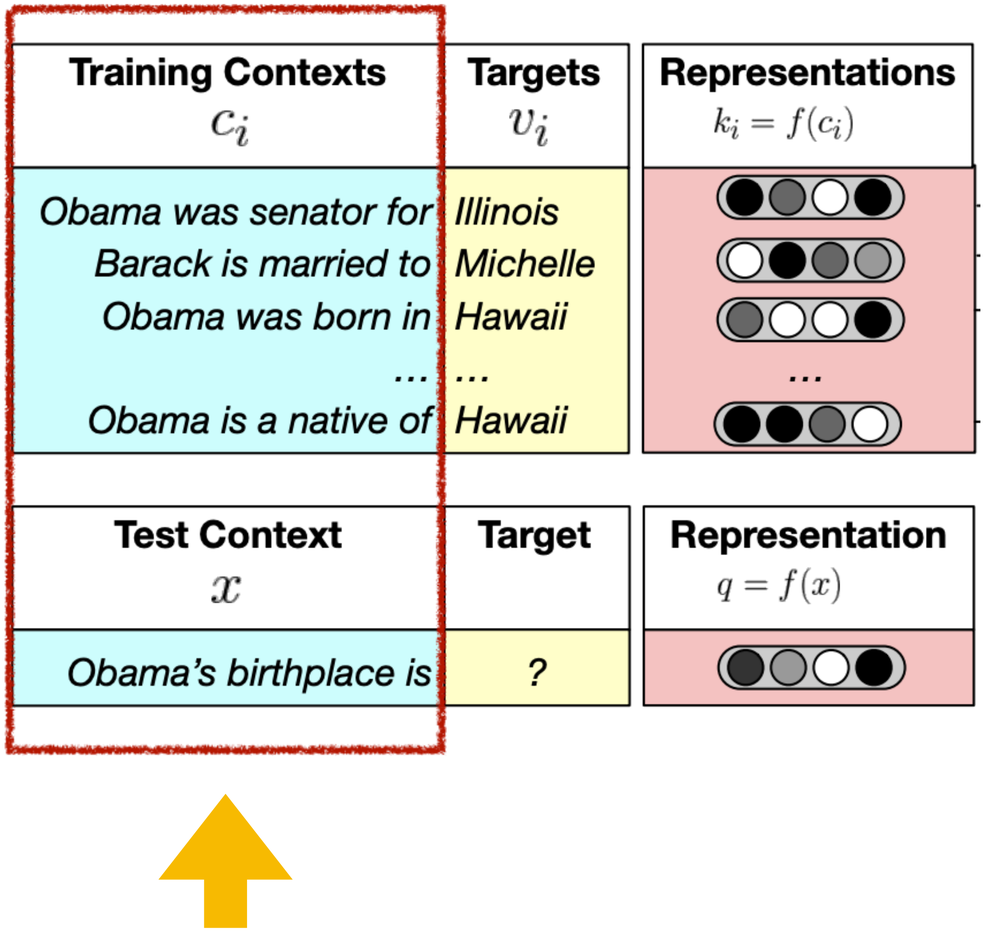


Which tokens in a datastore are close to the next token?

Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

kNN-LM (Khandelwal et al. 2020)

The size of the datastore = # of tokens in the corpus (>1B)



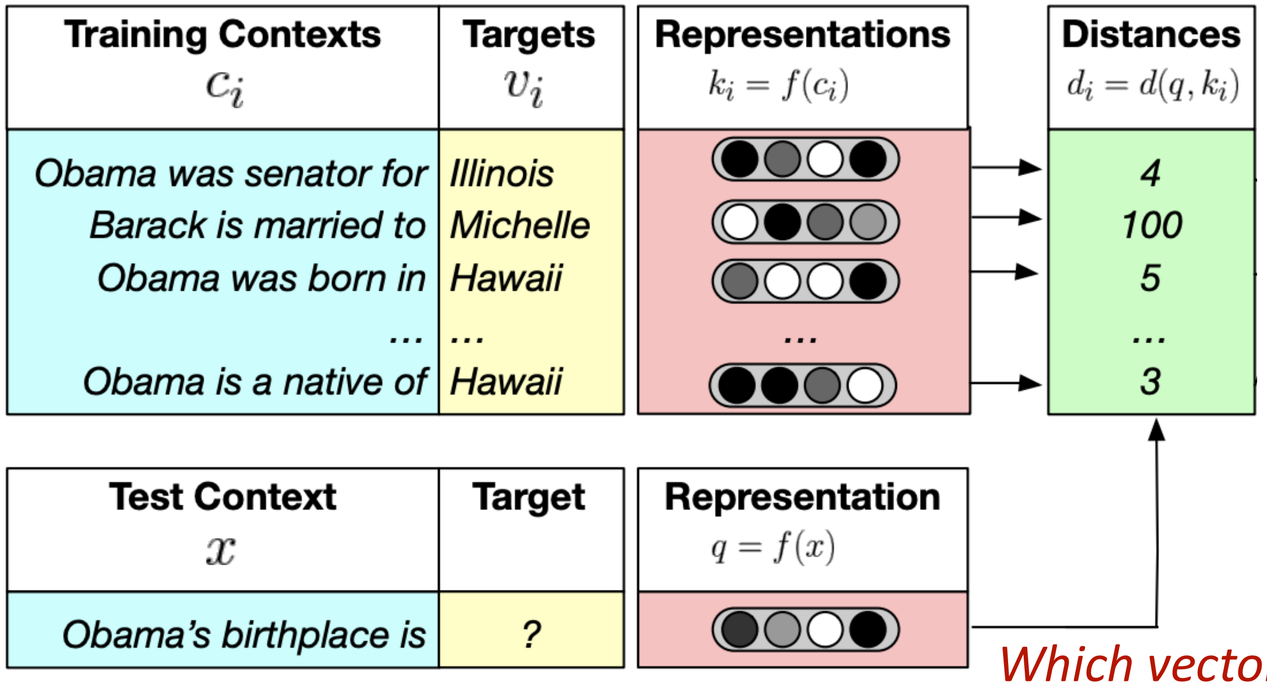
Which tokens <u>in a datastore</u> are close to the next token?

=

Which prefixes in a datastore are close to the prefix we have?

Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

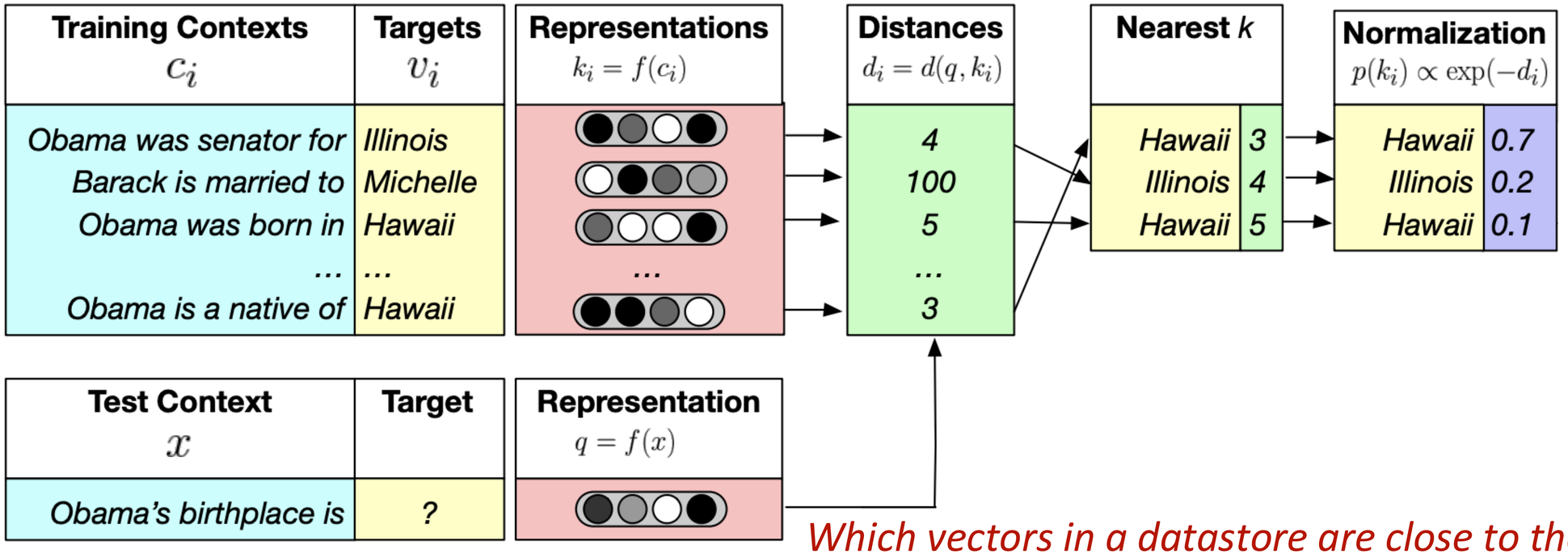
kNN-LM (Khandelwal et al. 2020)



Which vectors in a datastore are close to the vector we have?

Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

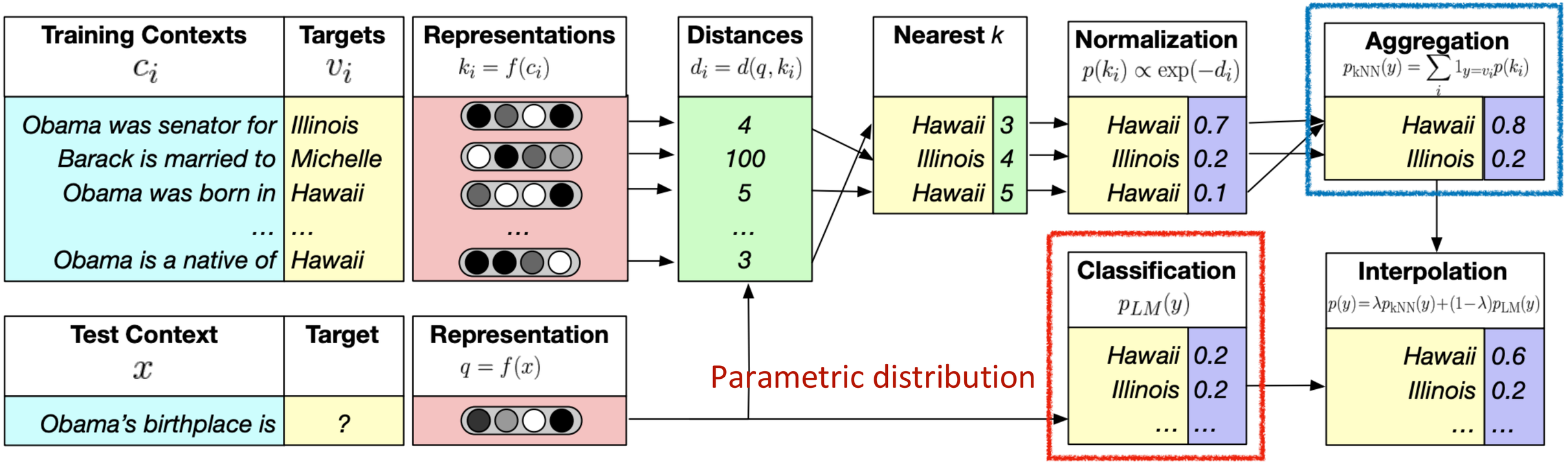
kNN-LM (Khandelwal et al. 2020)



Which vectors in a datastore are close to the vector we have?

Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

kNN-LM (Khandelwal et al. 2020)



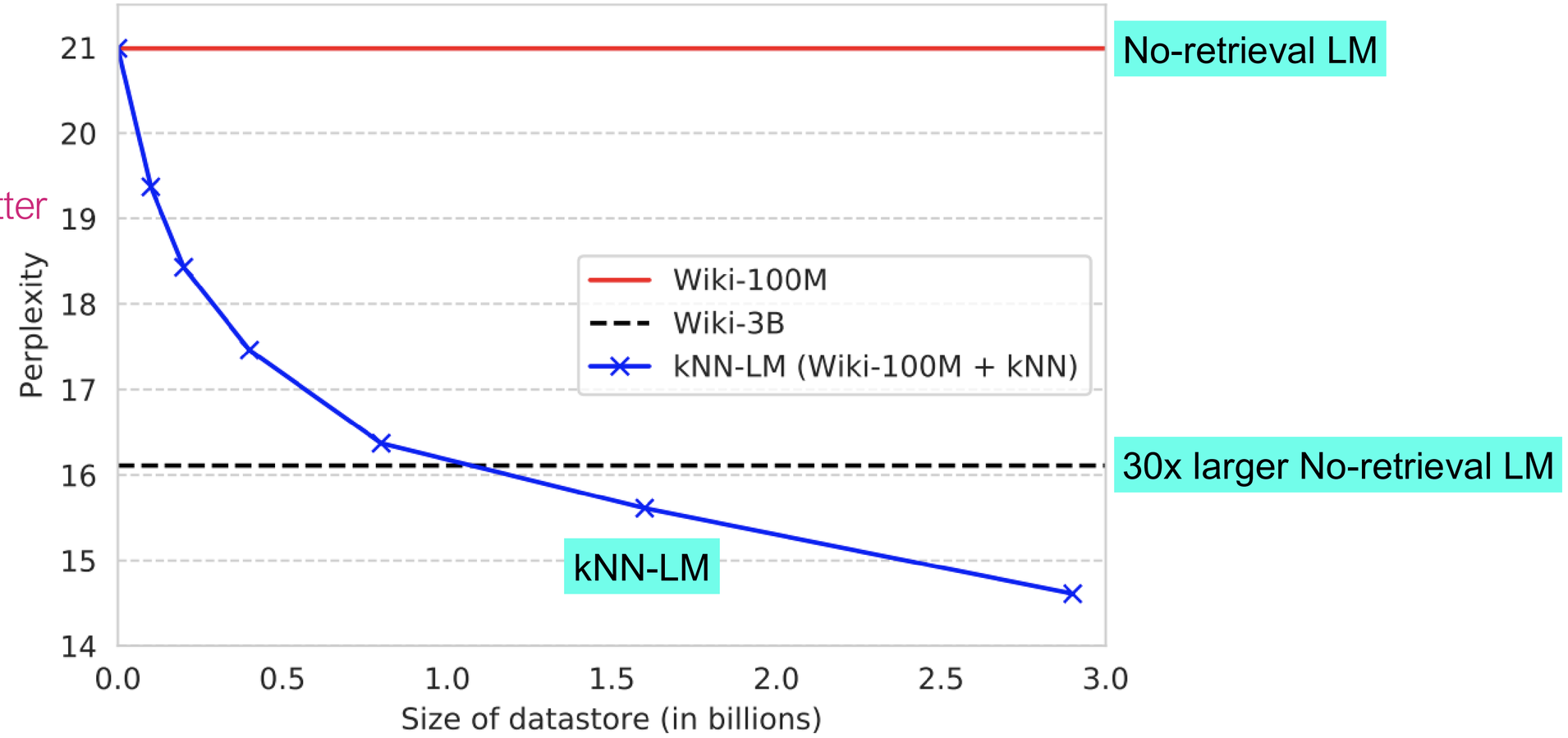
λ: hyperparameter

$$P_{k\text{NN-LM}}(y|x) = (1-\lambda)P_{\text{LM}}(y|x) + \lambda P_{k\text{NN}}(y|x)$$

Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

kNN-LM - results

The lower the better



Outperforms no-retrieval LM

Better with bigger datastore

Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

Three representative architectures

What: Text chunks
How: Input

REALM (Guu et al., 2020)

What: Text chunks
How: Intermediate

RETRO (Borgeaud et al., 2021)

What: Tokens
How: Output

kNN-LM (Khandelwal et al., 2020)

More details?

- Section 3 of tutorial (https://acl2023-retrieval-lm.github.io/)
- Position paper (Asai et al., 2024; https://akariasai.github.io/assets/pdf/ralm_position.pdf)

Today's outline

Question:
https://bit.ly/akari_ralm_lec

Scan me

Why do we need retrieval-augmented LMs?

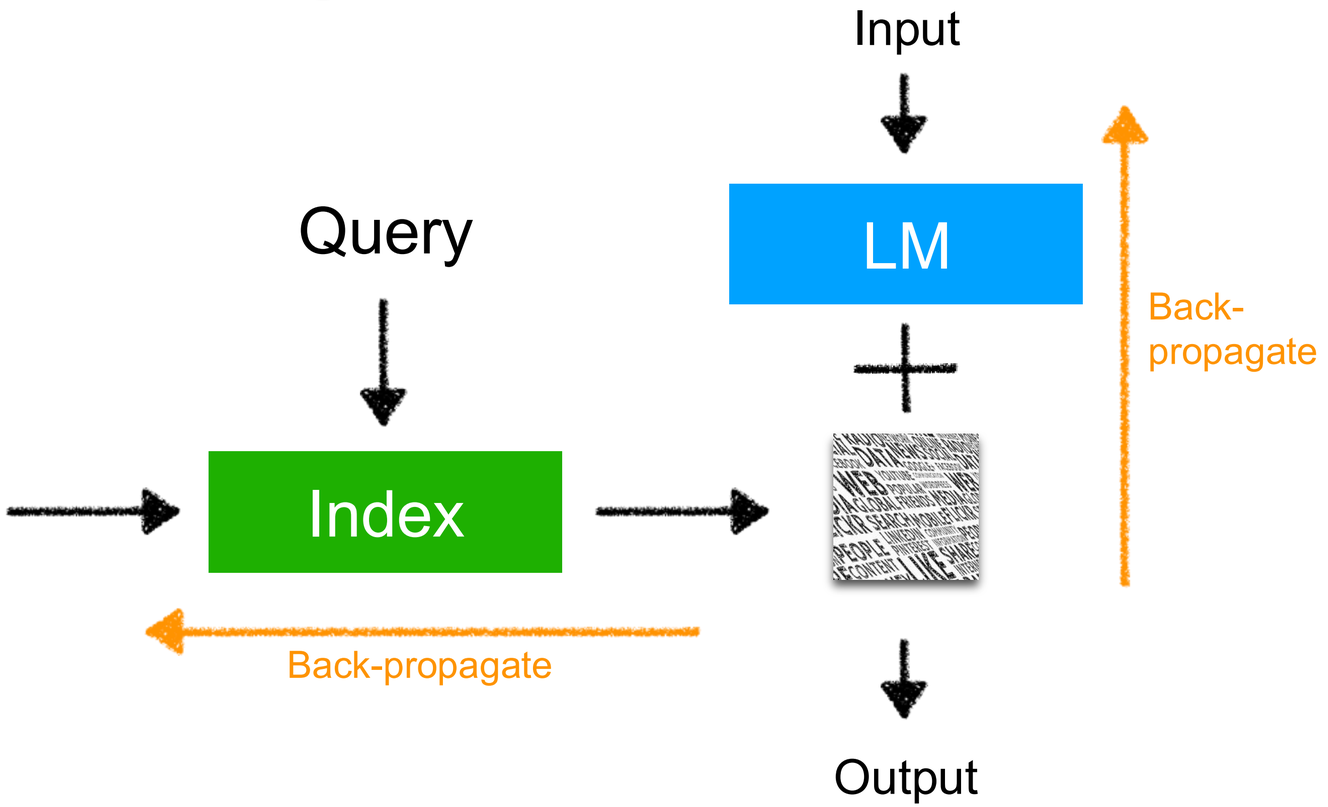
Architectures of retrieval-augmented LMs (Inference)

Training of retrieval-augmented LMs

Limitations and future directions

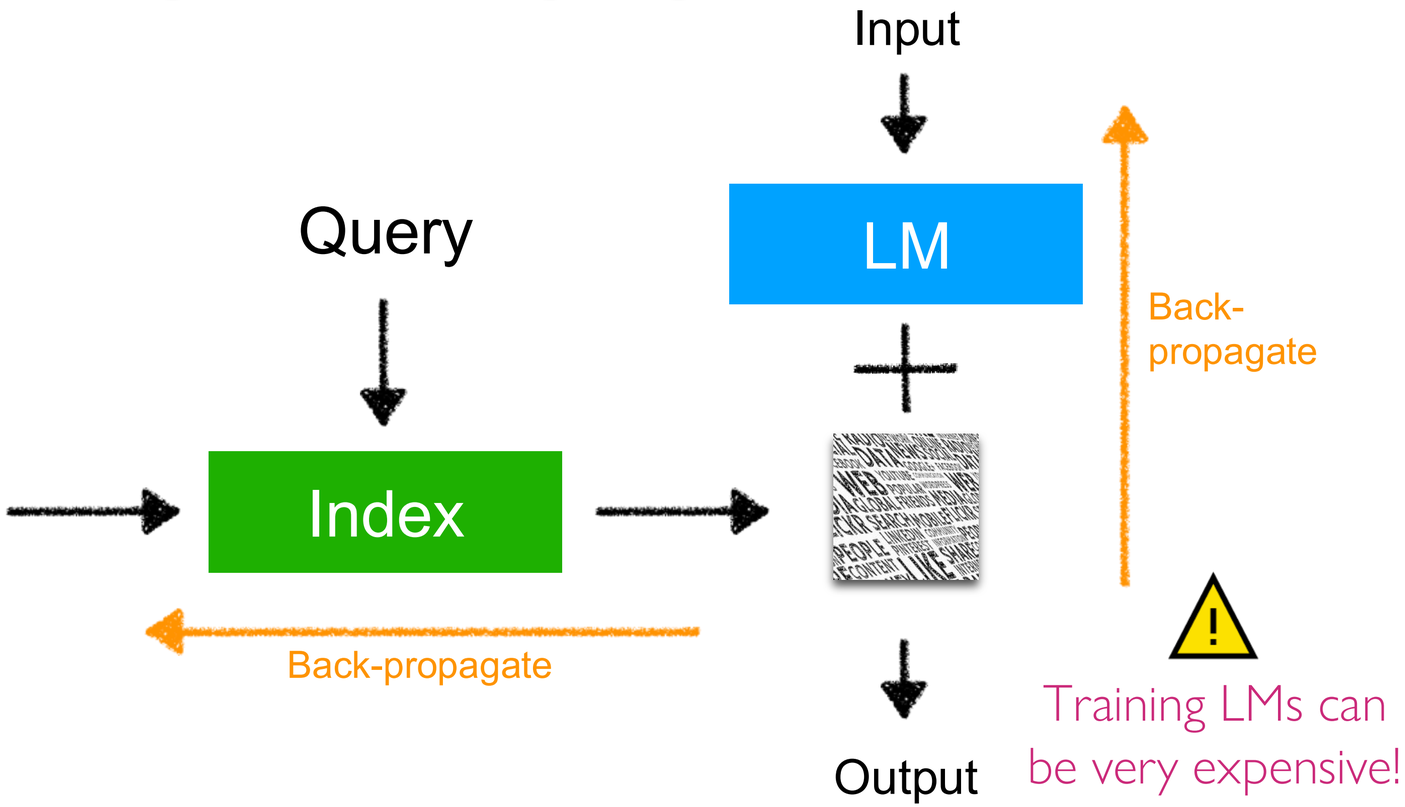
Training retrieval-augmented LMs

Datastore

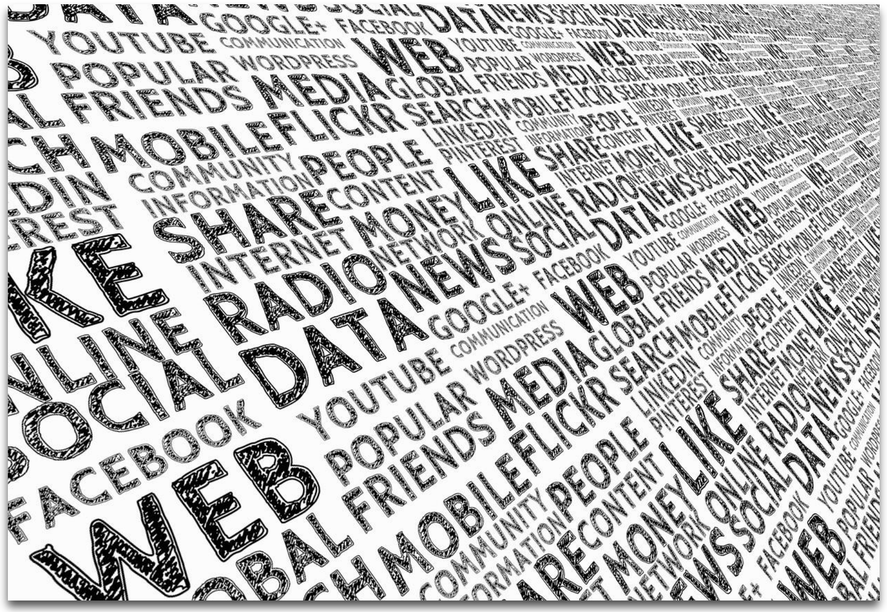


Why is training challenging?

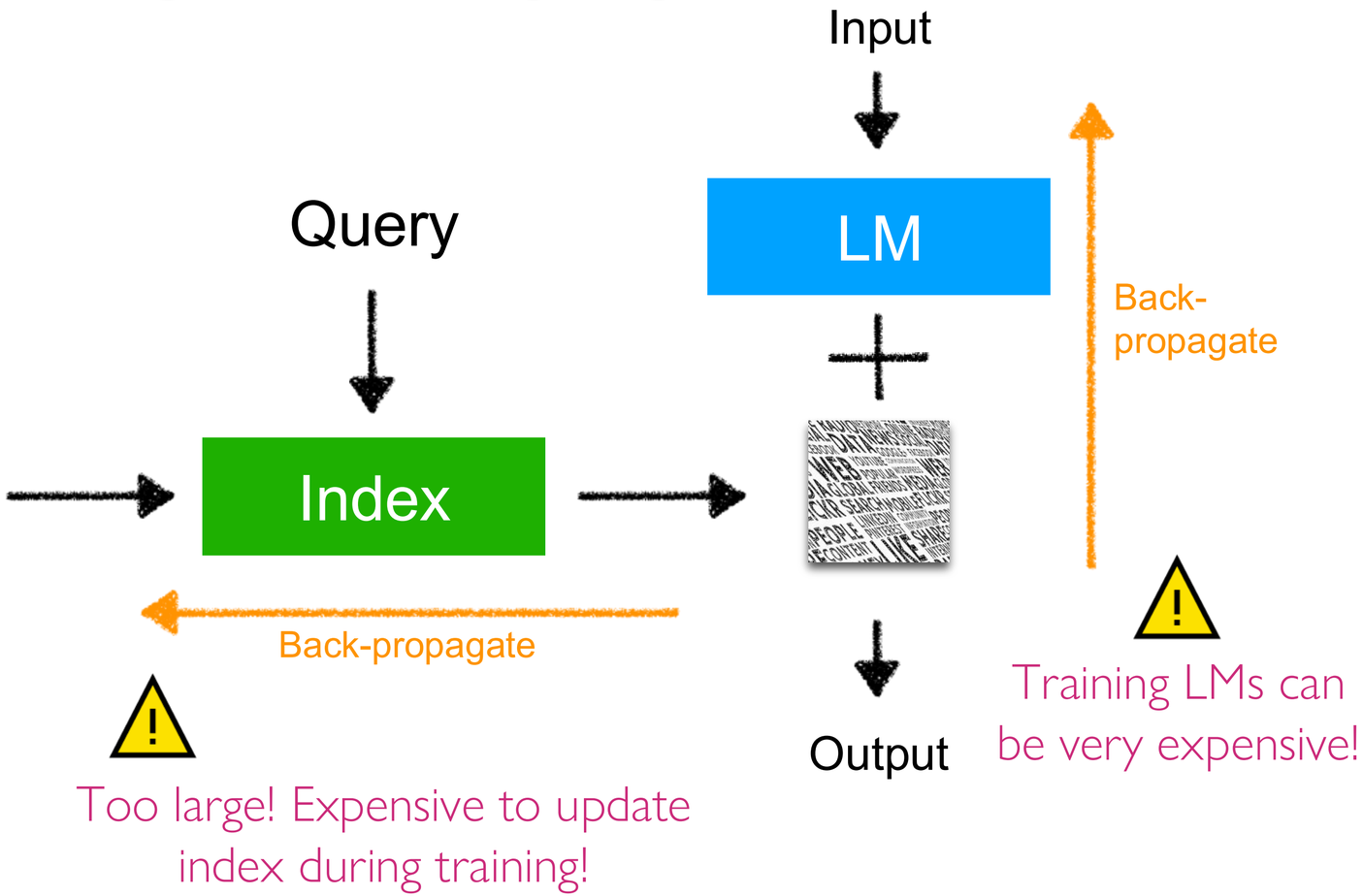
Datastore



Why is training challenging?



Datastore



Challenges of updating retrieval models

Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_3$

…

We may encode a lot of (>100M) text chunks using the encoder!

Challenges of updating retrieval models

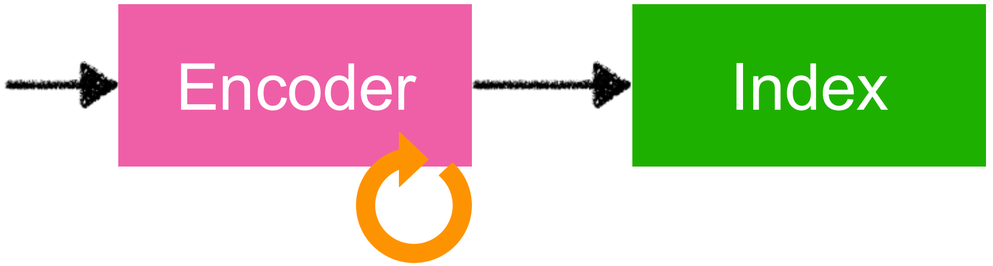
Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_3$

…



During training, we will update the encoder

Challenges of updating retrieval models

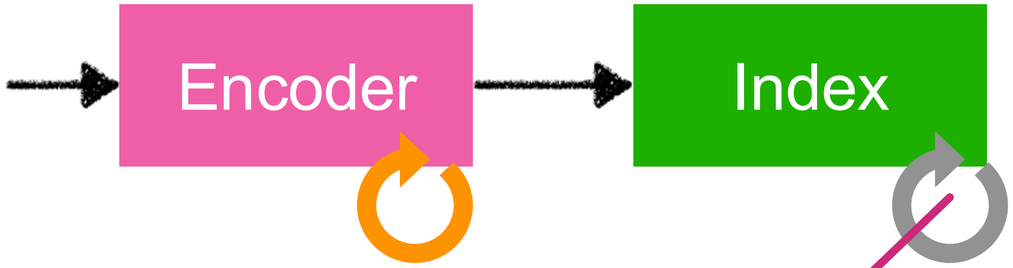
Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_3$

…



Re-indexing will be very expensive!

Training methods for retrieval-augmented LMs

- Independent training
- Sequential training
- Joint training w/ asynchronous index update
- Joint training w/ in-batch approximation

Training methods for retrieval-augmented LMs

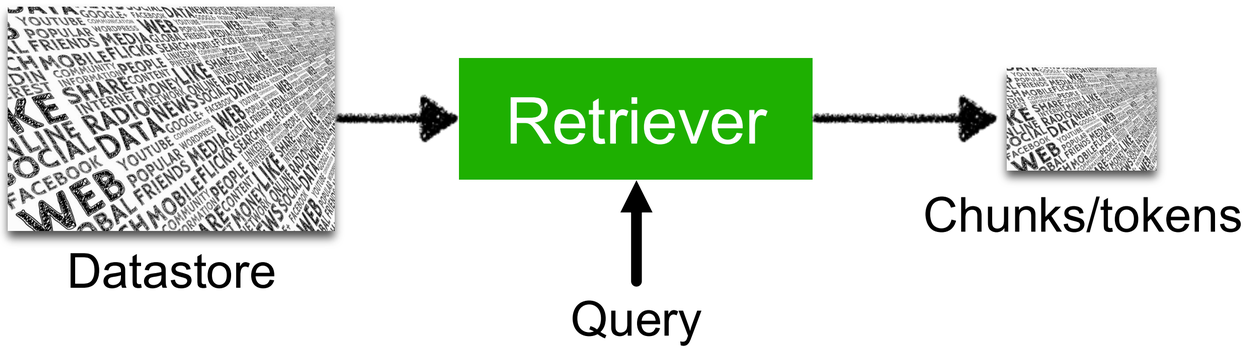
- **Independent training**
- Sequential training
- Joint training w/ asynchronous index update
- Joint training w/ in-batch approximation

Independent training

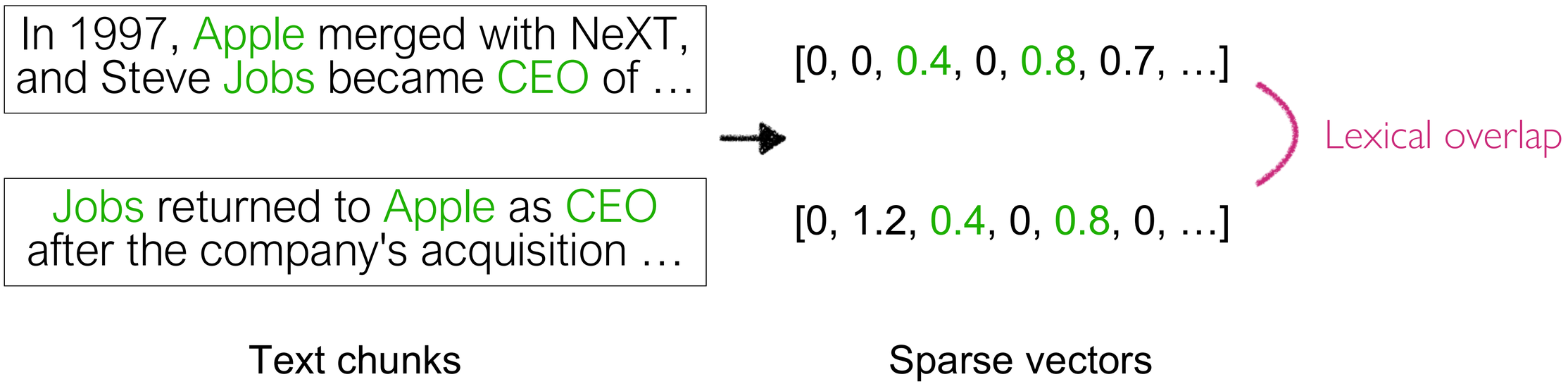
Retrieval models and language models are trained **independently**

- Training language models

- Training retrieval models



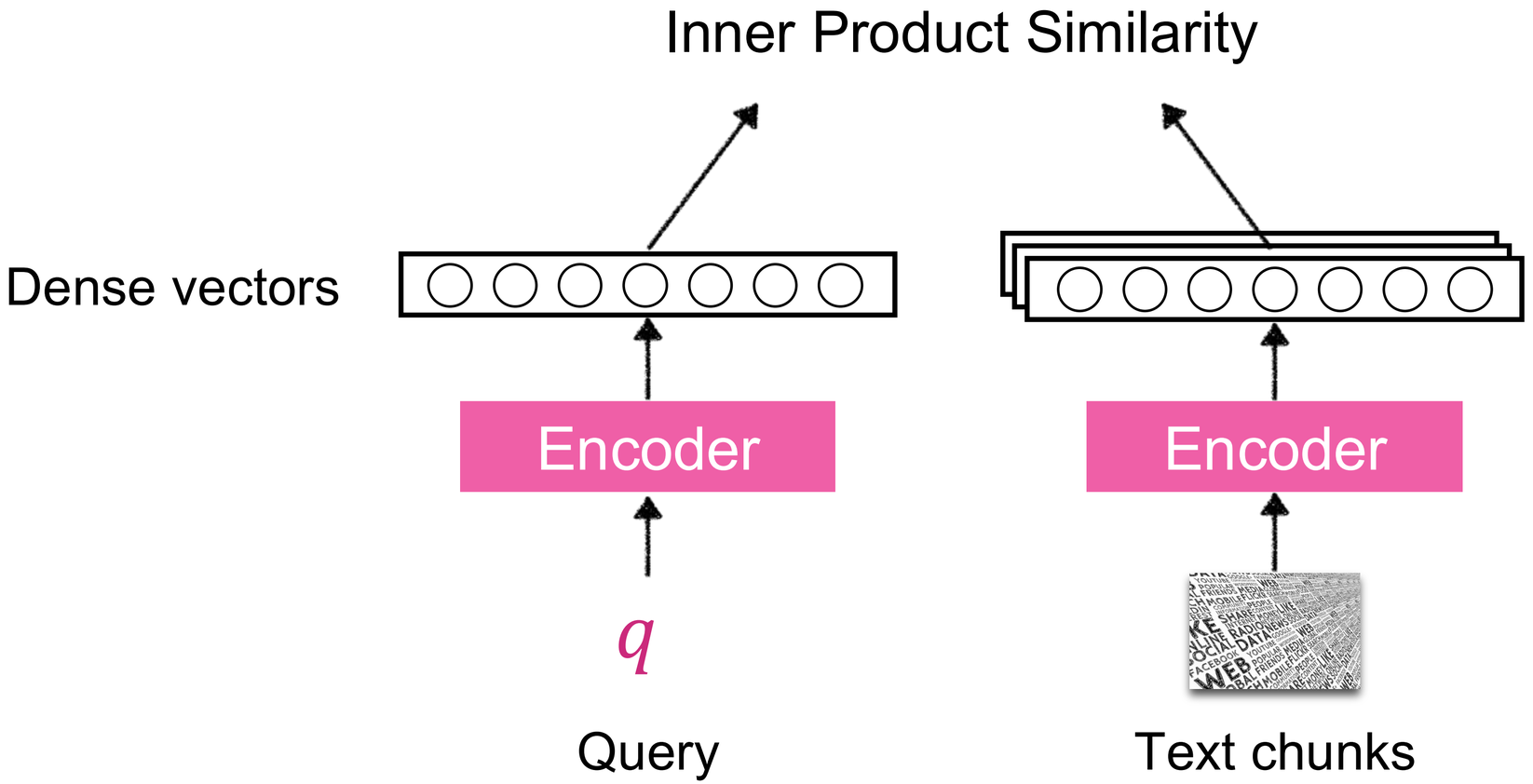
Sparse retrieval models: TF-IDF / BM25



No training needed!

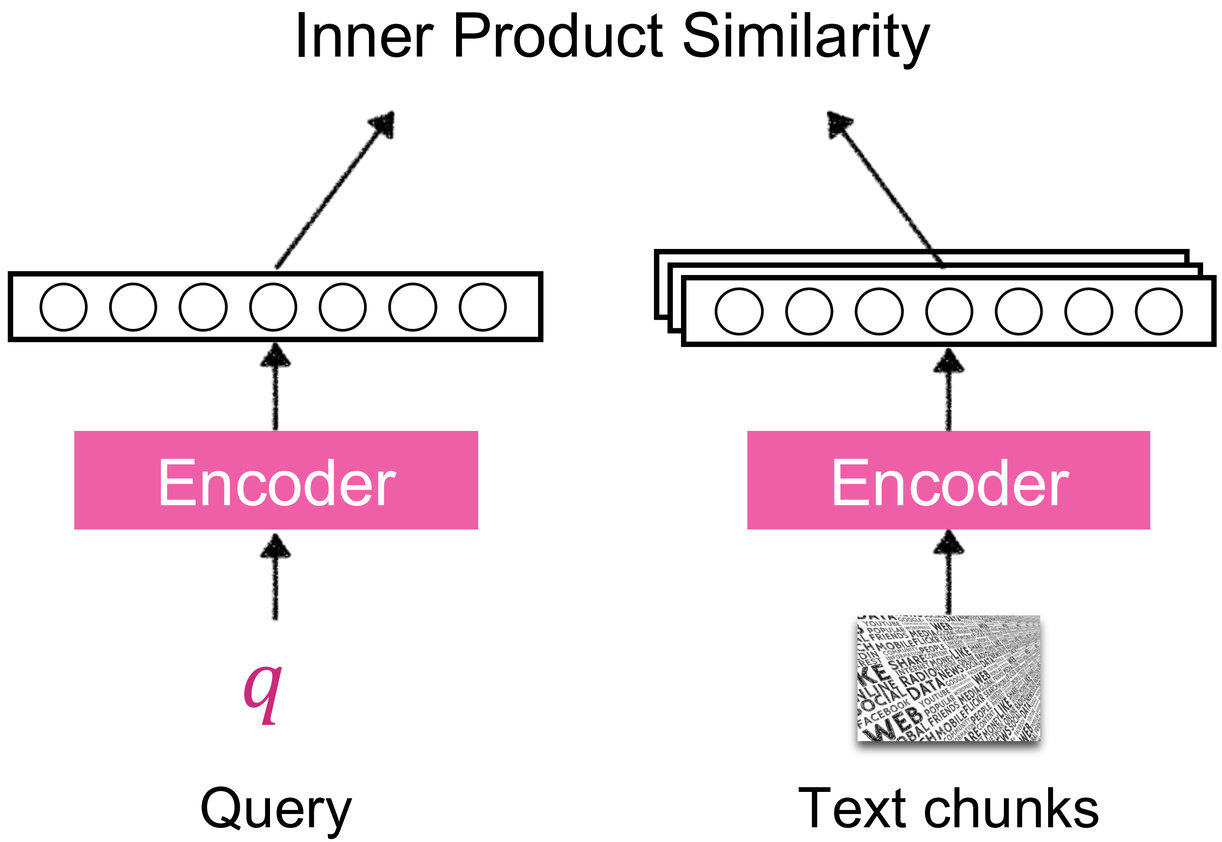
Ramos. Using TF-IDF to Determine Word Relevance in Document Queries. 2023.
Robertson and Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond. Foundations and Trends in Information Retrieval 2009.

Dense retrieval models: DPR (Karpukhin et al. 2020)



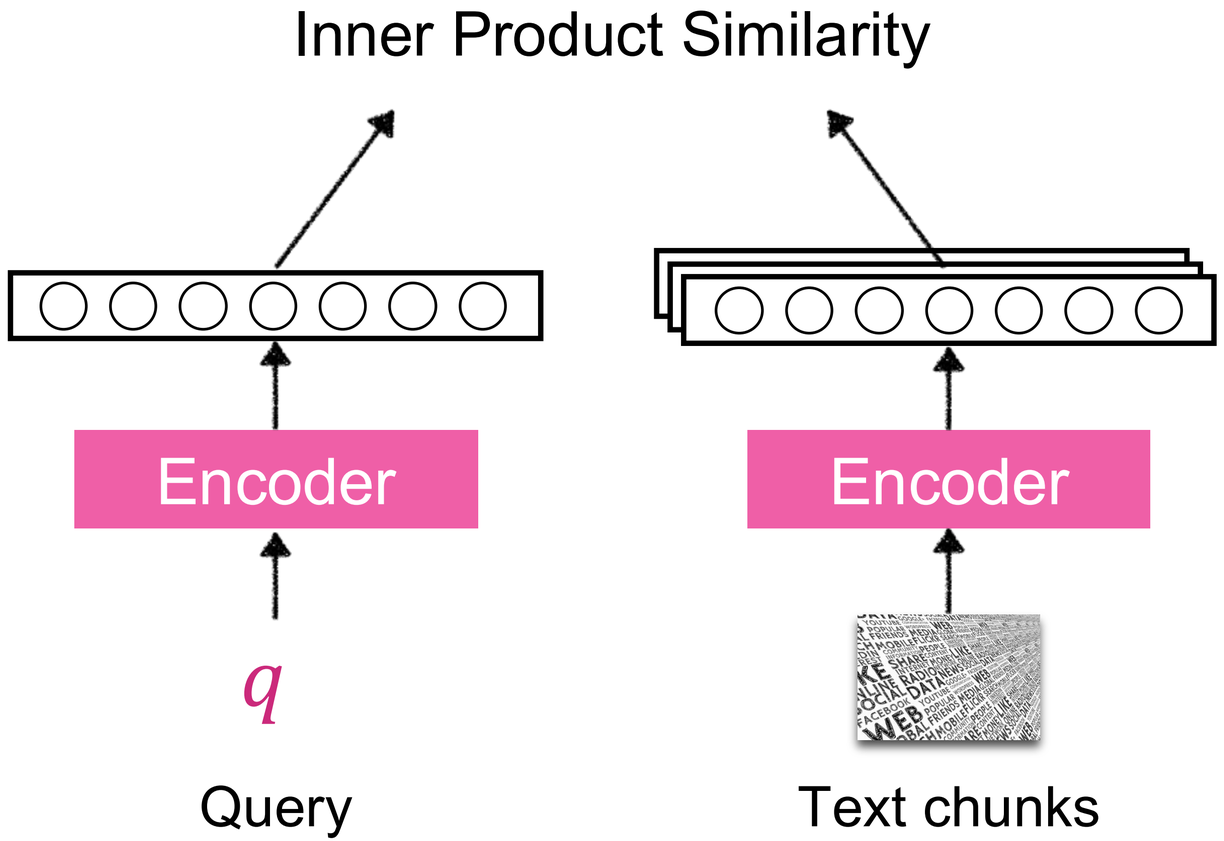
Karpukhin et al. Dense Passage Retrieval for Open-Domain Question Answering. EMNLP 2020.

Training dense retrieval models: DPR



Karpukhin et al. Dense Passage Retrieval for Open-Domain Question Answering. EMNLP 2020.

Training dense retrieval models: DPR

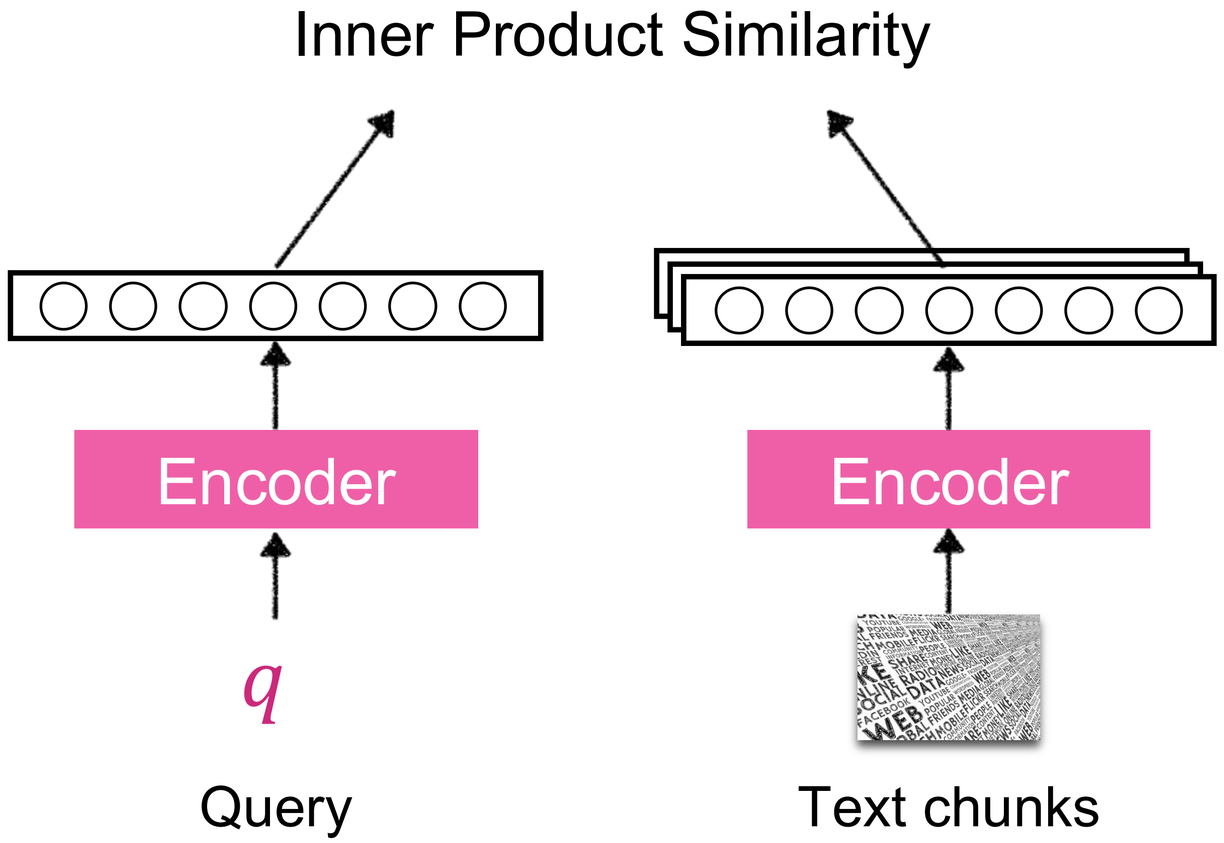


$$L(q, p^+, p_1^-, p_2^-, \ldots, p_n^-)$$

$$= -\log \frac{\exp(\text{sim}(q, p^+))}{\exp(\text{sim}(q, p^+)) + \sum_{j=1}^{n} \exp(\text{sim}(q, p_j^-)}$$

Karpukhin et al. Dense Passage Retrieval for Open-Domain Question Answering. EMNLP 2020.

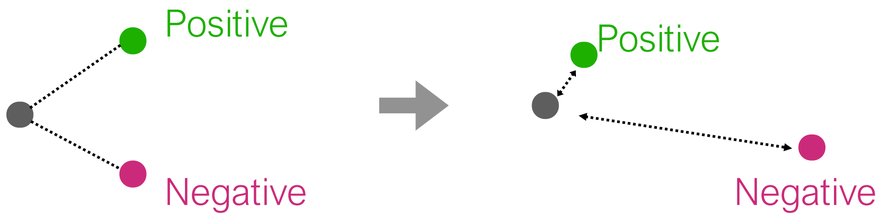
Training dense retrieval models: DPR



$$L(q, p^+, p_1^-, p_2^-, \ldots, p_n^-)$$

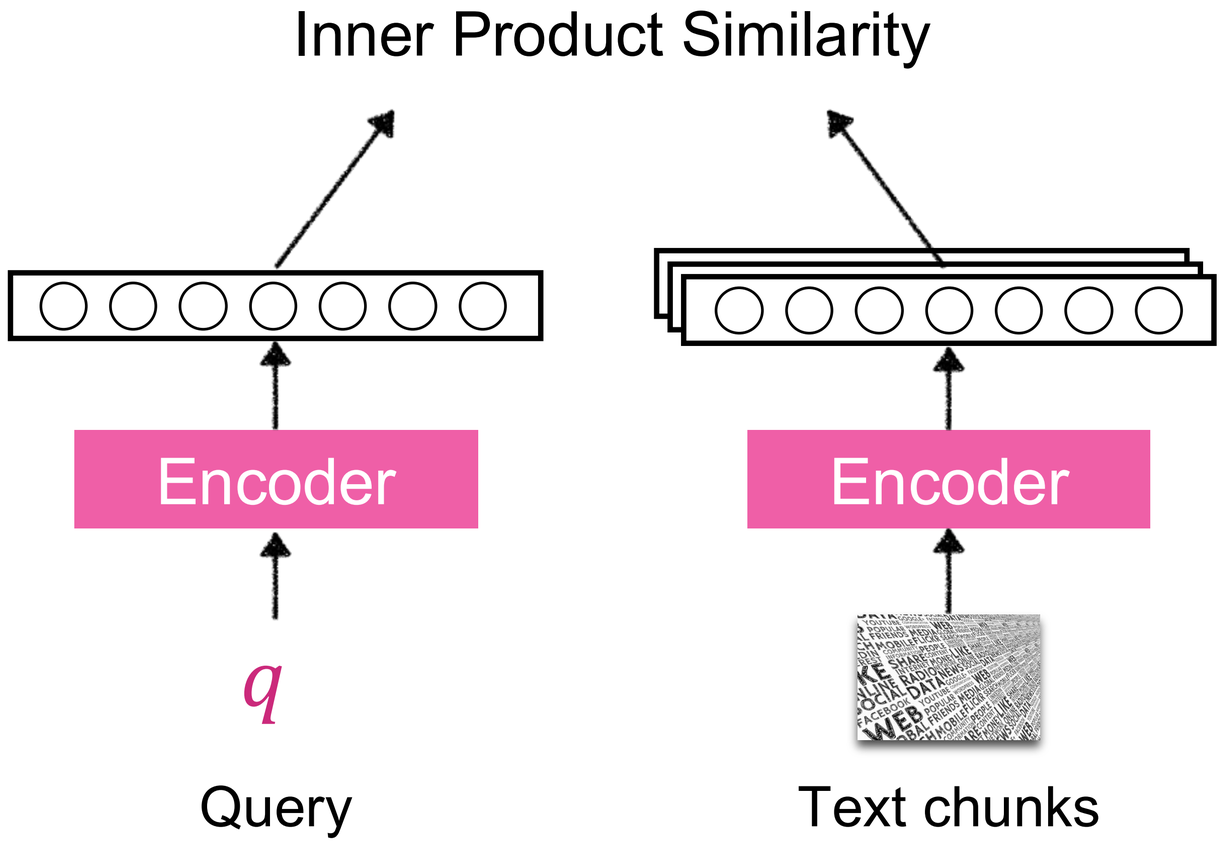
$$= -\log \frac{\exp(\text{sim}(q, p^+))}{\exp(\text{sim}(q, p^+)) + \sum_{j=1}^{n} \exp(\text{sim}(q, p_j^-))}$$

Contrastive learning



Karpukhin et al. Dense Passage Retrieval for Open-Domain Question Answering. EMNLP 2020.

Training dense retrieval models: DPR



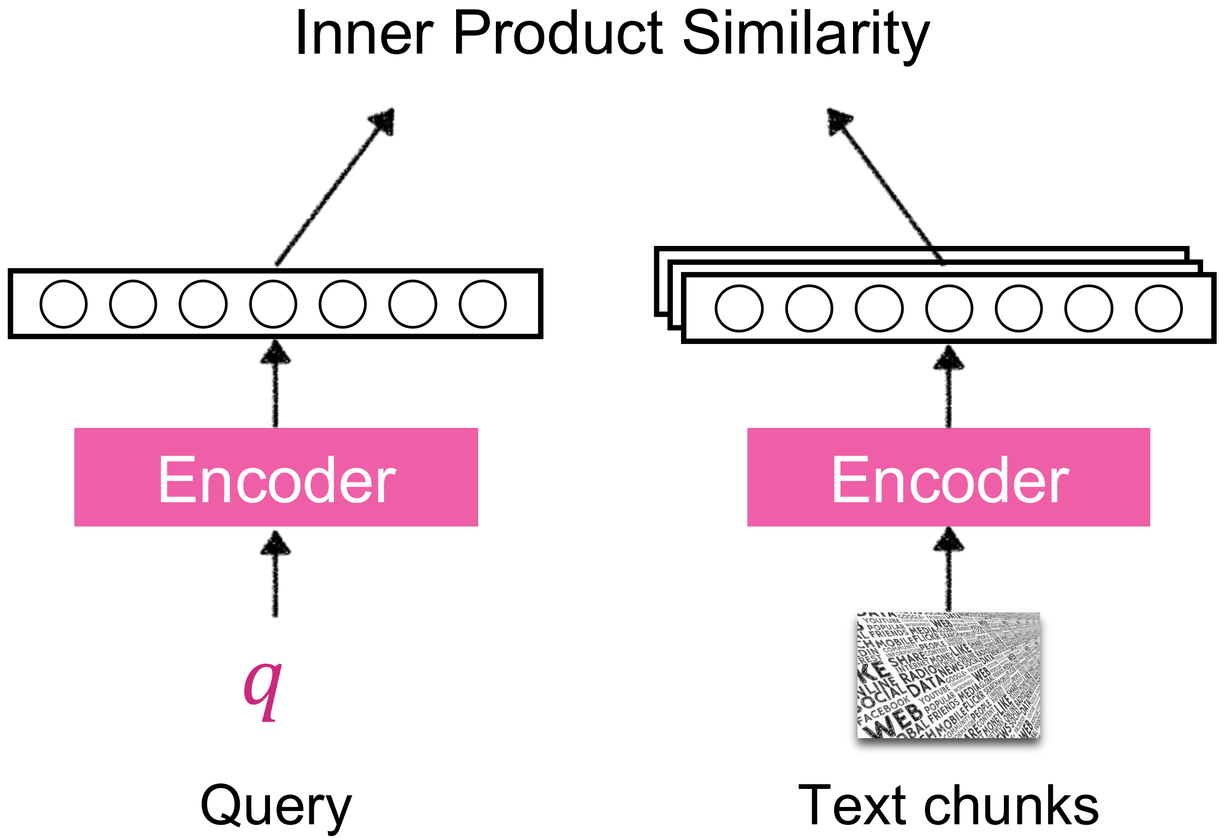
$$L(q, p^+, p_1^-, p_2^-, \ldots, p_n^-)$$

Positive passage

$$= -\log \frac{\exp(\text{sim}(q, p^+))}{\exp(\text{sim}(q, p^+)) + \sum_{j=1}^{n} \exp(\text{sim}(q, p_j^-)}$$

Karpukhin et al. Dense Passage Retrieval for Open-Domain Question Answering. EMNLP 2020.

Training dense retrieval models: DPR



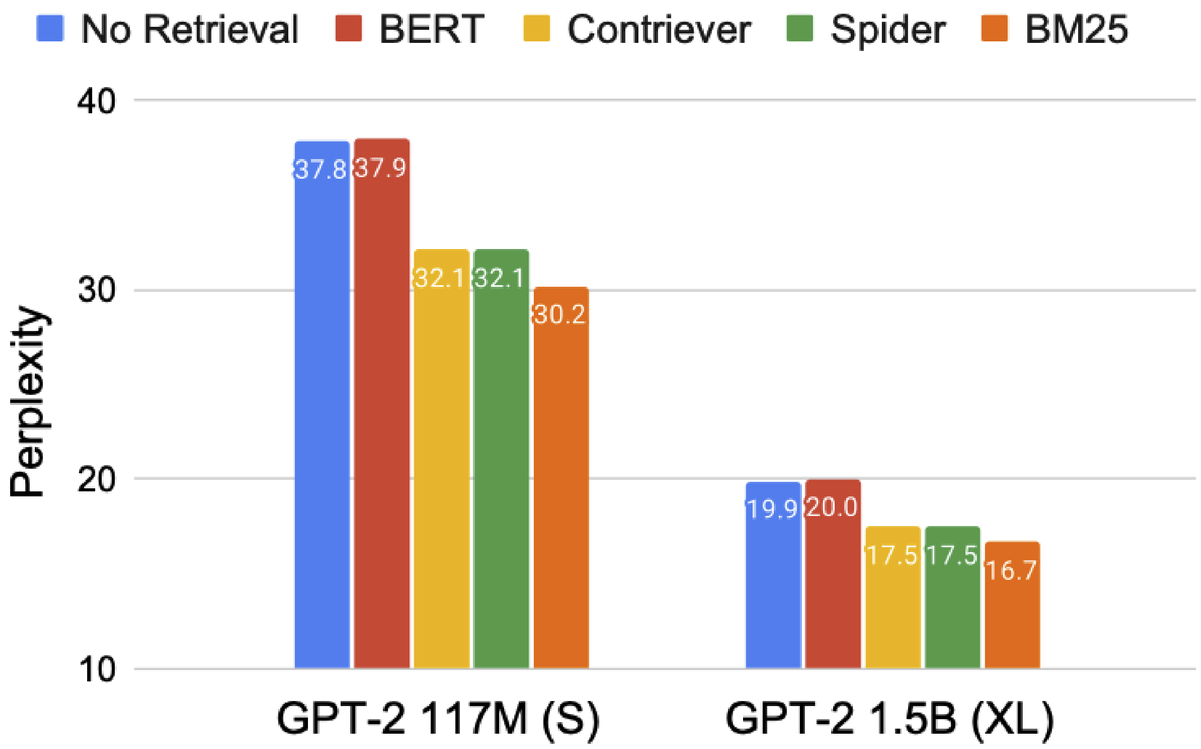
Negative passages
Too expensive to consider all negatives!

$$L(q, p^+, p_1^-, p_2^-, \ldots, p_n^-)$$

Positive passage

$$= -\log \frac{\exp(\mathrm{sim}(q, p^+))}{\exp(\mathrm{sim}(q, p^+)) + \sum_{j=1}^{n} \exp(\mathrm{sim}(q, p_j^-)}$$

RAG with LMs using different retrievers



Better retrieval model

Better base LMs

→ Better **retrieval-based LMs**

Each component can be improved separately

Ram et al. In-Context Retrieval-Augmented Language Models. TACL 2023.

Independent training

Work with off-the-shelf models (no extra training required)

Each part can be improved independently

Independent training

Work with off-the-shelf models (no extra training required)

Each part can be improved independently

LMs are not trained to leverage retrieval

Retrieval models are not optimized for LM tasks/domains

Training methods for retrieval-augmented LMs

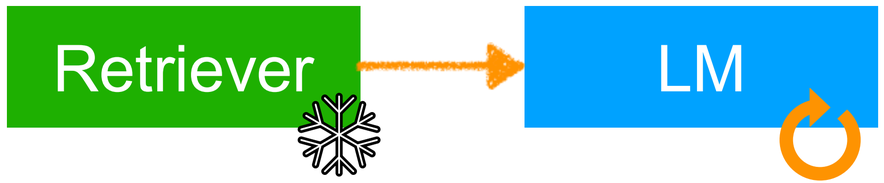
- Independent training
- **Sequential training**
- Joint training w/ asynchronous index update
- Joint training w/ in-batch approximation

Sequential training

- One component is first trained independently and then fixed
- The other component is trained with an objective that depends on the first one

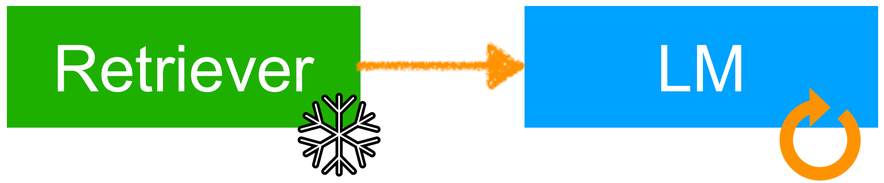
Sequential training

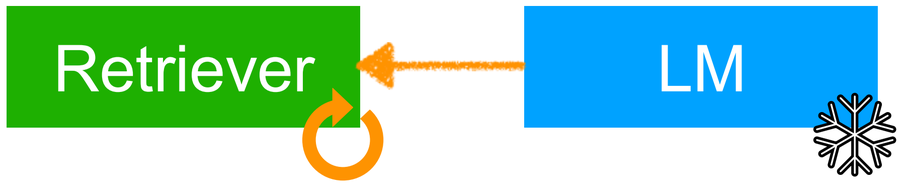
- One component is first trained independently and then fixed
- The other component is trained with an objective that depends on the first one



Sequential training

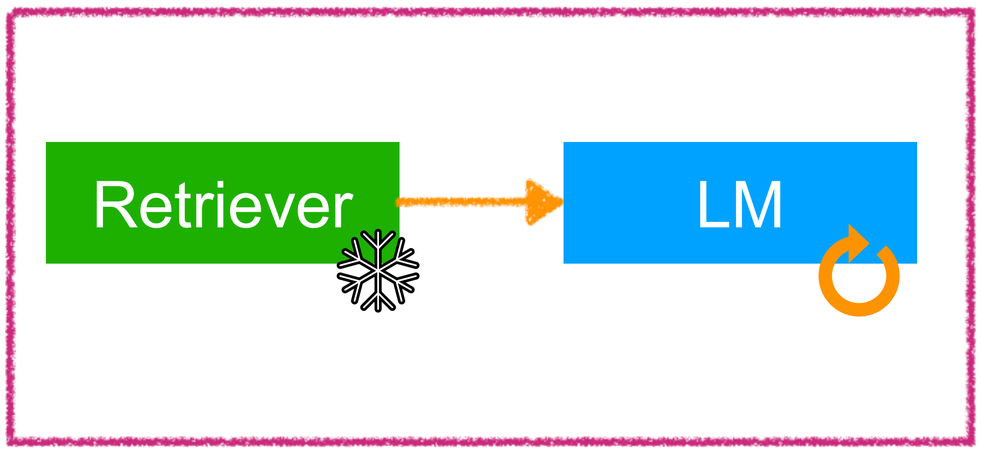
- One component is first trained independently and then fixed
- The other component is trained with an objective that depends on the first one



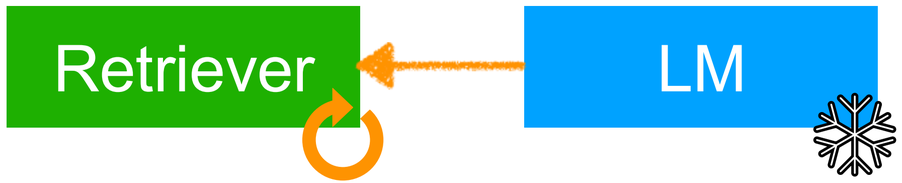


Sequential training

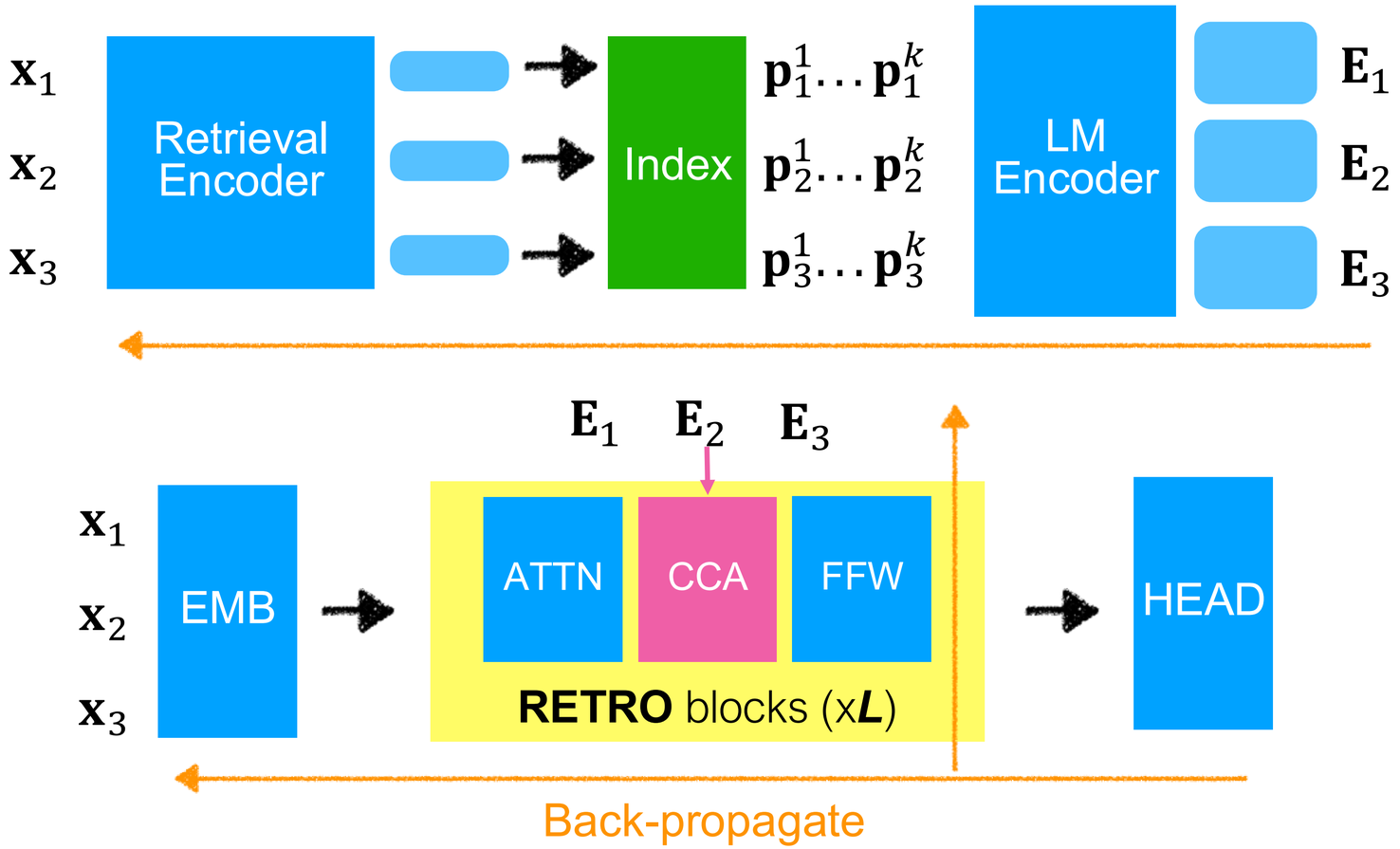
- One component is first trained independently and then fixed
- The other component is trained with an objective that depends on the first one



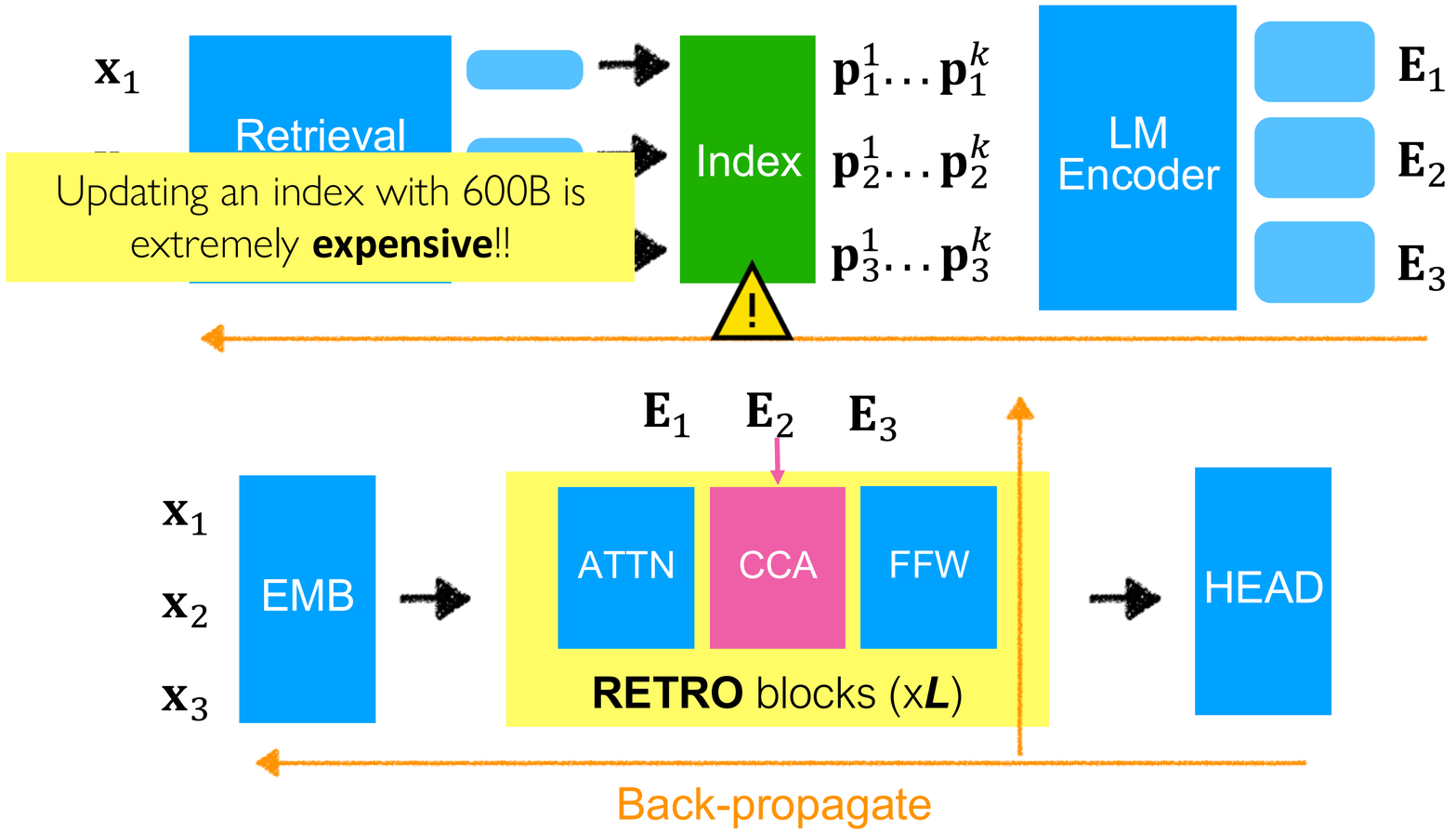
e.g., RETRO; WebGPT



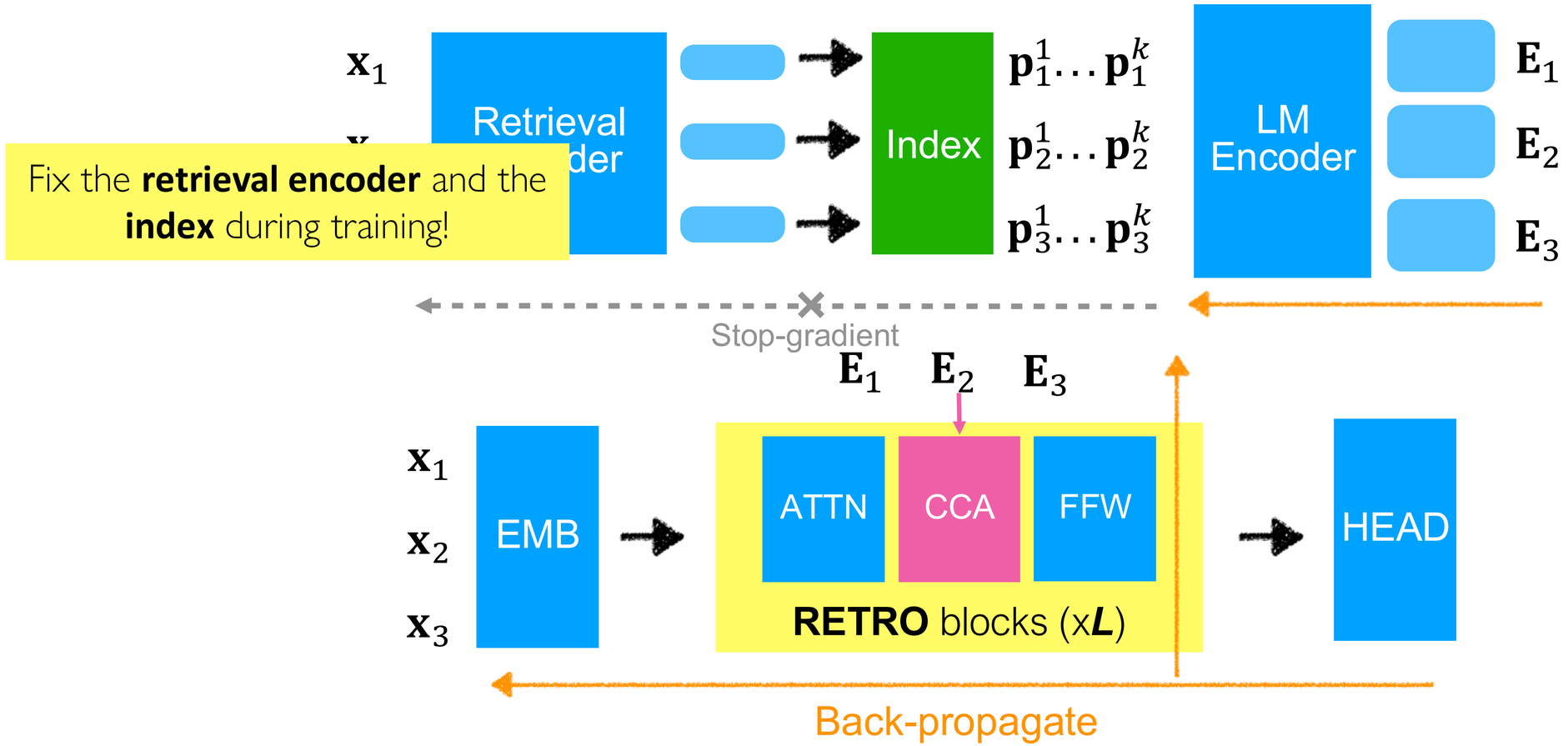
RETRO: Training



RETRO: Training



RETRO: Training



Sequential training


Work with off-the-shelf components (either a large index or a powerful LM)


LMs are trained to effectively leverage retrieval results


Retrievers are trained to provide text that helps LMs the most


One component is still fixed and not trained

Sequential training


Work with off-the-shelf components (either a large index or a powerful LM)


LMs are trained to effectively leverage retrieval results


Retrievers are trained to provide text that helps LMs the most


One component is still fixed and not trained

Let’s jointly train retrieval models and LMs!

Training methods for retrieval-augmented LMs

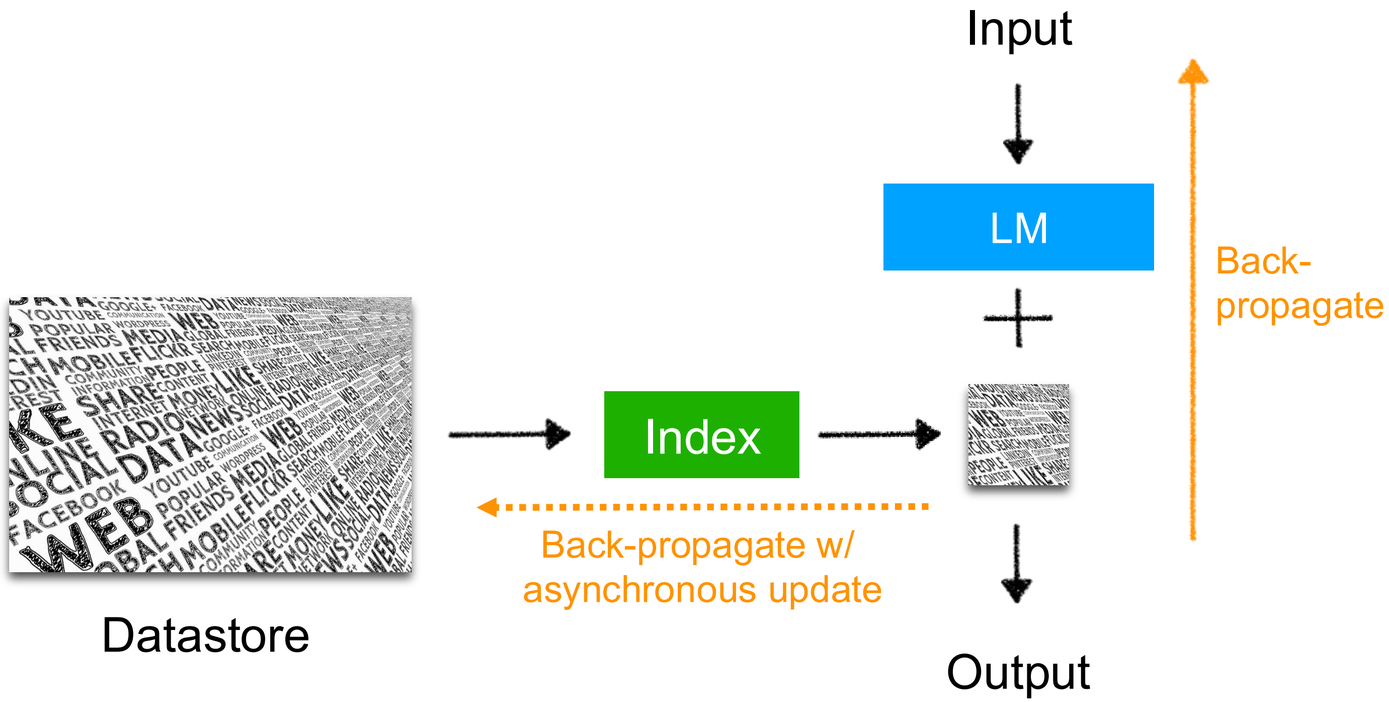
- Independent training
- Sequential training
- **Joint training w/ asynchronous index update**
- **Joint training w/ in-batch approximation**

Training methods for retrieval-augmented LMs

- Independent training
- Sequential training
- **Joint training w/ asynchronous index update**
- **Joint training w/ in-batch approximation**

Joint training w/ asynchronous index update

- Retrieval models and language models are trained jointly
- Allow the index to be “**stale**”; rebuild the retrieval index every T steps



Asynchronous index update

Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_3$

…

Encoder

Index

Asynchronous index update

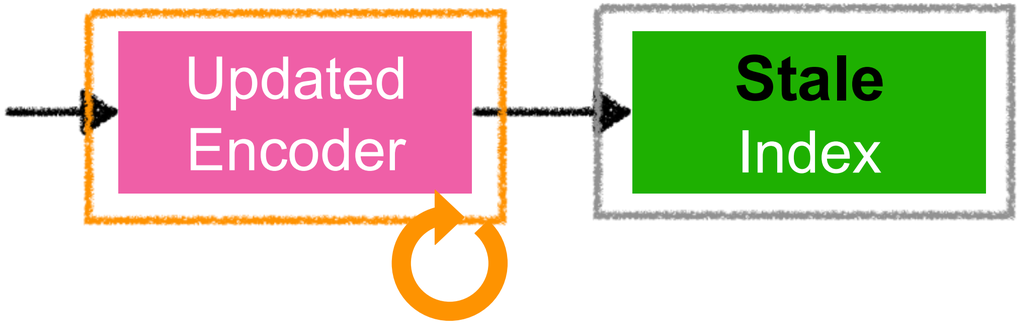
Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_3$

…



Asynchronous index update

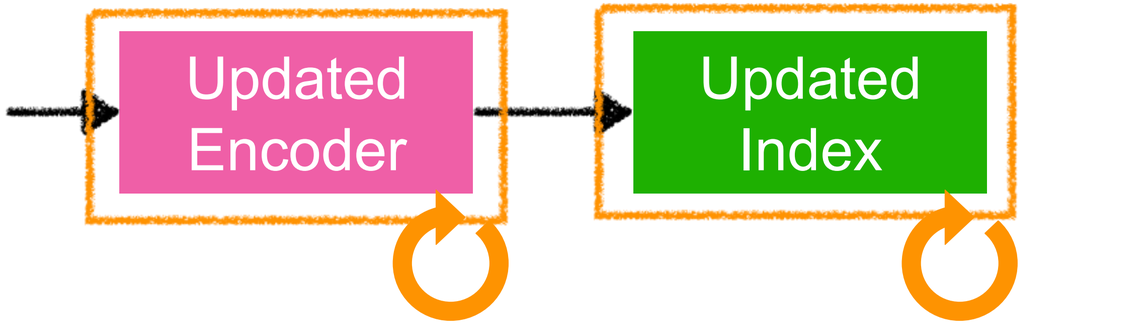
Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_3$

…

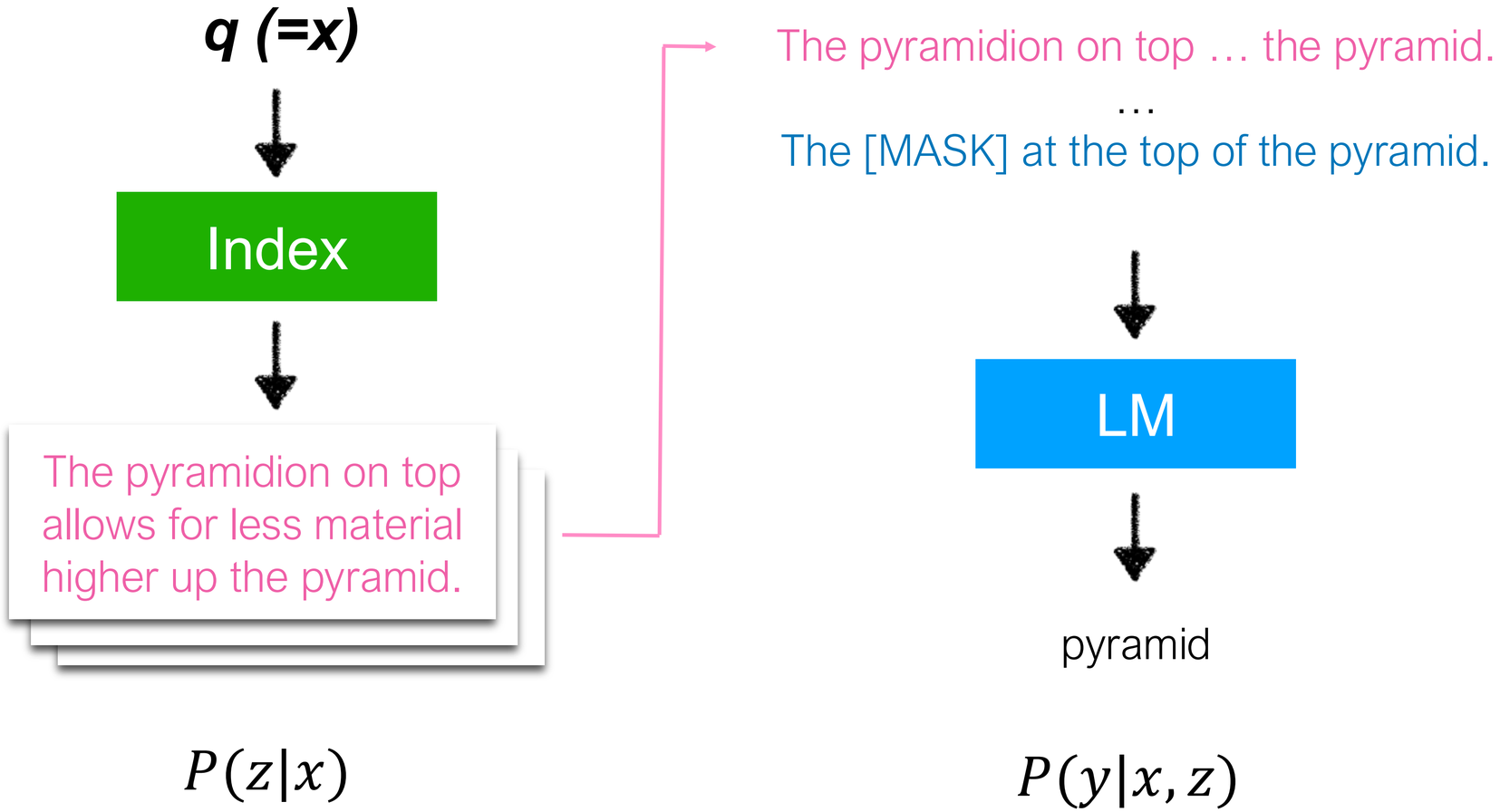


T updates

Refresh

REALM (Guu et al. 2020)

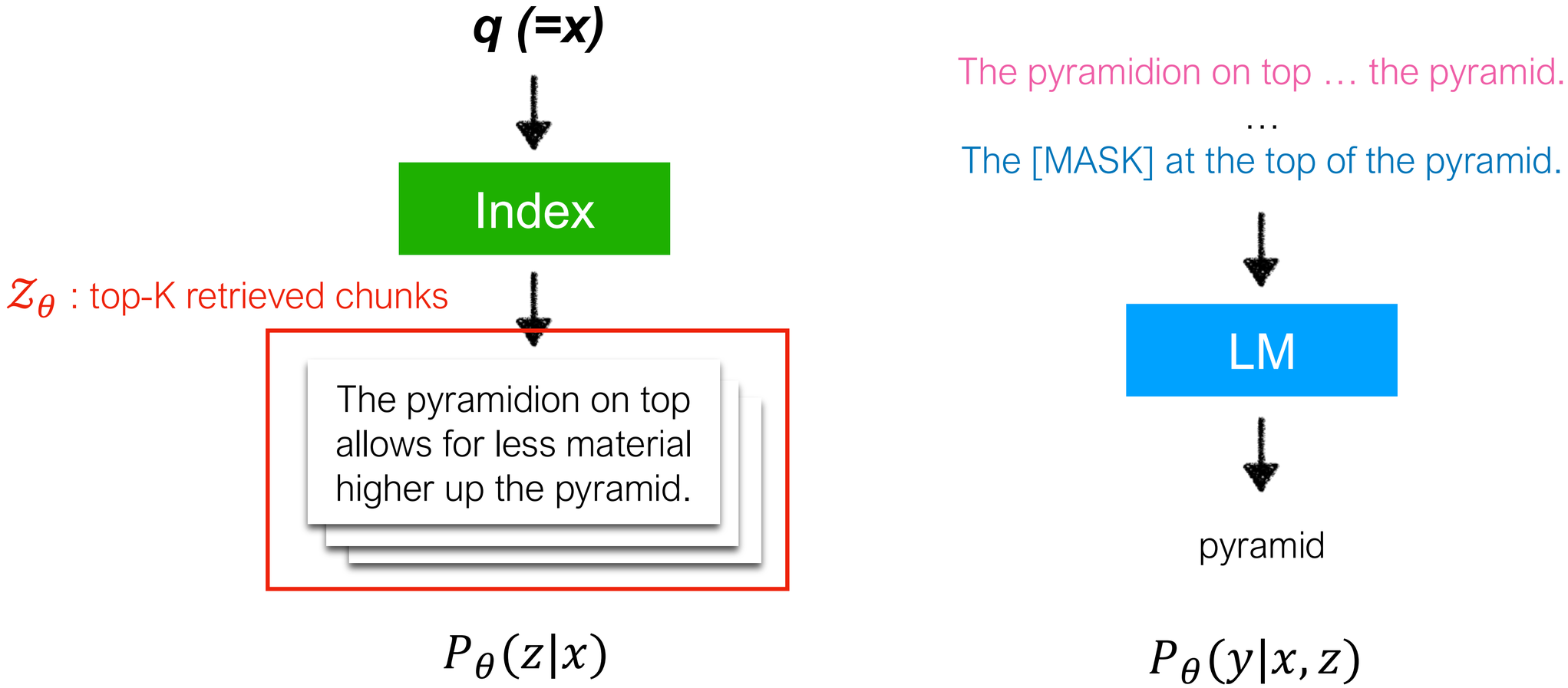
x = The [MASK] at the top of the pyramid.



Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

REALM: Training

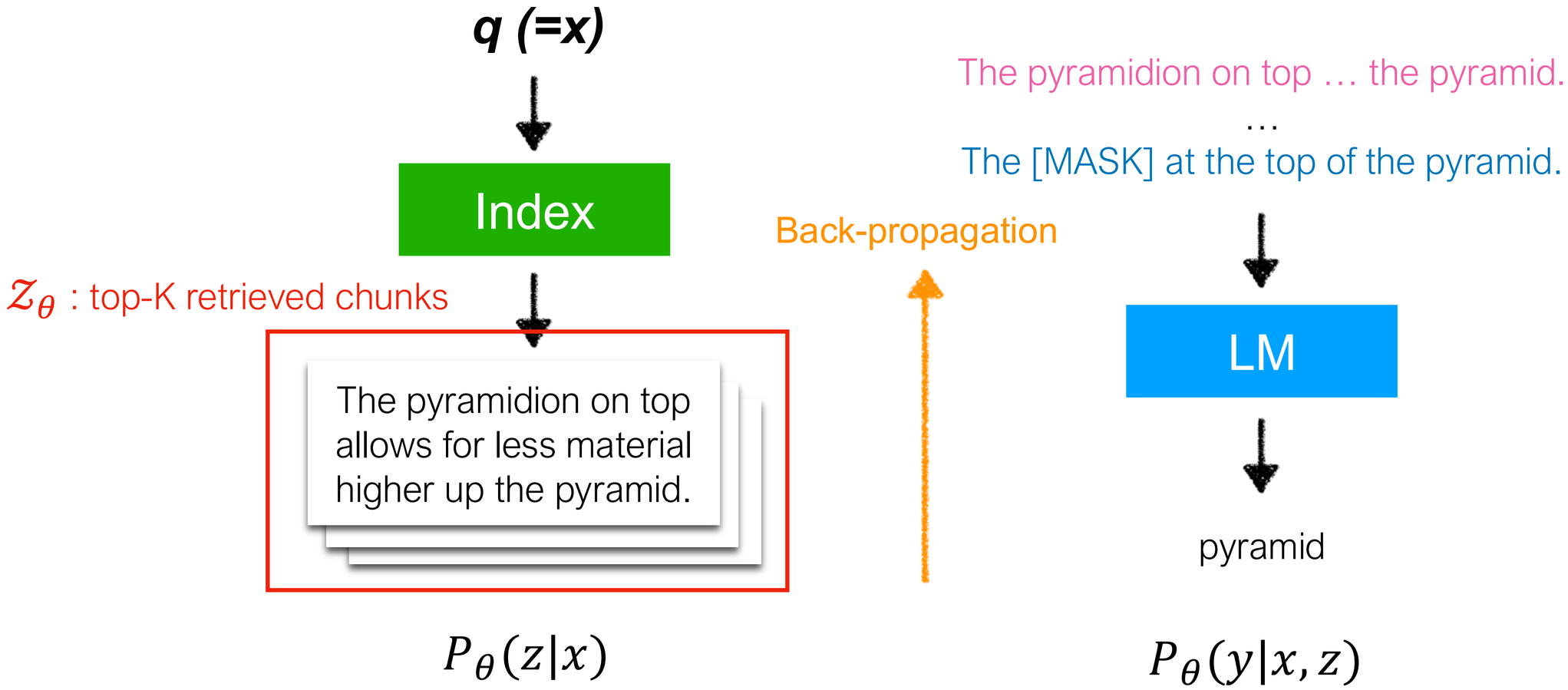
Objective: $\textbf{maximize} \sum_{z \in \mathcal{Z}_\theta} P_\theta(z|q) P_\theta(y|q,z)$



Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

REALM: Training

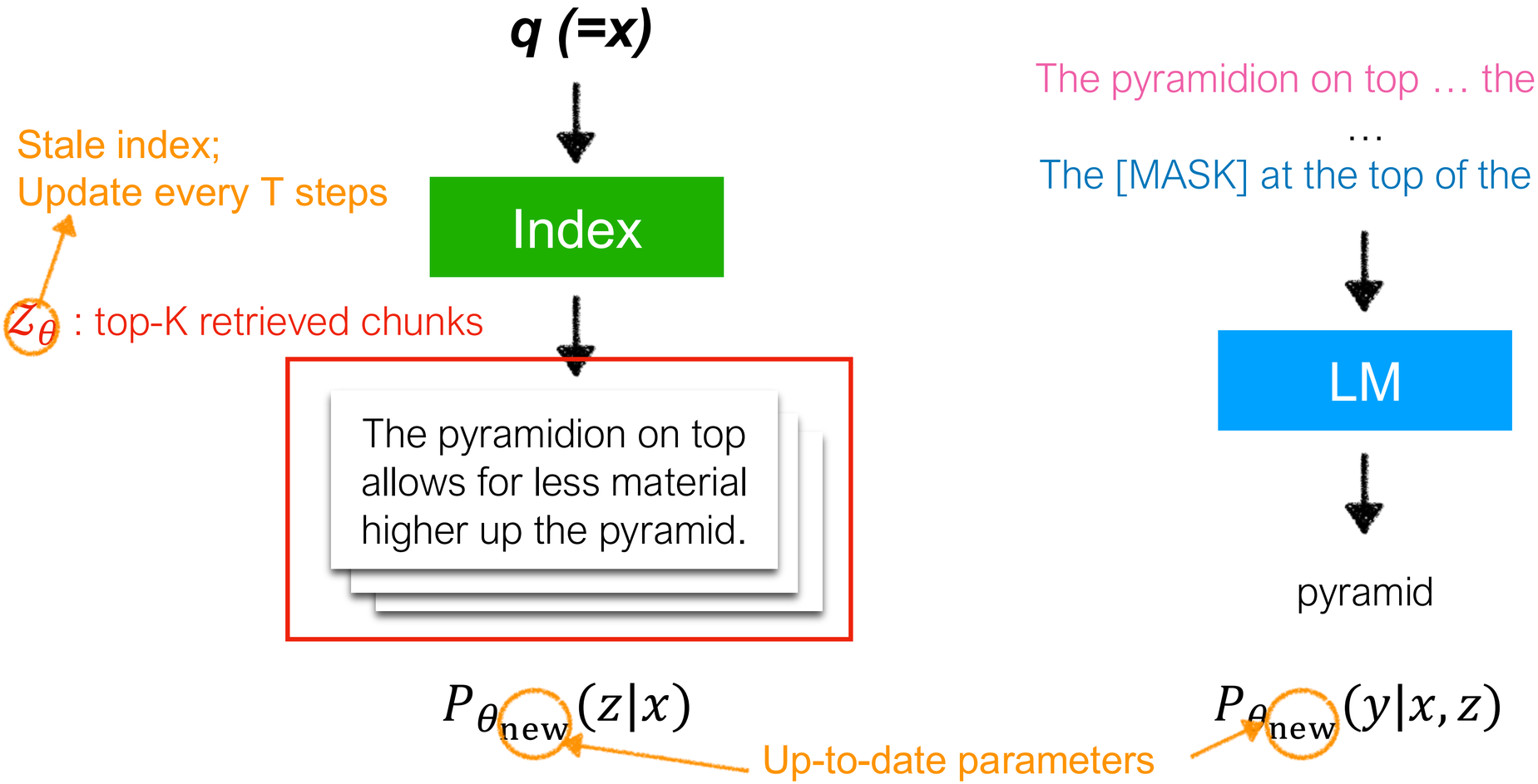
Objective: $\textbf{maximize} \sum_{z \in \mathcal{Z}_\theta} P_\theta(z|q) P_\theta(y|q,z)$



Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

REALM: Training

Objective: $\text{maximize} \sum_{z \in \mathcal{Z}_\theta} P_\theta(z|q) P_\theta(y|q,z)$

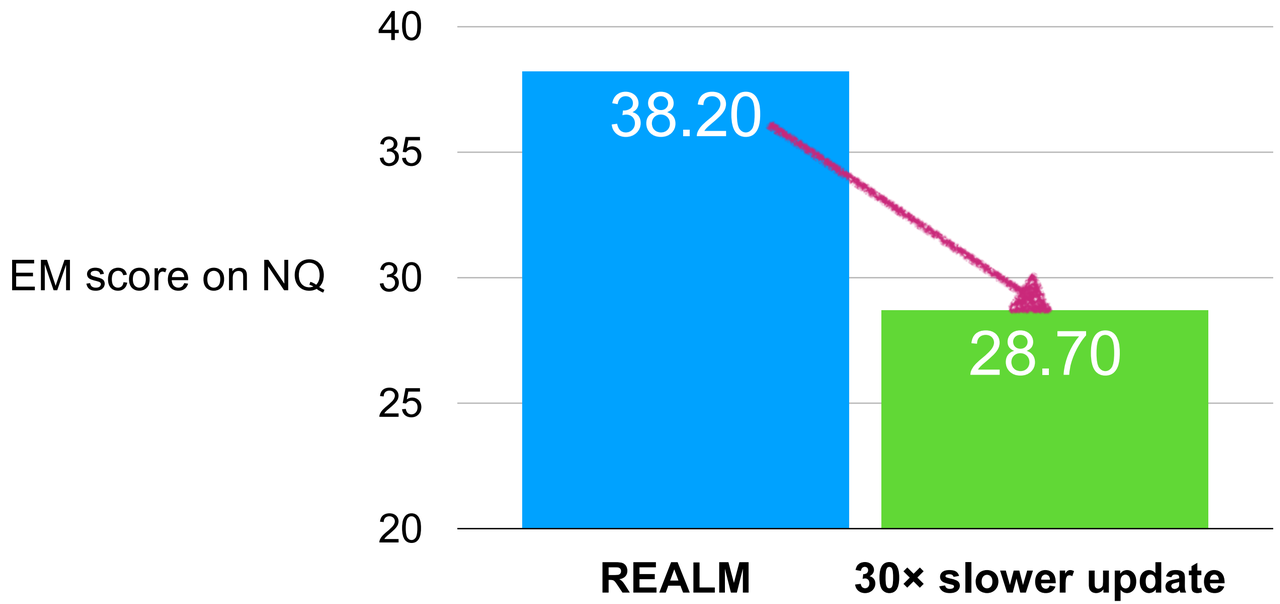


REALM: Index update rate

How often should we update the retrieval index?

- Frequency too high: expensive
- Frequency too slow: out-dated

REALM: updating the index every 500 training steps



Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

Joint training

End-to-end trained — each component is optimized

Good performance

Training is more complicated
(async update, overhead, data batching, etc)

Train-test discrepancy still remains

Today's outline

Why do we need retrieval-augmented LMs?

Architectures of retrieval-augmented LMs (Inference)

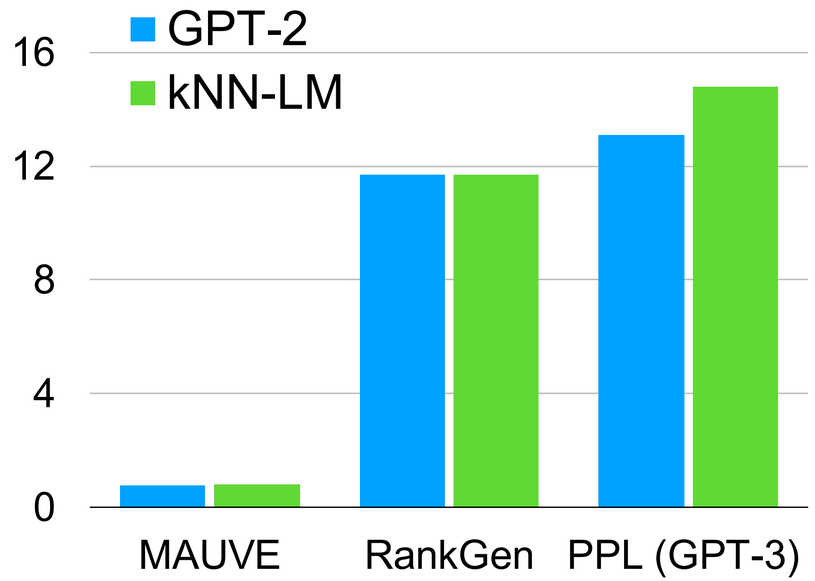
Training of retrieval-augmented LMs

Limitations and future directions

Challenge: retrieval-augmented LMs for applications

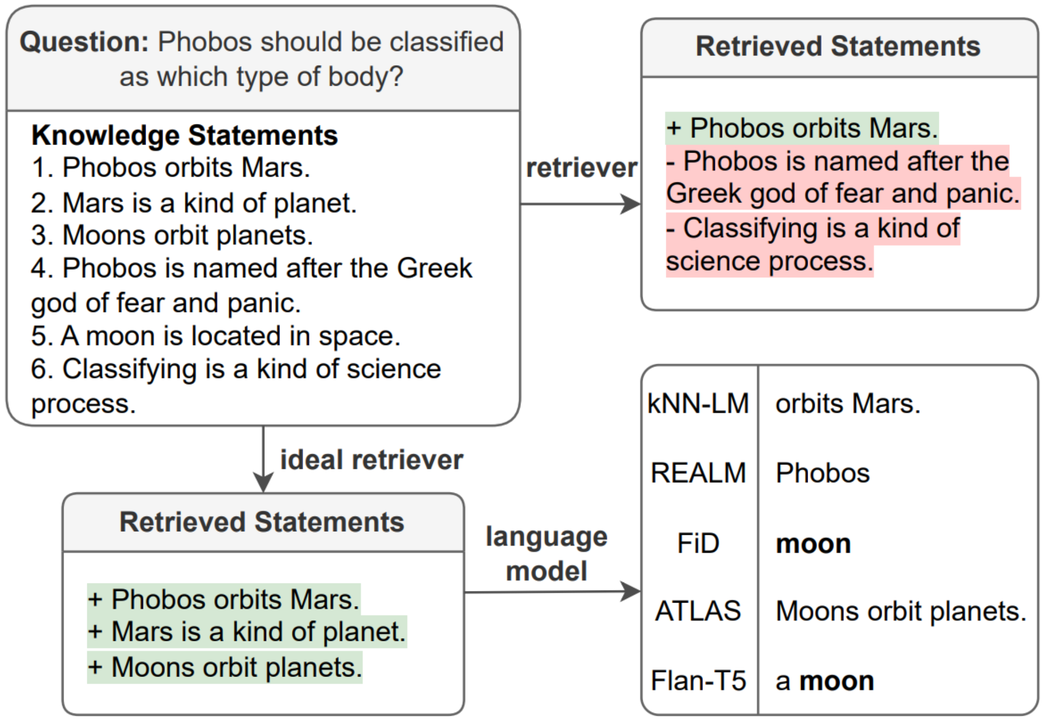
Open-ended text generation? Reasoning?

Doesn't improve open-ended generation



Wang et al. kNN-LM Does Not Improve Open-ended Text Generation. ACL 2023.

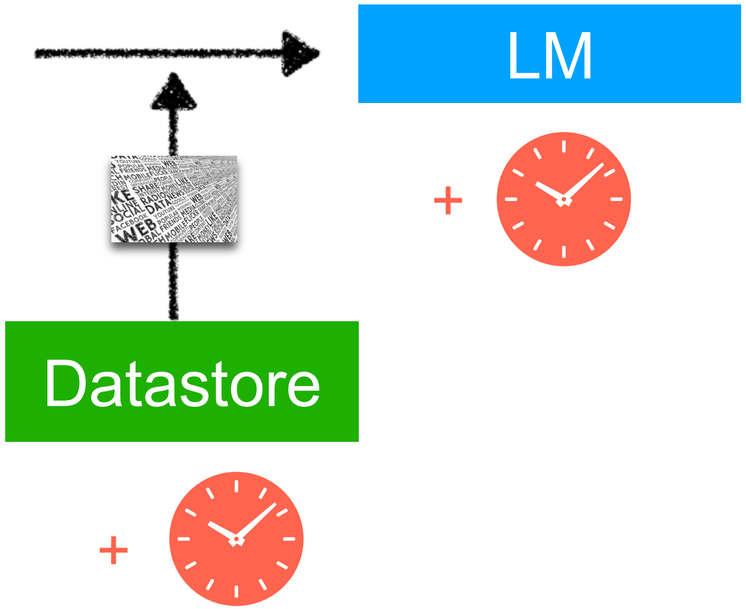
Failure of retrieval in reasoning task



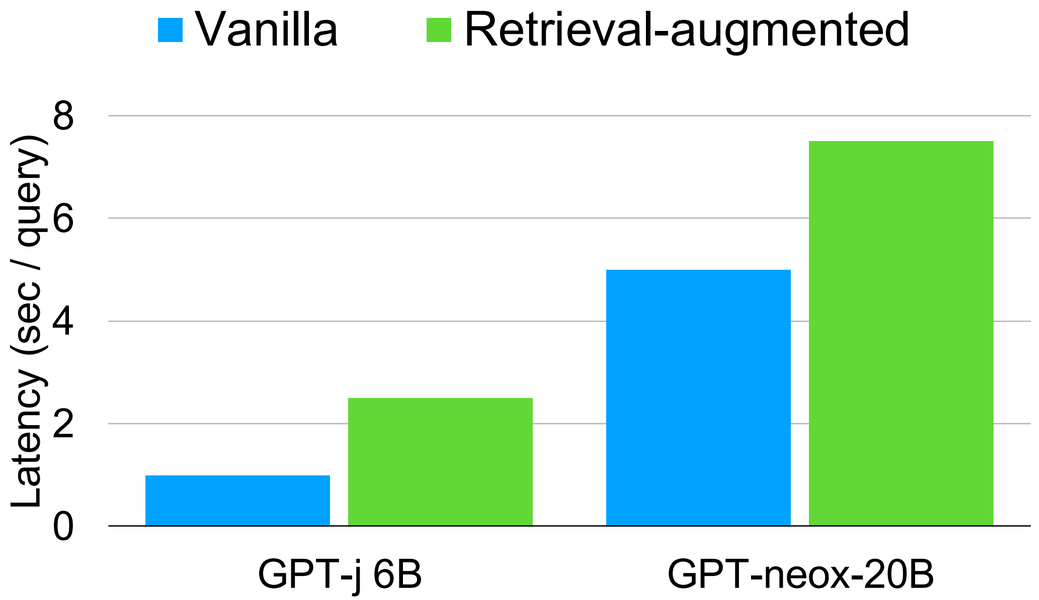
BehnamGhader et al. Can Retriever-Augmented Language Models Reason? The Blame Game Between the Retriever and the Language Model. EMNLP Findings 2023.

Challenge: efficiency retrieval-augmented LMs

Additional costs from retrieval augmentation



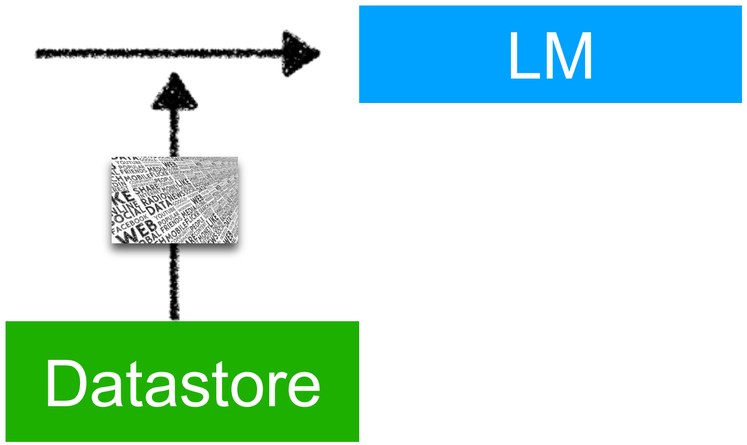
Retrieval-augmented LMs add inference costs



Mallen*, Asai* et al., When Not to Trust Language Models:
Investigating Effectiveness of Parametric and Non-Parametric Memories (Best Video; Oral) 2023.

Challenge: scaling retrieval-augmented LMs

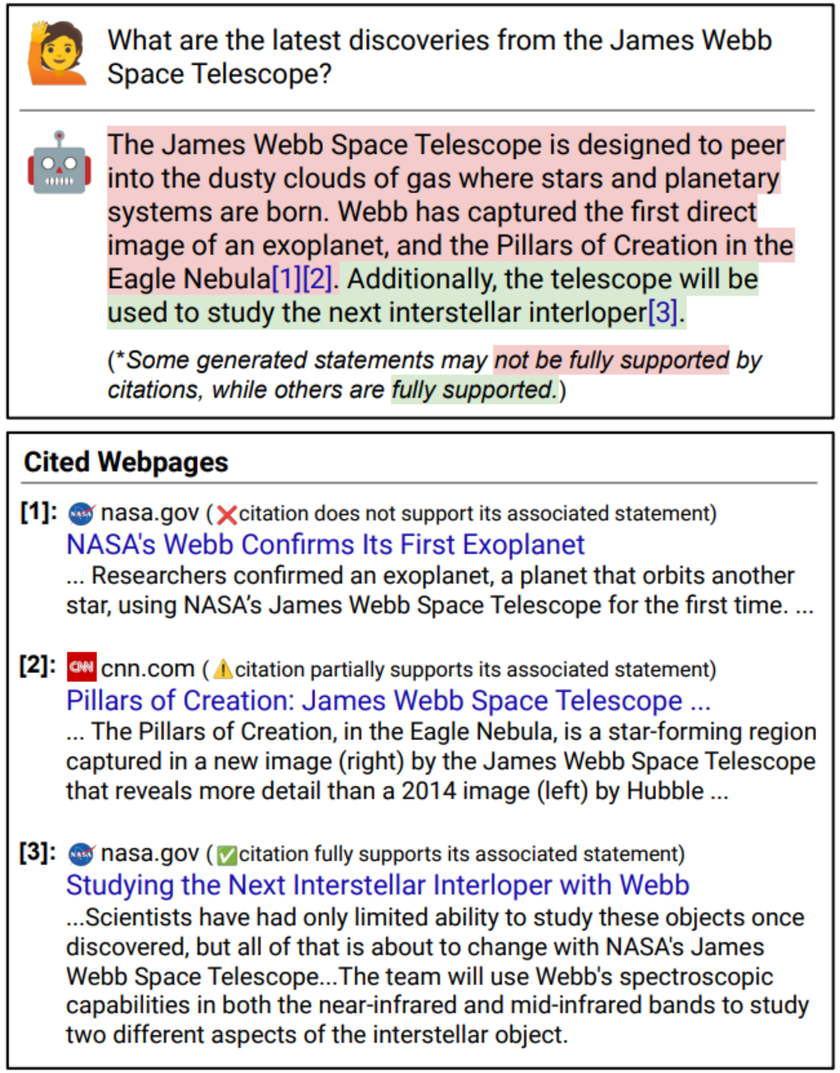
A small LM + a large datastore ≈ a large parametric LM?



vs.

	LM	Datastore
	# of parameters	# of tokens
kNN-LM (Khandelwal et al., 2020)	250M	≤ 3B
NPM (Min et al., 2023)	350M	1B
Atlas (Izacard et al., 2022)	11B	~30B
RETRO (Borgeaud et al., 2021)	7B	2T
REPLUG (Shi et al., 2023)	≤ 175B	~5B

Challenge: robustness and controllability



Retrieval-augmented LMs can still hallucinate

Liu et al. Evaluating Verifiability in Generative Search Engines. Findings of EMNLP 2023.

Roadmap to advance retrieval-augmented LMs

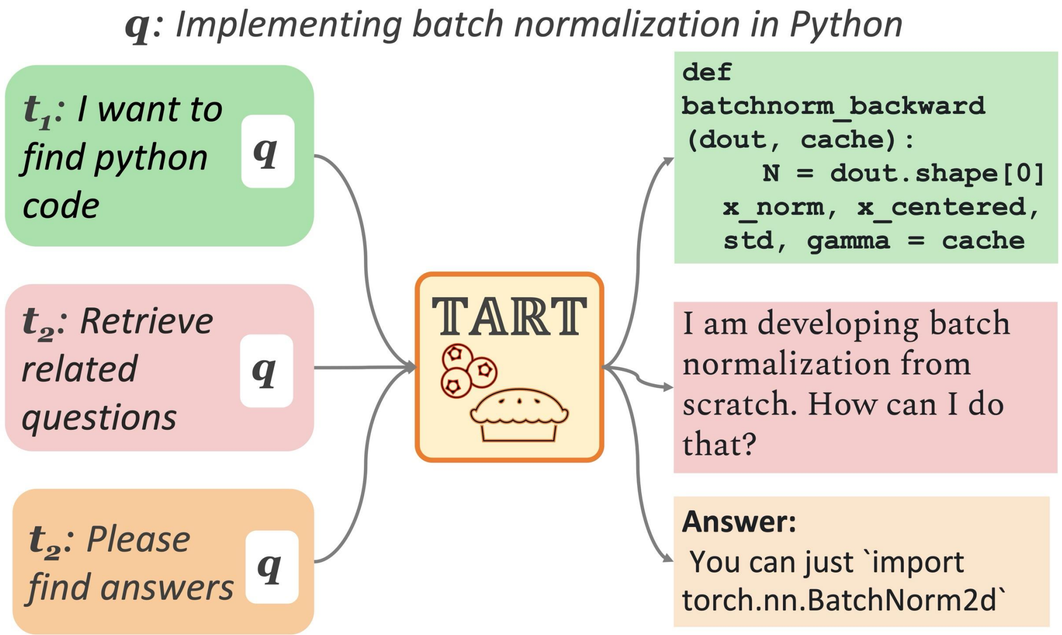
Rethink Retrieval and Datastore

Advance Architectures & Retrieval-aware Training

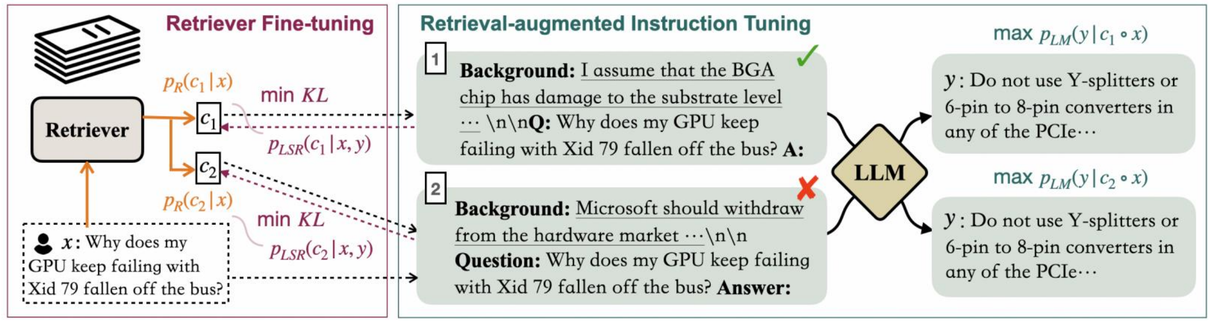
Investment Infrastructures for Training and Inference at Scale

Beyond semantic and lexical-similarity based search

Training retrievers to optimize end-to-end retrieval-augmented LM performance in diverse tasks tasks



Asai et al., Task-aware Retrieval with Instruction. Findings of ACL 2023.



0-shot	BoolQ	PIQA	SIQA	HellaSwag	WinoGrande
LLAMA 65B	85.3	82.8	52.3	84.2	77.0
RA-DIT 65B w/o retrieval	**86.7**	83.7	57.9	85.1	79.8
RA-DIT 65B	85.6	**84.4**	**58.4**	**85.4**	**80.0**

Lin et al., RA-DIT: Retrieval-Augmented Dual Instruction Tuning. ICLR 2024.

Roadmap to advance retrieval-augmented LMs

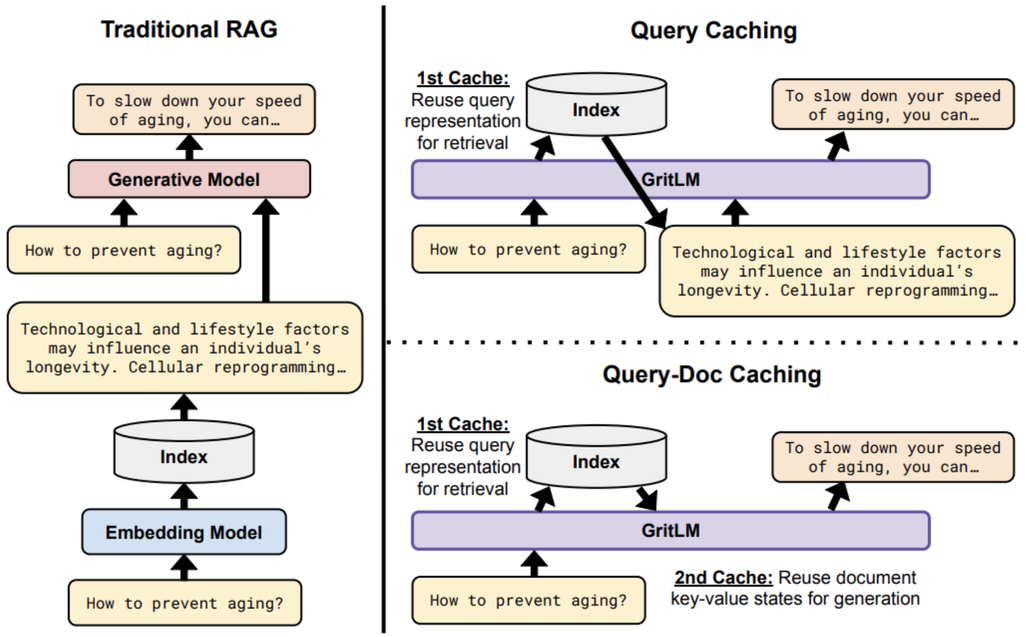
Rethink Retrieval and Datastore

Advance Architectures & Retrieval-aware Training

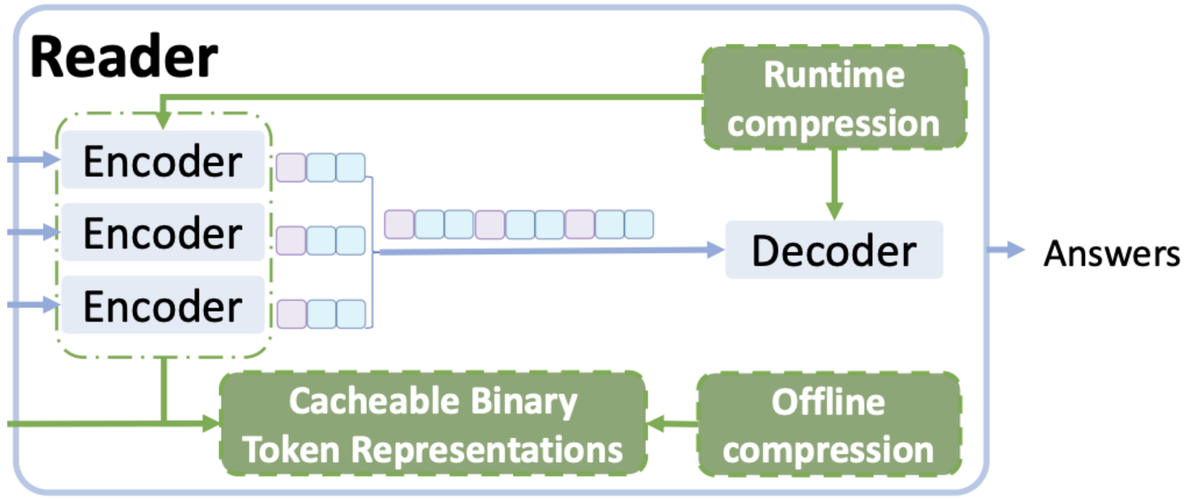
Investment Infrastructures for Training and Inference at Scale

New architectures for performance and efficiency

Further explorations of unified architectures & caching



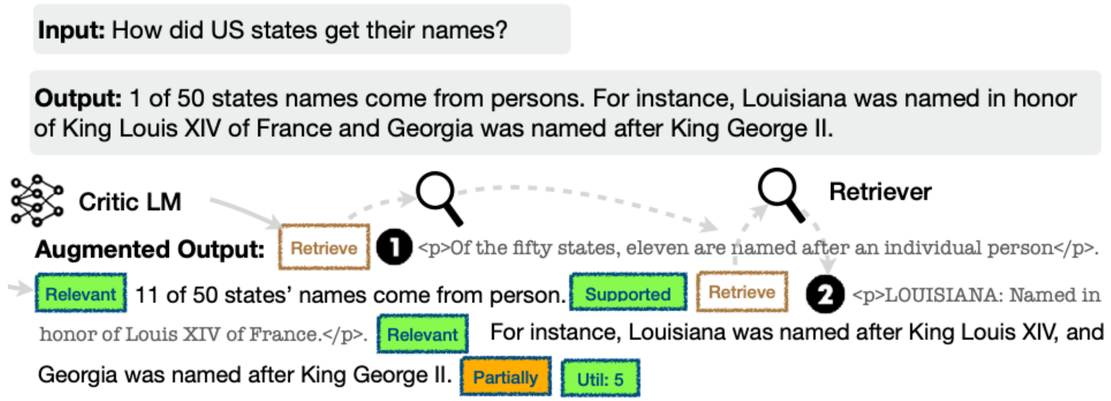
Muennighoff et al. Generative Representational Instruction Tuning. 2024.



Cao et al. BTR: Binary Token Representations for Efficient Retrieval Augmented Language Models. ICLR 2024.

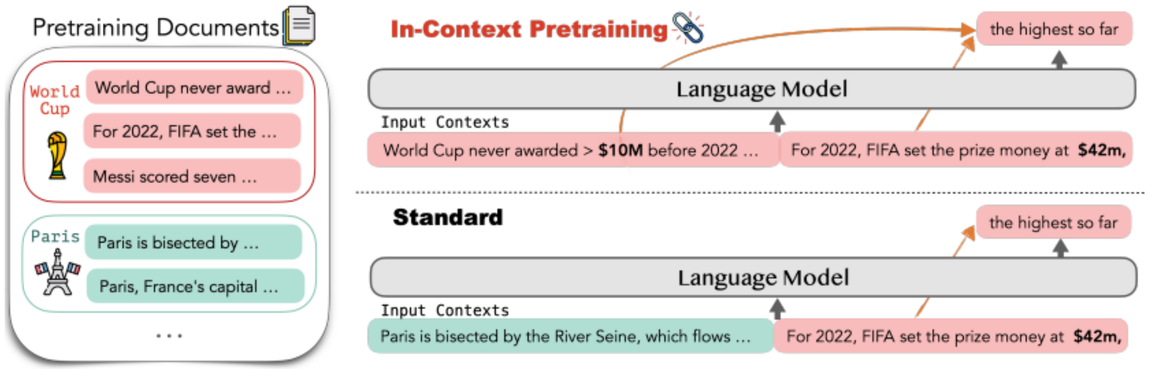
Training LMs with Retrieval

Training LMs to learn to use retrieval during pre-training or instruction-tuning



Instruction-tuning with retrieval

Asai et al. Self-RAG: Learning to Retrieve, Generate and Critique with Retrieval. ICLR 2024.



Retrieval-aware pre-training

Shi. et al. In-Context Pretraining: Language Modeling Beyond Document Boundaries. ICLR 2024.

Roadmap to advance retrieval-augmented LMs

Rethink Retrieval and Datastore

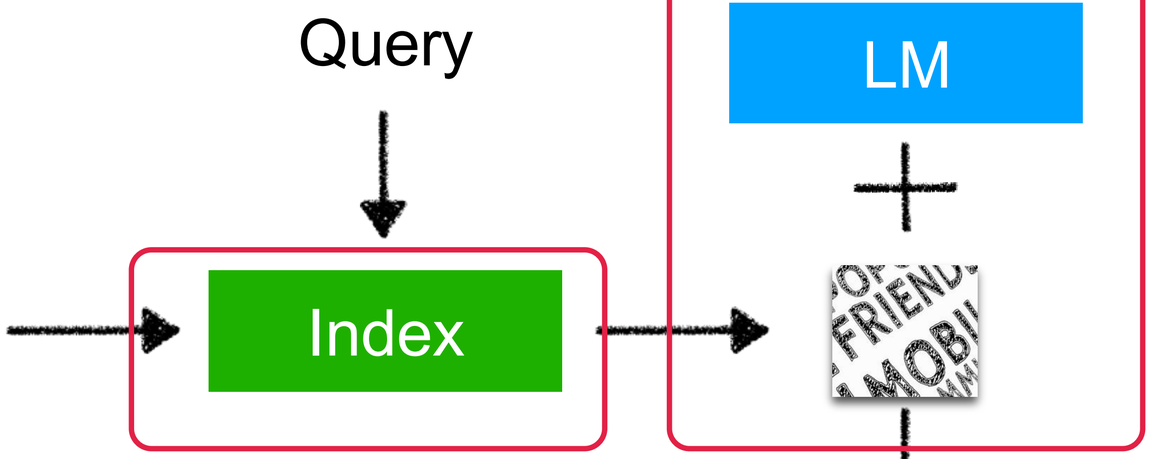
Advance Architectures &
Retrieval-aware Training

Investment Infrastructures for Training and Inference at Scale

Retrieval-augmented LMs can be really expensive!

Datastore

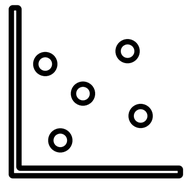
Scaling up DS to trillion tokens



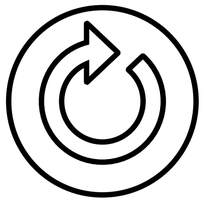
End-to-end training w/ Index updates

Inference with many documents

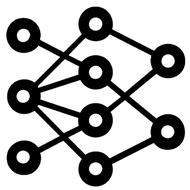
More open-sourced and collaborative opportunities



System / Algorithmic improvements for massive Datastore



Standardized implementations for efficient training



Fast inference algorithms for retrieval-augmented LMs

Summary & QA

Retrieval-augmented LMs can solve many issues e.g., hallucinations

Various architectures (not just RAG) exist with different pros&cons

Jointly training retrieval-augmented LMs is important but hard

ACL 2023 tutorial: https://acl2023-retrieval-lm.github.io/

Position paper: https://akariasai.github.io/assets/pdf/ralm_position.pdf

Contact: akari@cs.washington.edu

Website: https://akariasai.github.io/

Twitter: @AkariAsai

References (1)

Akari Asai, Sewon Min, Zexuan Zhong, Danqi Chen. Retrieval-based Language Models and Applications. ACL Tutorial 2023.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, Matt Gardner. Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus. EMNLP 2021.

Alex Mallen*, Akari Asai*, Victor Zhong, Rajarshi Das, Daniel Khashabi, Hannaneh Hajishirzi. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories. ACL 2023.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, Colin Raffel. Large Language Models Struggle to Learn Long-Tail Knowledge. ICML 2023.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, Kentaro Inui. RealTime QA: What's the Answer Right Now?. NeurIPS (Dataset & Benchmark) 2023.

References (2)

Sewon Min, Suchin Gururangan, Eric Wallace, Hannaneh Hajishirzi, Noah A. Smith, Luke Zettlemoyer. SILO Language Models: Isolating Legal Risk In a Nonparametric Datastore. ICLR 2024.

Juan Ramos. Using TF-IDF to Determine Word Relevance in Document Queries. The first instructional conference on machine learning 2003.

Robertson and Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond. Foundations and Trends® in Information Retrieval 2009.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, Wen-tau Yih. Dense Passage Retrieval for Open-Domain Question Answering. EMNLP 2020.

Akari Asai, Zexuan Zhong, Danqi Chen, Pang Wei Koh, Luke Zettlemoyer , Hannaneh Hajishirzi, Wen-tau Yih. Reliable, Adaptable, and Attributable Language Models with Retrieval. Arxiv 2024.

References (3)

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, Ming-Wei Chang. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, Yoav Shoham. In-Context Retrieval-Augmented Language Models. arXiv 2023.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, Wen-tau Yih. REPLUG: Retrieval-Augmented Black-Box Language Models. arXiv 2023.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George 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 W. Rae, Erich Elsen, Laurent Sifre. Improving language models by retrieving from trillions of tokens. arXiv 2021.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, Mike Lewis. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

References (4)

Shufan Wang, Yixiao Song, Andrew Drozdov, Aparna Garimella, Varun Manjunatha, Mohit Iyyer. KNN-LM Does Not Improve Open-ended Text Generation. ACL 2023.

Parishad BehnamGhader, Santiago Miret, Siva Reddy. Can Retriever-Augmented Language Models Reason? The Blame Game Between the Retriever and the Language Model. Findings of EMNLP 2023.

Nelson F. Liu, Tianyi Zhang, Percy Liang. Evaluating Verifiability in Generative Search Engines. Findings of EMNLP 2023.

Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, Wen-tau Yih. Task-aware Retrieval with Instructions. Findings of ACL 2023.

Xi Victoria Lin, Xilun Chen, Mingda Chen, Weijia Shi, Maria Lomeli, Rich James, Pedro Rodriguez, Jacob Kahn, Gergely Szilvasy, Mike Lewis, Luke Zettlemoyer, Scott Yih. RA-DIT: Retrieval-Augmented Dual Instruction Tuning. ICLR 2024.

References (5)

Niklas Muennighoff, Hongjin Su, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, Douwe Kiela. Generative Representational Instruction Tuning. Arxiv 2024.

Qingqing Cao, Sewon Min, Yizhong Wang, Hannaneh Hajishirzi. BTR: Binary Token Representations for Efficient Retrieval Augmented Language Models. ICLR 2024.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, Hannaneh Hajishirzi. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. ICLR 2024.

Weijia Shi, Sewon Min, Maria Lomeli, Chunting Zhou, Margaret Li, Rich James, Xi Victoria Lin, Noah A. Smith, Luke Zettlemoyer, Scott Yih, Mike Lewis. In-Context Pretraining: Language Modeling Beyond Document Boundaries. ICLR 2024.