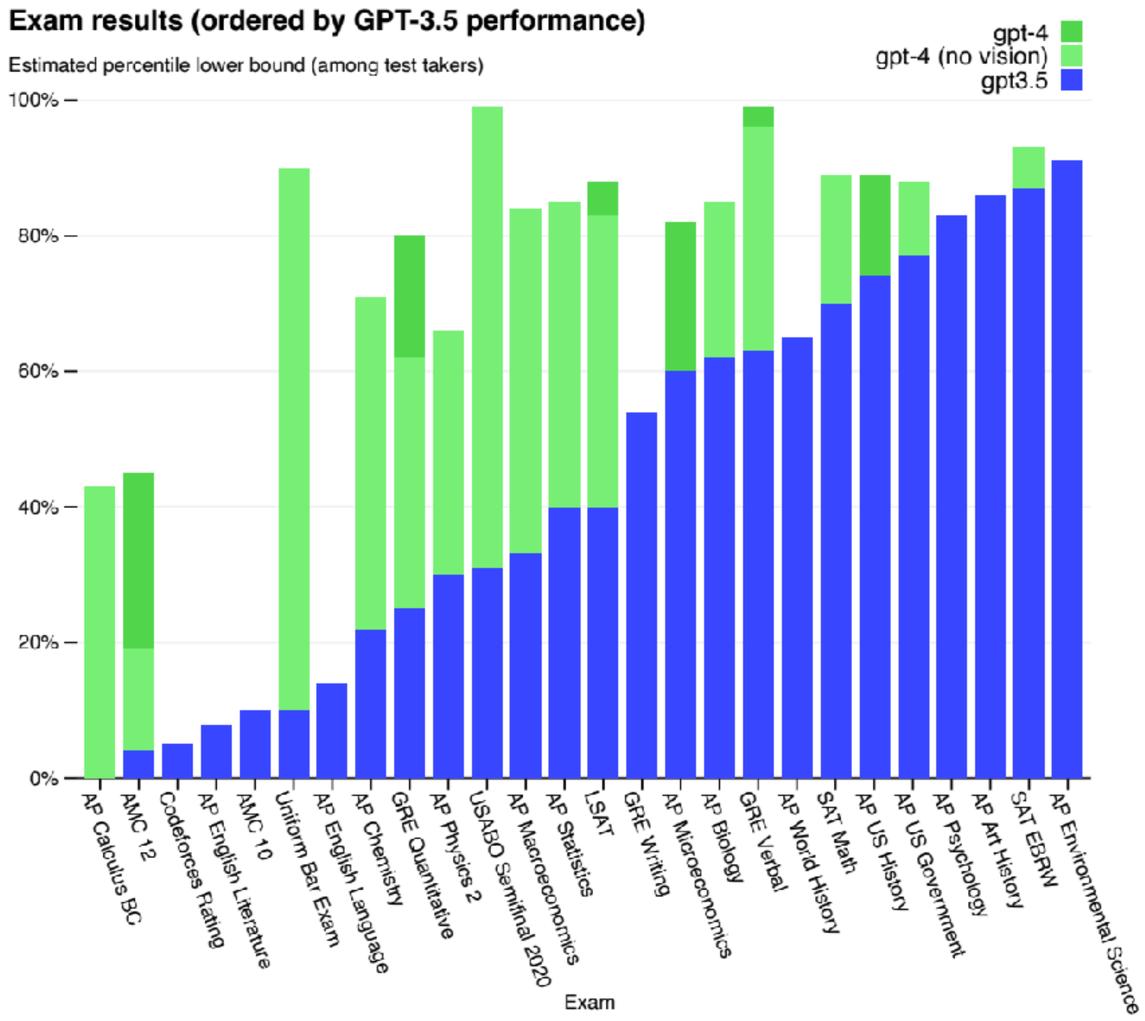
Retrieval-Augmented Language Models

CS6120: Natural Language Processing
Northeastern University

David Smith
with slides from Graham Neubig and Akari Asai

Knowledgable and versatile Language Models (LMs)



Open AI. 2023. GPT4 Technical Report.

Knowledgable and versatile Language Models (LMs)

AK **You**

Write a poem about large language models, including the following three words: coffee, croissants and banana (*February 18, 2024)

Knowledgable and versatile Language Models (LMs)

AK **You**

Write a poem about large language models, including the following three words: coffee, croissants and banana (*February 18, 2024)

ChatGPT

In the realm where thoughts converge and flow,
Resides a wonder, a grand tableau,
Where words dance in intricate array,
In the domain of the model's sway.

From the depths of bytes, they come alive,
In the language vast, where dreams contrive,
Coffee-fueled, the model's might,
Awakens realms in the dead of night.

Croissants of syntax, buttered with care,
Layered meanings, rich and rare,
Each line a pastry, crisp and sweet,
In the banquet of words, where senses meet.

Banana-shaped metaphors, ripe and bold,
In the orchard of thoughts, they unfold,
Symbolic fruits of the mind's terrain,
Nourishing narratives, ripe with gain.

Knowledgable and versatile Language Models (LMs)

AK **You**

Write a poem about large language models, including the following three words: coffee, croissants and banana (*February 18, 2024)

ChatGPT

In the realm where thoughts converge and flow,
Resides a wonder, a grand tableau,
Where words dance in intricate array,
In the domain of the model's sway.

From the depths of bytes, they come alive,
In the language vast, where dreams contrive,
Coffee-fueled, the model's might,
Awakens realms in the dead of night.

Croissants of syntax, buttered with care,
Layered meanings, rich and rare,
Each line a pastry, crisp and sweet,
In the banquet of words, where senses meet.

Banana-shaped metaphors, ripe and bold,
In the orchard of thoughts, they unfold,
Symbolic fruits of the mind's terrain,
Nourishing narratives, ripe with gain.

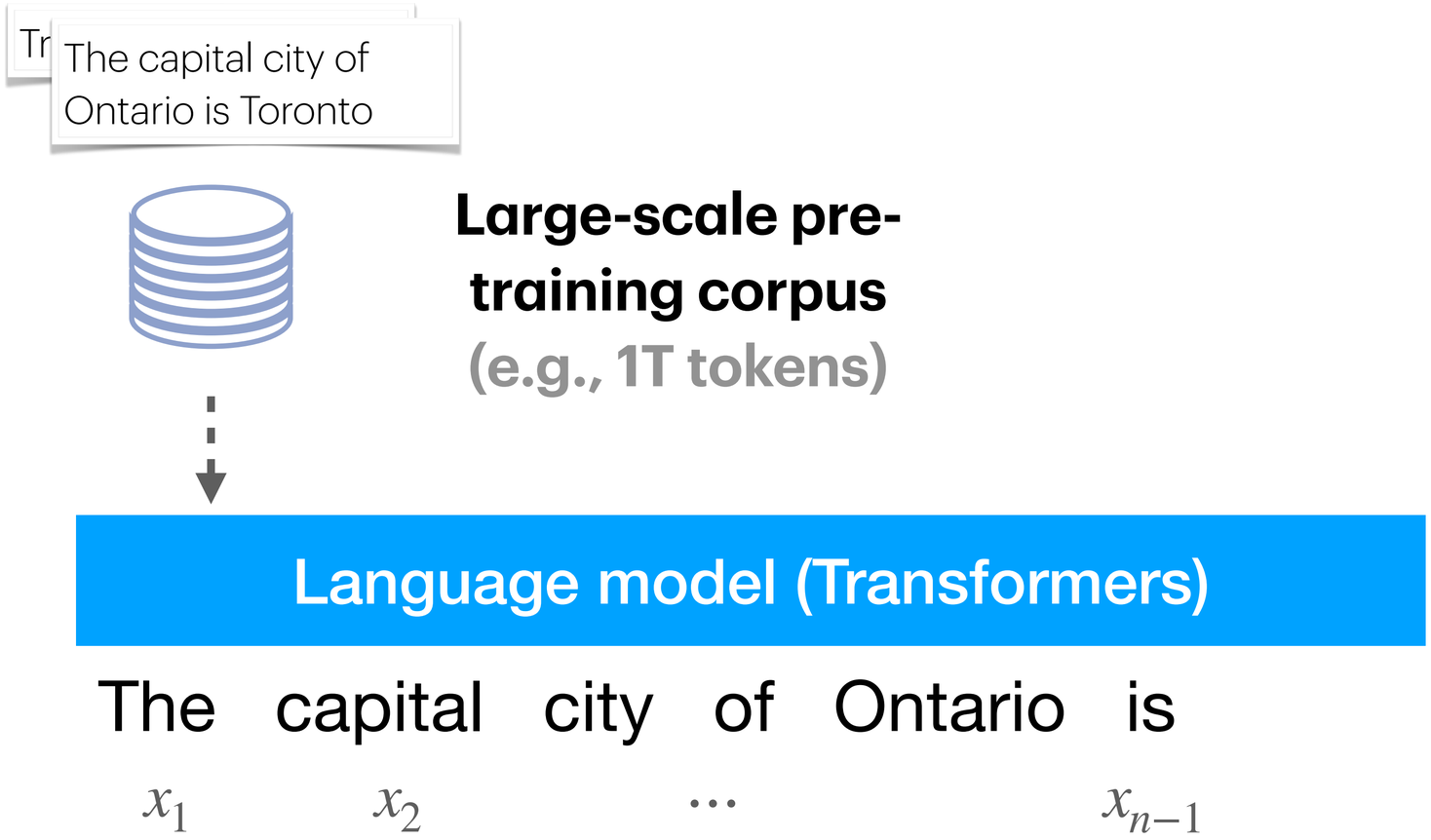
How do such parametric LMs work?

Large-scale pre-training corpus
(e.g., 1T tokens)

How do such parametric LMs work?

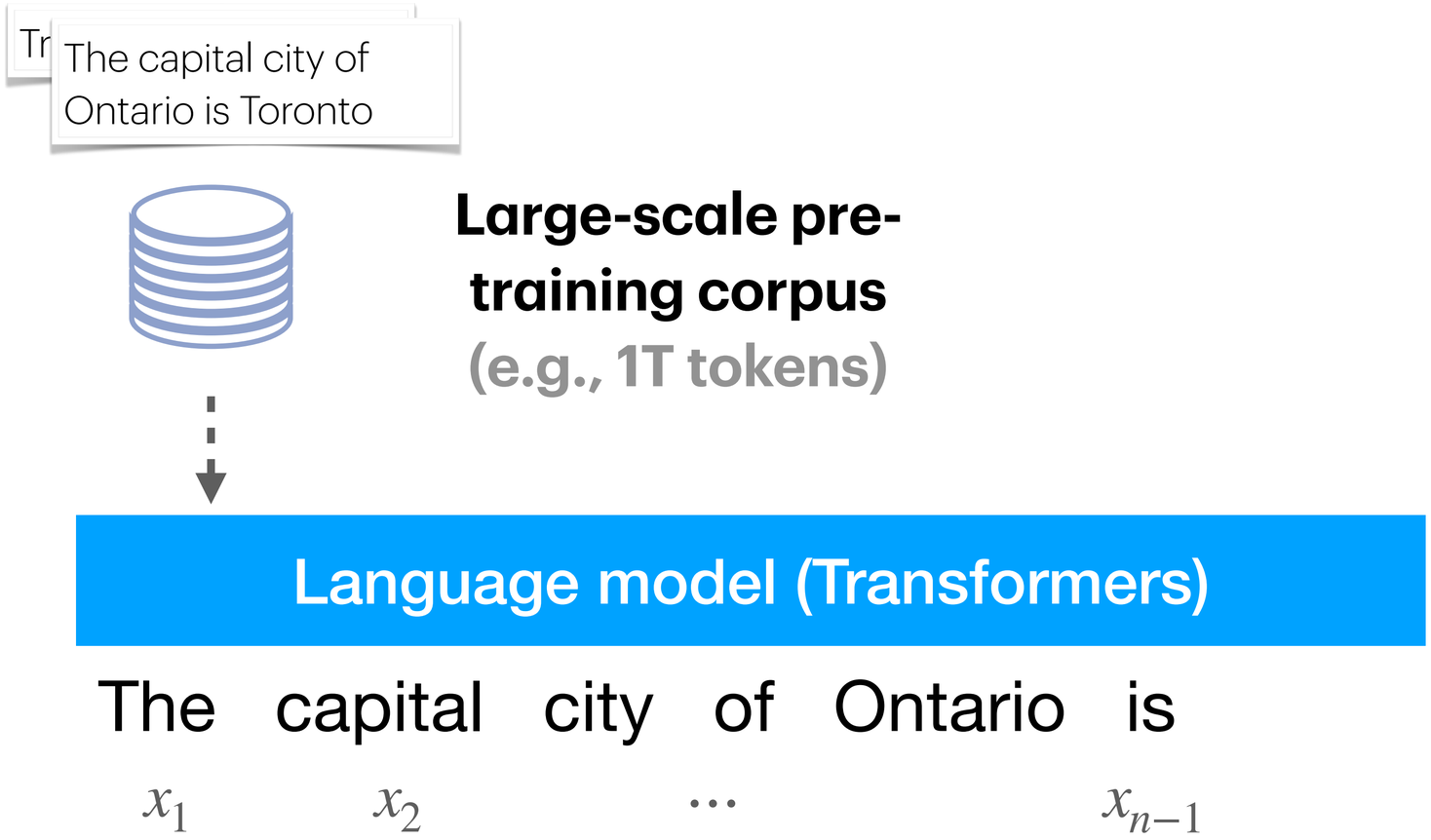
Large-scale pre-training corpus
(e.g., 1T tokens)

How do such parametric LMs work?



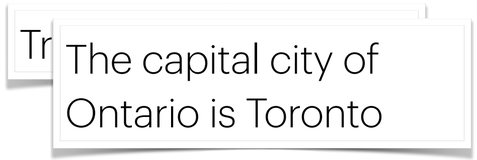
How do such parametric LMs work?

$$P(x_n \mid x_1, x_2, \cdots, x_{n-1})$$

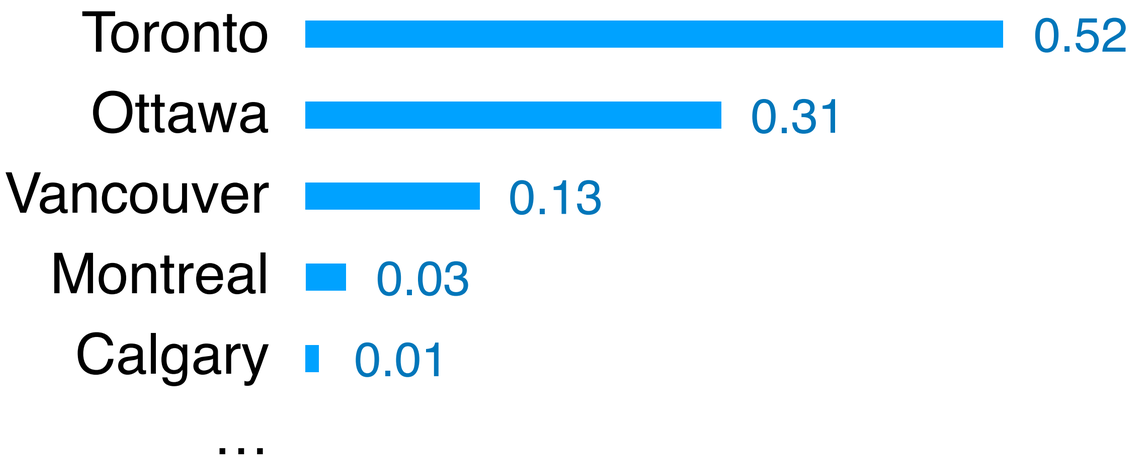


How do such parametric LMs work?

$$P(x_n \mid x_1, x_2, \cdots, x_{n-1})$$



Large-scale pre-training corpus
(e.g., 1T tokens)



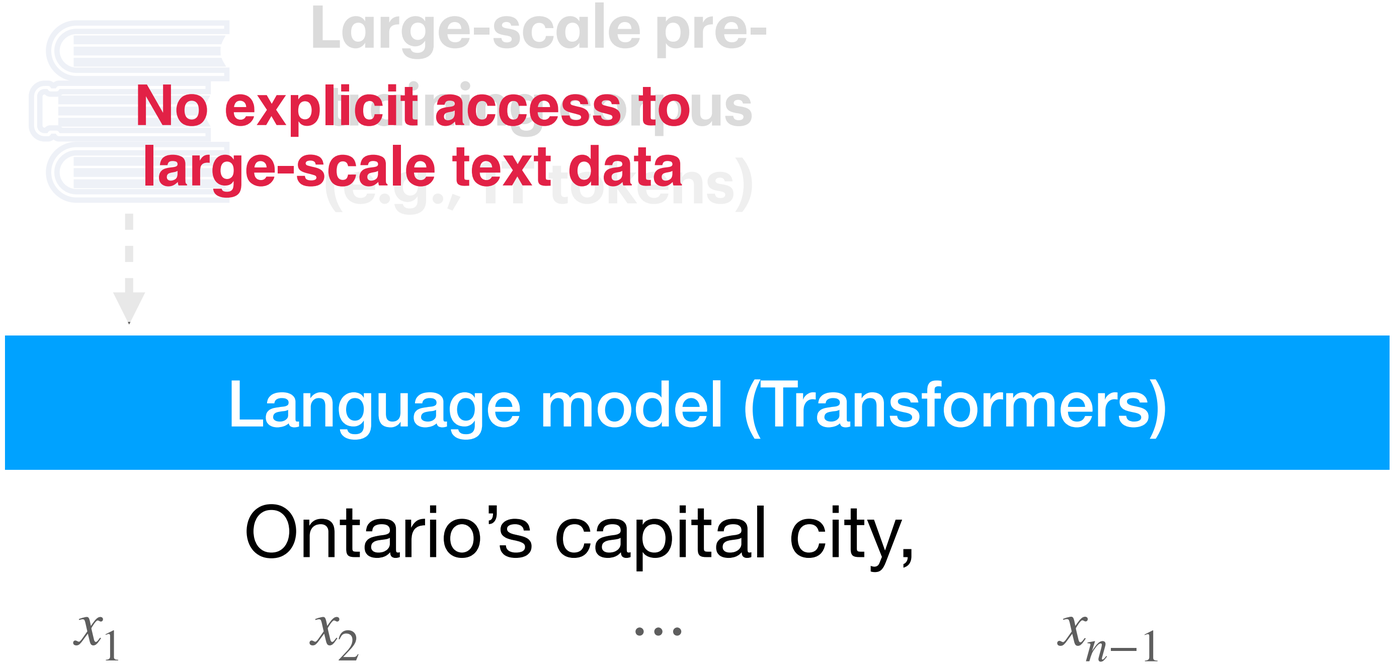
Language model (Transformers)

The capital city of Ontario is

x_1 x_2 … x_{n-1}

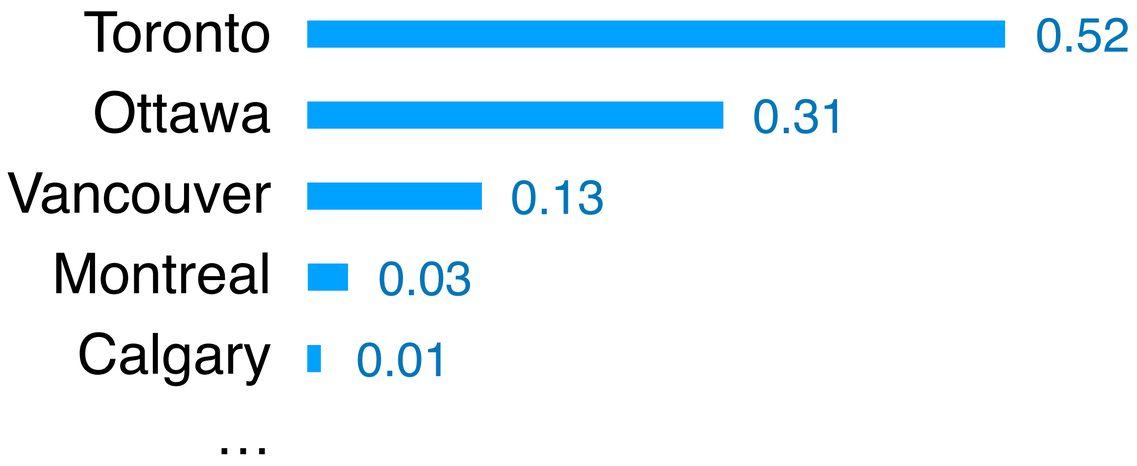
How do such parametric LMs work?

$$P(x_n \mid x_1, x_2, \cdots, x_{n-1})$$



How do such parametric LMs work?

$$P(x_n \mid x_1, x_2, \cdots, x_{n-1})$$



No explicit access to large-scale text data

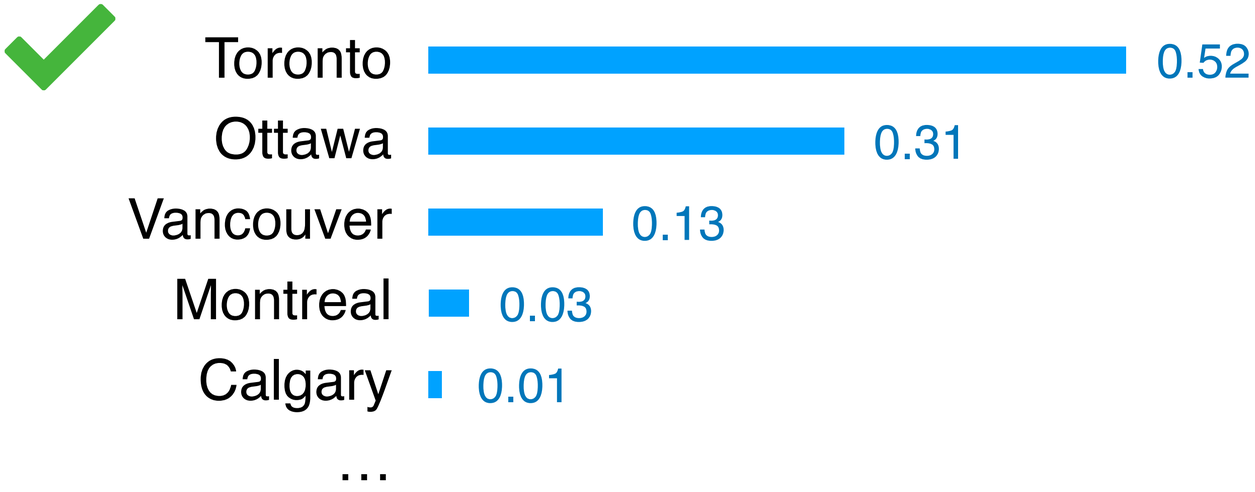
Language model (Transformers)

Ontario's capital city,

x_1 x_2 $\cdots$ x_{n-1}

How do such parametric LMs work?

$$P(x_n \mid x_1, x_2, \cdots, x_{n-1})$$



No explicit access to large-scale text data

Language model (Transformers)

Ontario's capital city,

x_1 x_2 $\cdots$ x_{n-1}

How do such parametric 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 (DeepSeek R1)

Explain David Smith, a computer scientist at Northeastern University, including a list of his famous papers.
Thinking...
Okay, the user wants me to explain David Smith, a computer scientist at Northeastern University, including a list of his famous papers. Let me start by recalling who David Smith is. He's known for his work in programming languages and compilers. But wait, there are several David Smiths in academia, so I need to make sure I'm referring to the right one. Northeastern University's David Smith—let me check his main areas.

Hmm, his research focuses on programming languages, particularly the Rust compiler. That's a big project, so he must be a key contributor. The user might be interested in his technical contributions. But they also asked for a list of famous papers. I should verify which papers are most cited or influential. The Rust project papers from 2019-2020 come to mind. Those are probably the most well-known.

Wait, the user might not just want a list. They might be looking for someone to recommend for a project related to Rust or compilers. Or maybe they're a student considering working with him. Including his role as a principal investigator on the Rust compiler could be important. Also, his position as an Associate Professor—maybe the user is interested in academic collaborations or job opportunities.

I should structure the explanation to highlight his main contributions first, then the papers. But how detailed should the papers be? The user might not need every paper, just the most significant ones. Let me list the Rust papers and maybe mention others like the ownership system. But wait, are those papers standalone or part of the Rust project? The ownership system is a key part of Rust, so including that would add value.

Also, considering the user's possible deeper needs: they might be researching for a paper, looking for references, or seeking an expert in the field. Including his affiliation with Northeastern and his role in Rust could help others find his contact or his lab. Maybe they want to know his impact in the industry too, since Rust is widely used.

I need to make sure the information is accurate. Let me cross-check the publication years and authors. The Rust papers from 2019-2020 were indeed major releases. The ownership system paper is from 2021. Other papers like the borrow checker might be part of the same body of work. Should I mention his other roles, like being on the Rust team? Yes, that adds credibility.

Wait, the user might not know that David Smith is part of the Rust team. Including that could be useful. Also, explaining the significance of each paper briefly would help. For example, the 2019 paper introduced key features, the 2020 one was about incremental compilation, which is crucial for large projects. The ownership system paper details the language's memory safety, a big deal in systems programming.

The hallucination continues

Okay, let's break down David Smith, a prominent computer scientist at Northeastern University.

Who is David Smith (Northeastern University)?

* **Position:** He is an Associate Professor of the Practice in the Department of Computer Science.
* **Research Area:** His primary research focus is on Programming Languages and Compilers.
* **Significance:** He is one of the **Principal Investigators (PIs)** on the **Rust programming language project** at Northeastern University. This is one of the most influential projects in modern systems programming.
* **Key Contributions:** His work is deeply involved in the design, implementation, and evolution of the Rust compiler and language. He contributes significantly to the core infrastructure, language features, and tooling of Rust.

Why is David Smith Important?

His importance stems directly from his leadership and technical contributions to the **Rust programming language**. Rust has gained immense popularity for its unique combination of performance, safety (guaranteeing memory safety without a garbage collector), and parallelism, making it a cornerstone for building reliable and efficient systems software. Working as a PI on such a high-profile and impactful open-source project places him at the forefront of a major trend in computer science.

List of His Notable Papers (Primarily Co-authored, Focusing on Rust)

While David Smith is part of a large team contributing to Rust, several key publications stand out, often representing major releases or significant architectural changes to the language/compiler:

1. **Smith, David; et al.** (Rust Team). "Rust Programming Language." *Proceedings of the ACM Programming Languages*, 2019. (Often cited for the 1.0 release or major features introduced around that time, e.g., const generics preview, improved Cargo).
   * *Significance:* Represents the foundational work and key features solidifying Rust's capabilities.
2. **Smith, David; et al.** (Rust Team). "Rust 2021: An Update on the Revision of the Rust Programming Language." *Proceedings of the ACM Programming Languages*, 2020. (Often refers to the 2021 language release cycle).

ChatGPT hallucinates

Describe David Smith, a computer scientist at Northeastern University, including some of his most important papers.

David Smith is a well-respected computer scientist who is associated with Northeastern University. His research primarily revolves around areas like **data science, machine learning, natural language processing (NLP),** and **computational linguistics.** While specific personal or career details might vary over time, he is known for his work in fields that bridge the gap between theoretical foundations and practical applications.

Some of his significant contributions can be linked to:

1. Natural Language Processing (NLP) and Text Mining

David Smith has worked on computational techniques for processing human language, a cornerstone of NLP. His research often combines computational models with large-scale data to improve language understanding and machine learning processes.

Some Key Papers (and Topics) Associated with David Smith:

- **"The Oxford Dictionary of Computational Linguistics"**
  This is a significant contribution to the field of computational linguistics. While Smith might not be the sole author, his involvement in such works has helped define technical terminology and concepts that are essential for the development of language-processing tools and systems.
- **"Fast, Scalable Construction of Parsing Models"**
  This paper discusses efficient ways to build computational models for parsing sentences in natural language, helping algorithms understand syntax more effectively and at scale.
- **"Topic Models: A Survey"**
  Smith contributed to the survey of topic models, which are critical for understanding large corpora of text by automatically discovering topics within. This has widespread applications, from document clustering to recommendation systems.
- **"Statistical Approaches to Natural Language Processing"**
  This paper outlines various statistical models used for language understanding. It highlights the shift from rule-based systems to statistical models and is foundational for modern NLP research.

ChatGPT hallucinates

Describe David Smith, a computer scientist at Northeastern University, including some of his most important papers.

David Smith is a well-respected computer scientist who is associated with Northeastern University. His research primarily revolves around areas like **data science, machine learning, natural language processing (NLP),** and **computational linguistics.** While specific personal or career details might vary over time, he is known for his work in fields that bridge the gap between theoretical foundations and practical applications.

Some of his significant contributions can be linked to:

Natural Language Processing (NLP) and Text Mining

David Smith has worked on computational techniques for processing human language, a cornerstone of NLP. His research often combines computational models with large-scale data to improve language understanding and machine learning processes.

Some Key Papers (and Topics) Associated with David Smith:

"The Oxford Dictionary of Computational Linguistics"
This is a significant contribution to the field of computational linguistics. While Smith might not be the sole author, his involvement in such works has helped define technical terminology and concepts that are essential for the development of language-processing tools and systems.

"Fast, Scalable Construction of Parsing Models"
This paper discusses efficient ways to build computational models for parsing sentences in natural language, helping algorithms understand syntax more effectively and at scale.

"Topic Models: A Survey"
Smith contributed to the survey of topic models, which are critical for understanding large corpora of text by automatically discovering topics within. This has widespread applications, from document clustering to recommendation systems.

"Statistical Approaches to Natural Language Processing"
This paper outlines various statistical models used for language understanding. It highlights the shift from rule-based systems to statistical models and is foundational for modern NLP research.

When I log in, ChatGPT searches the web, and then…

- **"Two-stage query segmentation for information retrieval."** M. Bendersky, W. Bruce Croft, D. A. Smith. In SIGIR, 2009. Khoury College c...
  - Contribution: In the IR domain, proposed segmenting user queries into meaningful phrases (two-stage segmentation) to improve retrieval performance.
  - Significance: Bridges IR (information retrieval) and NLP, showing Smith's cross-disciplinary reach.
- **"Automatic collation for diversifying corpora: Commonly copied texts as distant supervision for handwritten text recognition."** D. A. Smith, J. Murel, J. Parkes Allen, M. T. Miller. CHR 2023. Khoury College c... +1
  - Contribution: Addresses historical-print handwritten text recognition, leveraging automatic collation (aligning repeated texts) to create supervision for recognition models.
  - Significance: Demonstrates how NLP methods can be applied to digital humanities (e.g., manuscript studies) where labelled data is scarce.

These are just a representative sample; his publication list is quite broad, covering topics from syntactic parsing to retrieval, to digital humanities tasks.

Impact & relevance

- Smith's work has contributed to core NLP methodologies (e.g., parsing, latent structure) and expanded into domains such as IR and digital humanities.
- His position at the NULab shows his interest in interdisciplinary work — combining text analysis, spatial/network data, and social science.
- By bridging "pure" NLP/ML and the humanities/digital libraries, his research has been influential for scholars working at that intersection.
- His teaching includes courses such as Natural Language Processing (CS6120) and Information

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 fined.

ERIC MCGREGOR—LIGHTROCKET/GETTY IMAGES

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 fined.
ERIC MCGREGOR—LIGHTROCKET/GETTY IMAGES

Air Canada must honor refund policy invented by airline's chatbot

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

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

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

ERIC 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

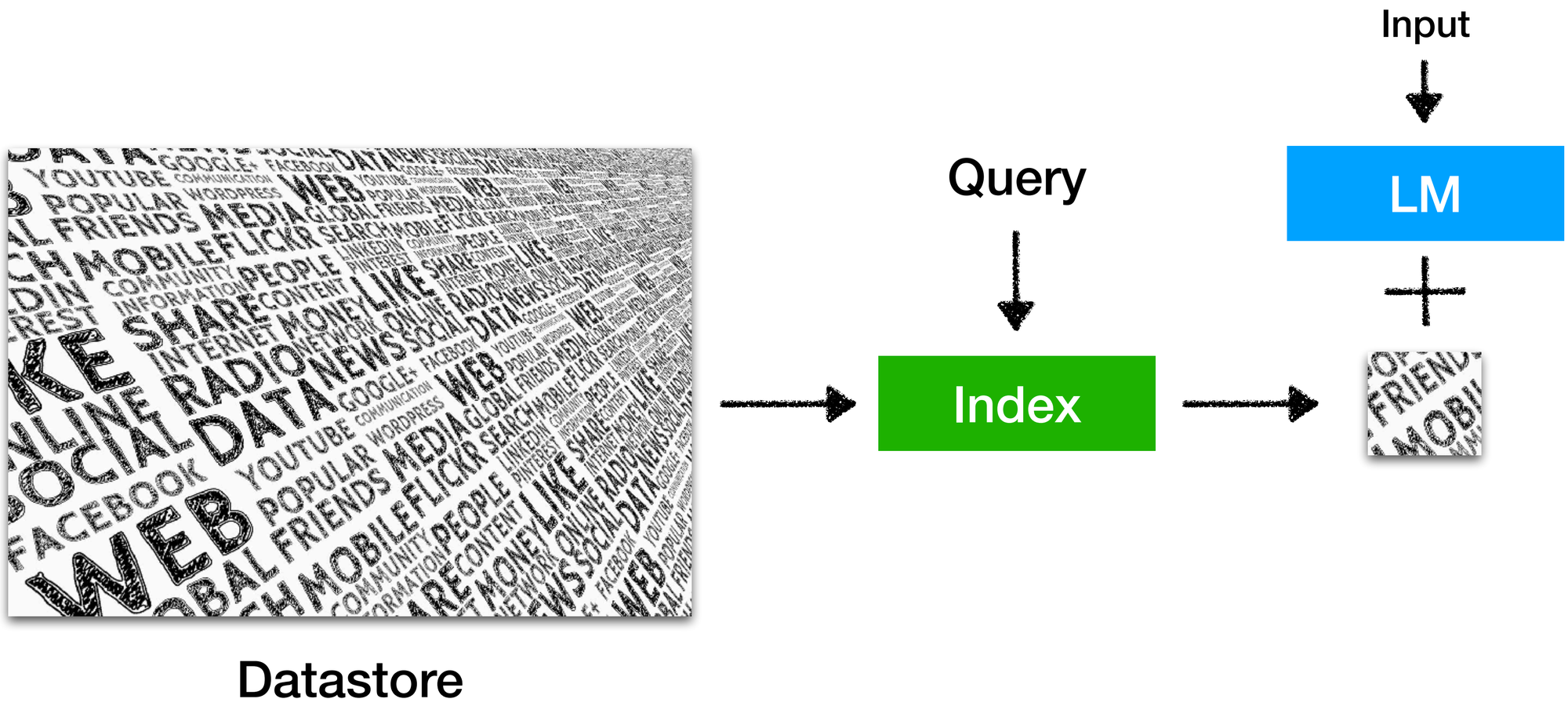
Training time

The capital city of Ontario is _____

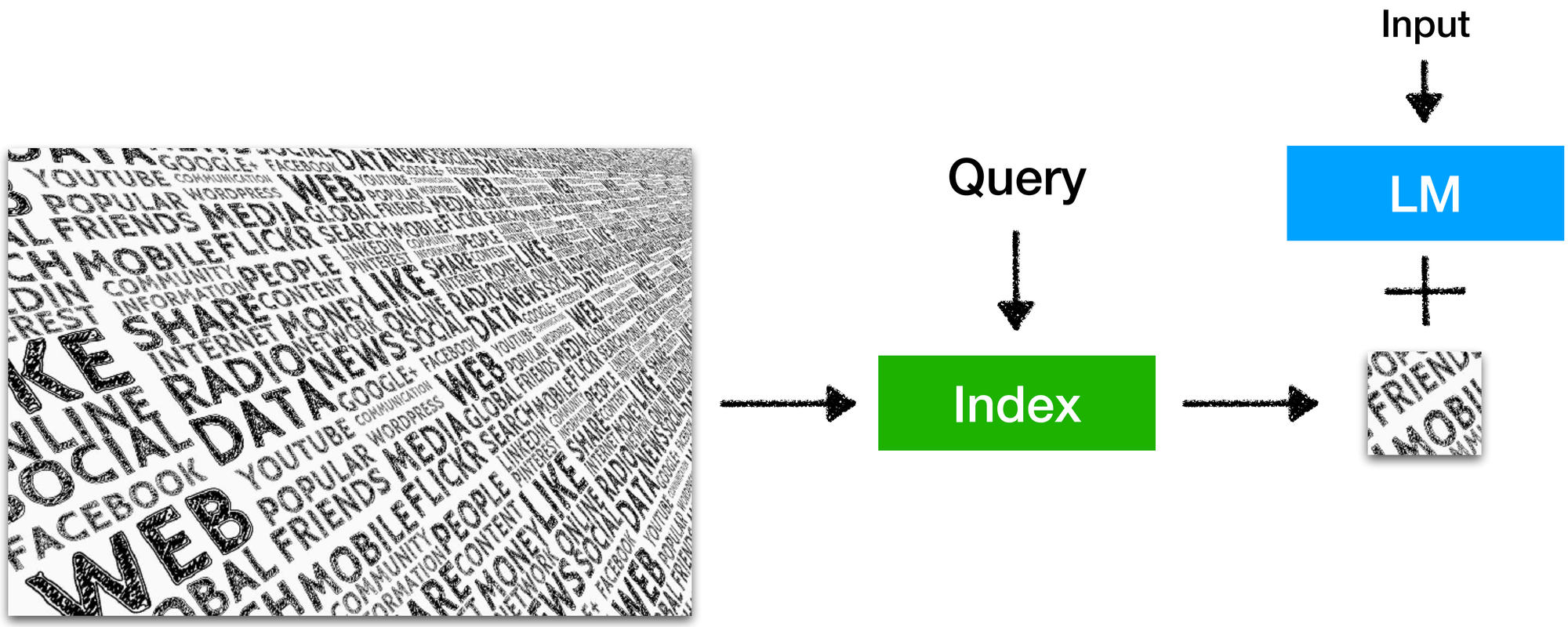
LM

Test time

Inference



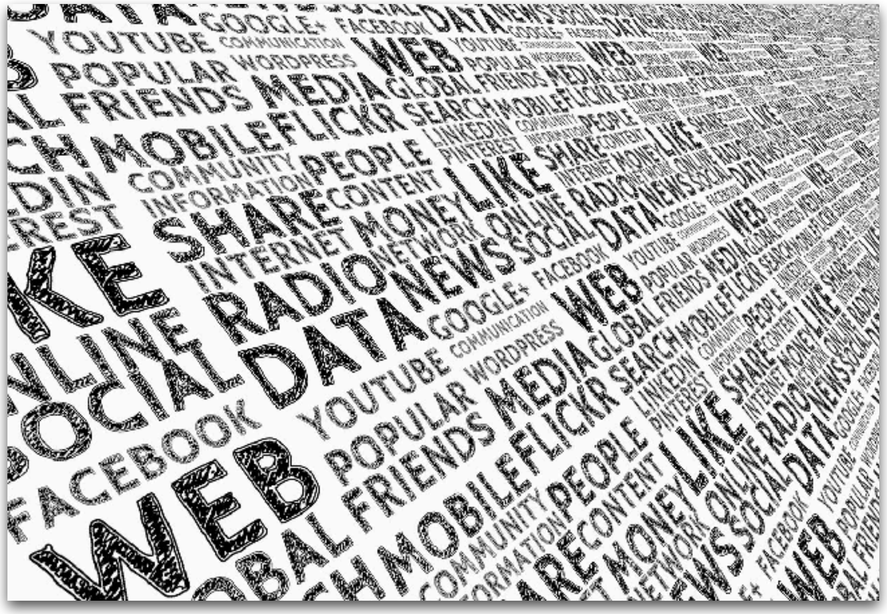
Inference: Datastore



Datastore

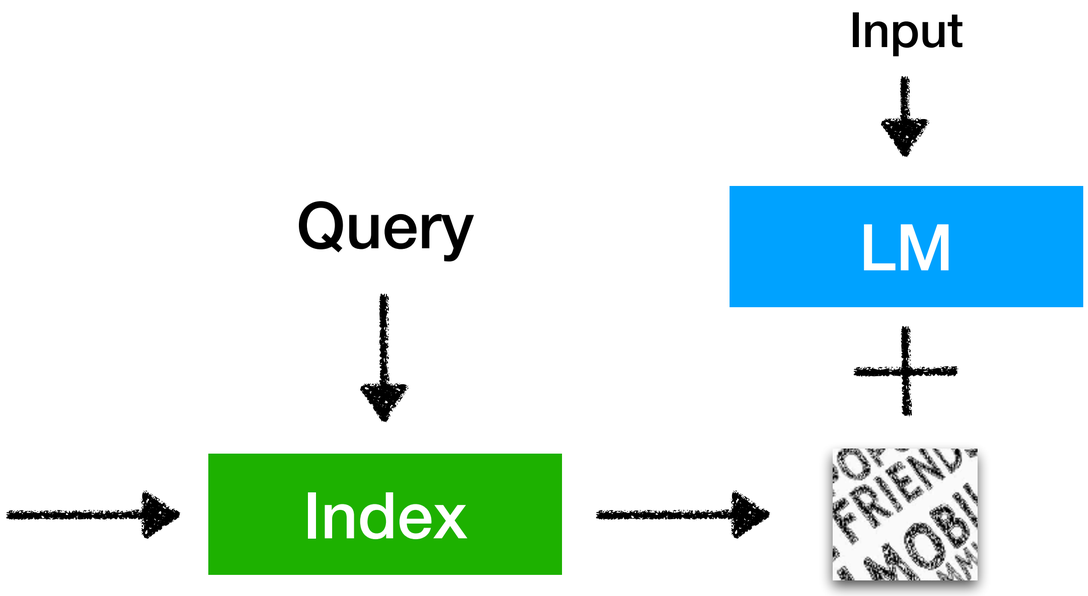
Raw text corpus

Inference: Datastore



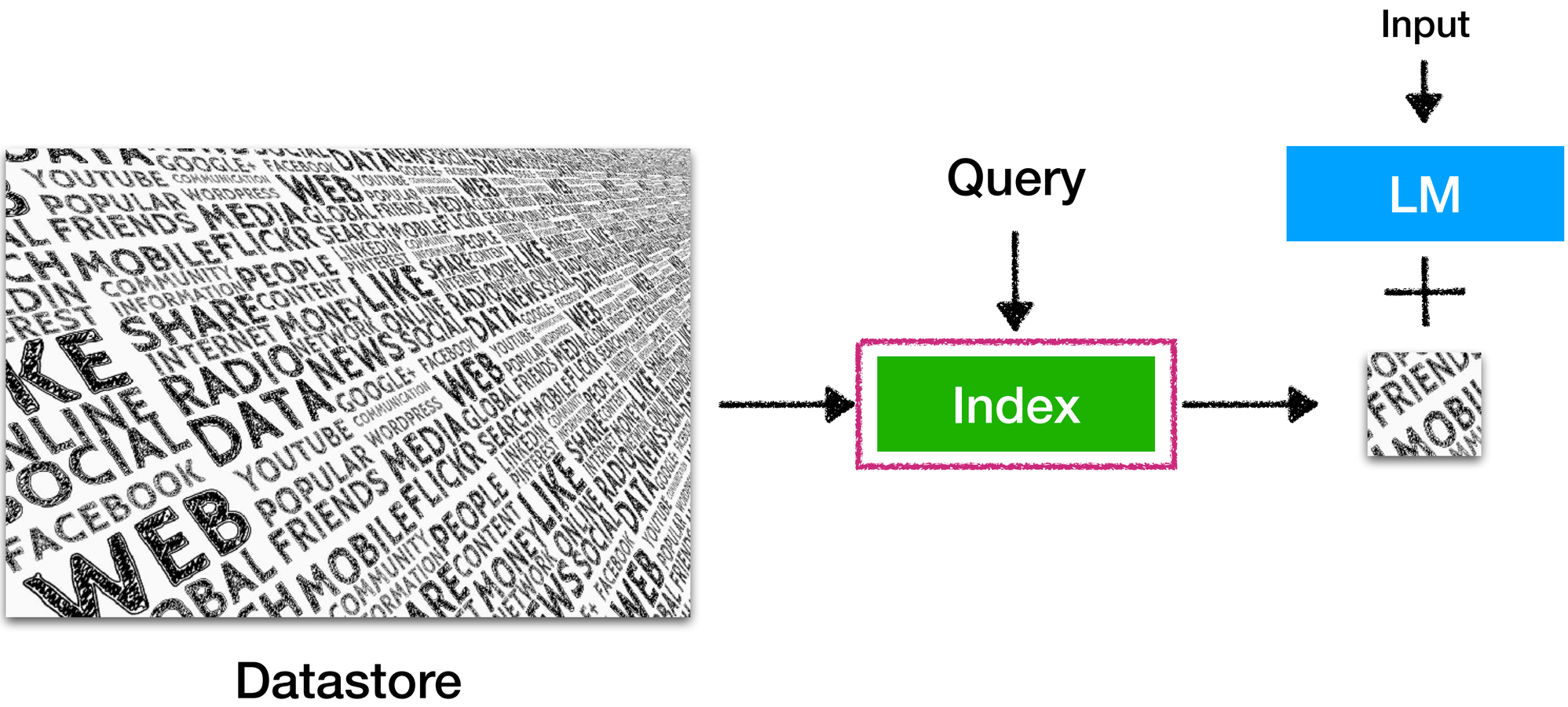
Datastore

Raw text corpus

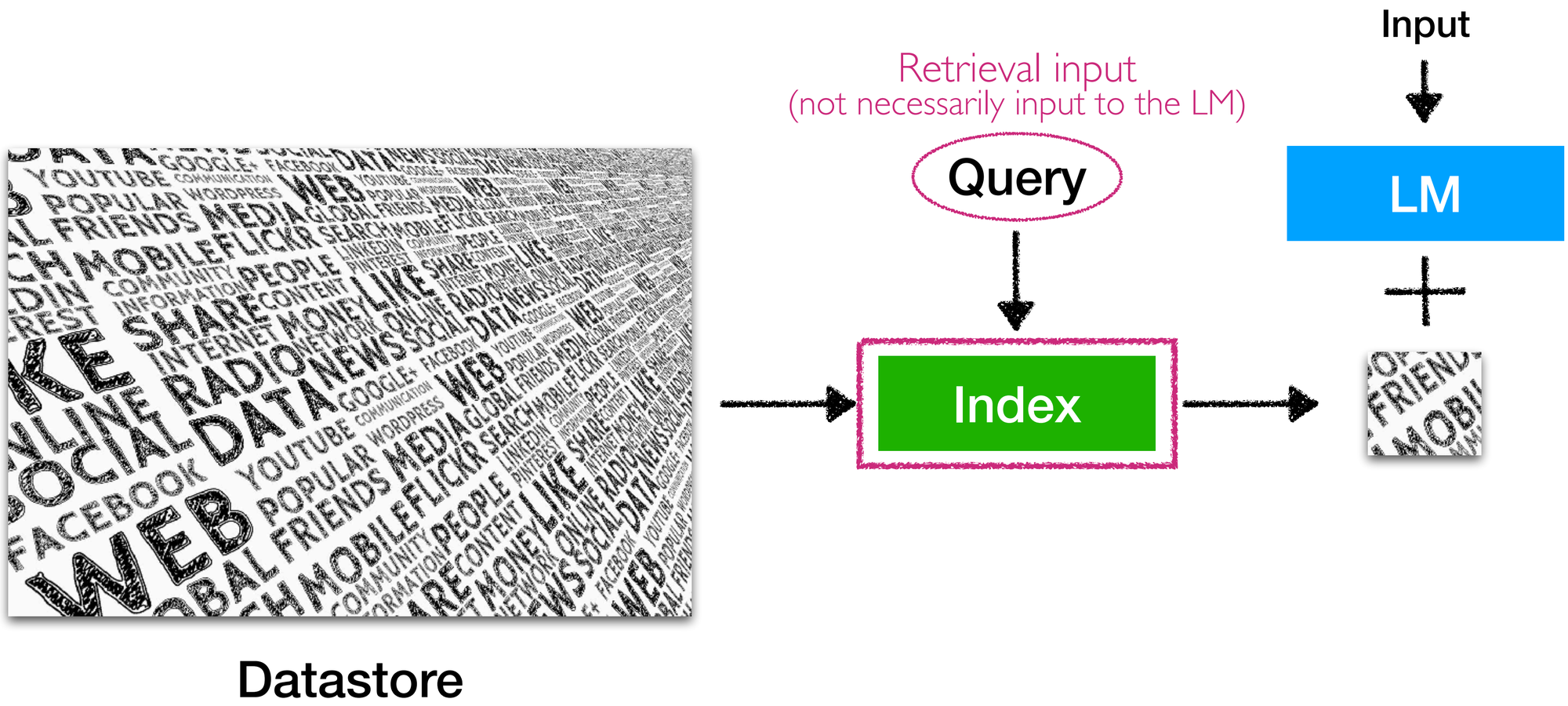


At least billions~trillions of tokens
Not labeled datasets
Not structured data (knowledge bases)

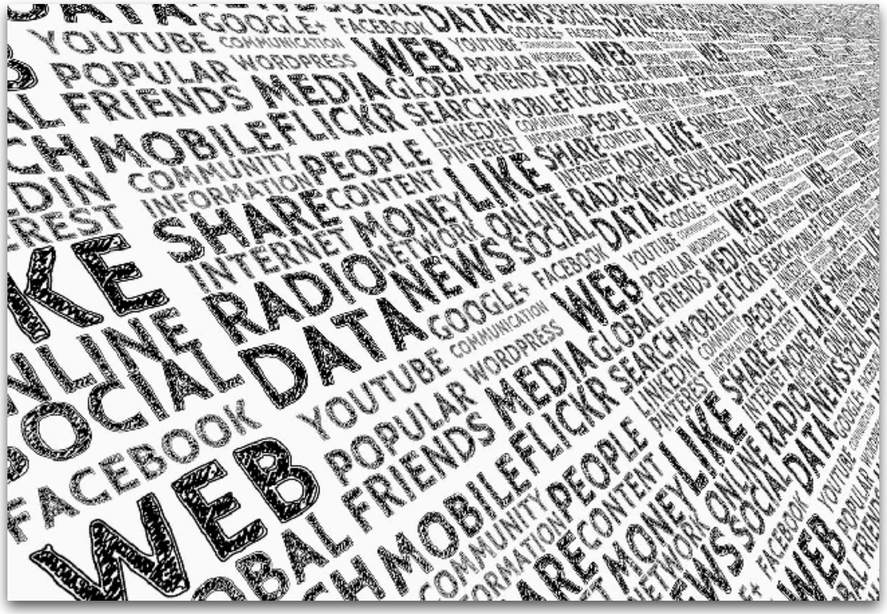
Inference: Index



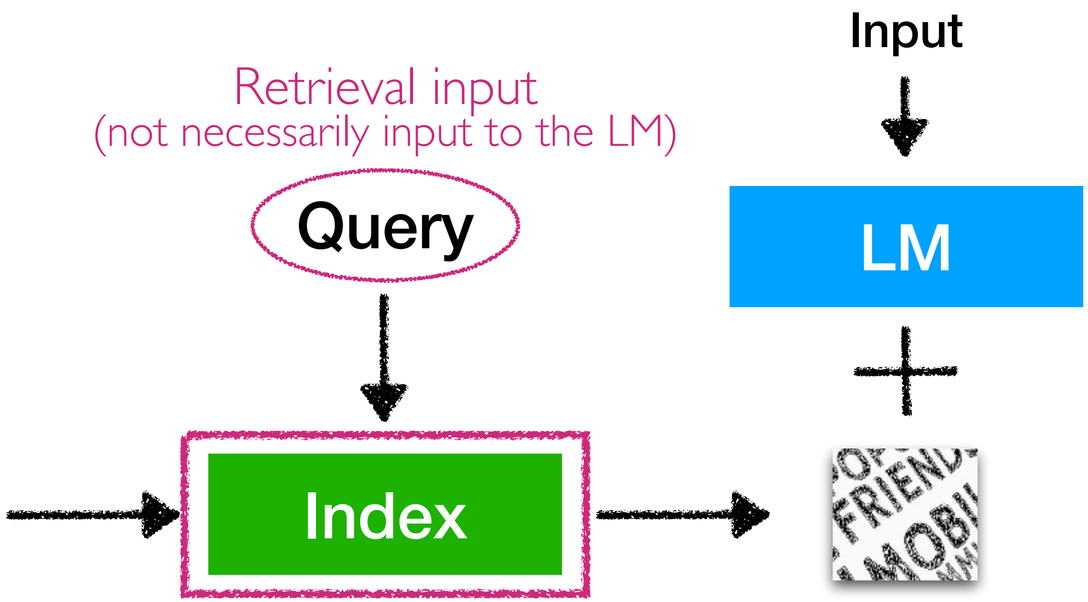
Inference: Index



Inference: Index

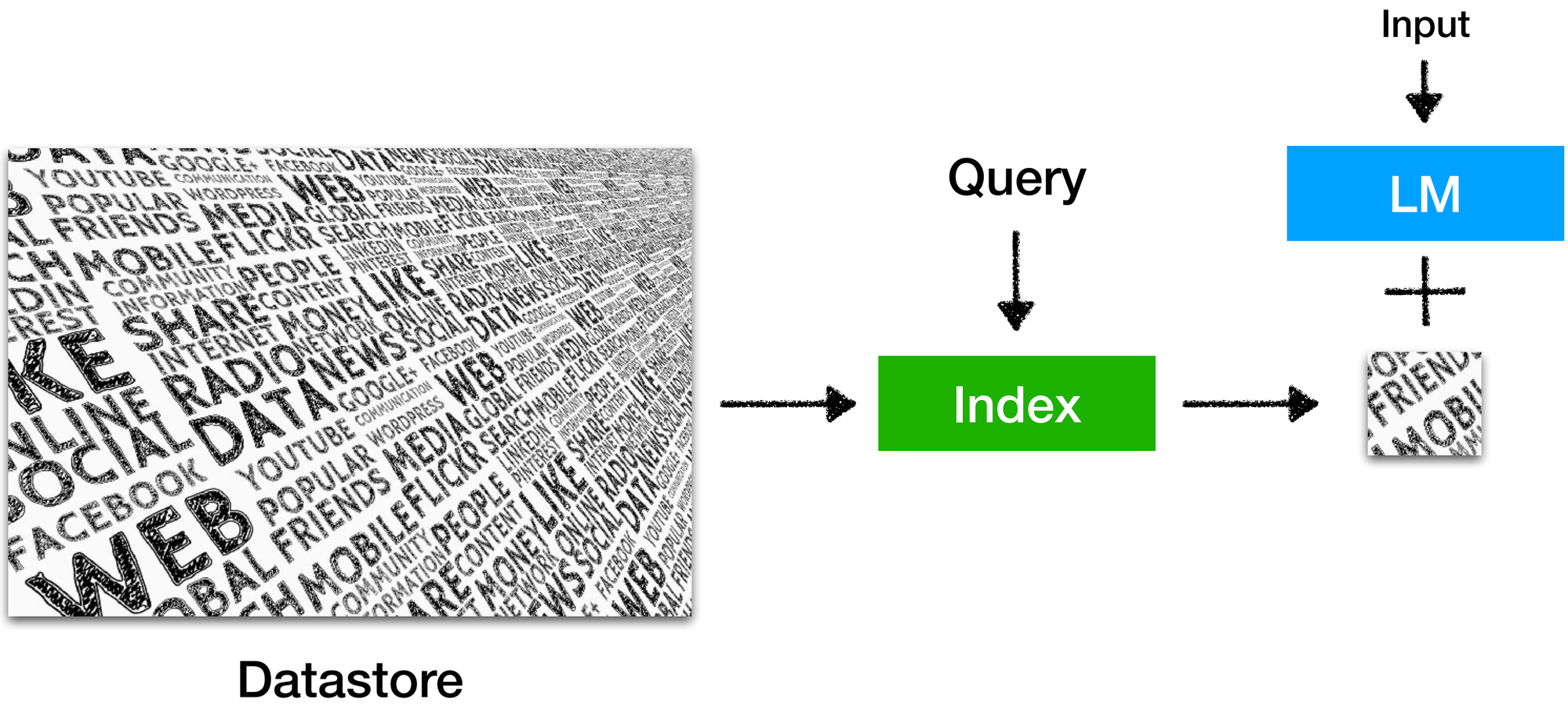


Datastore

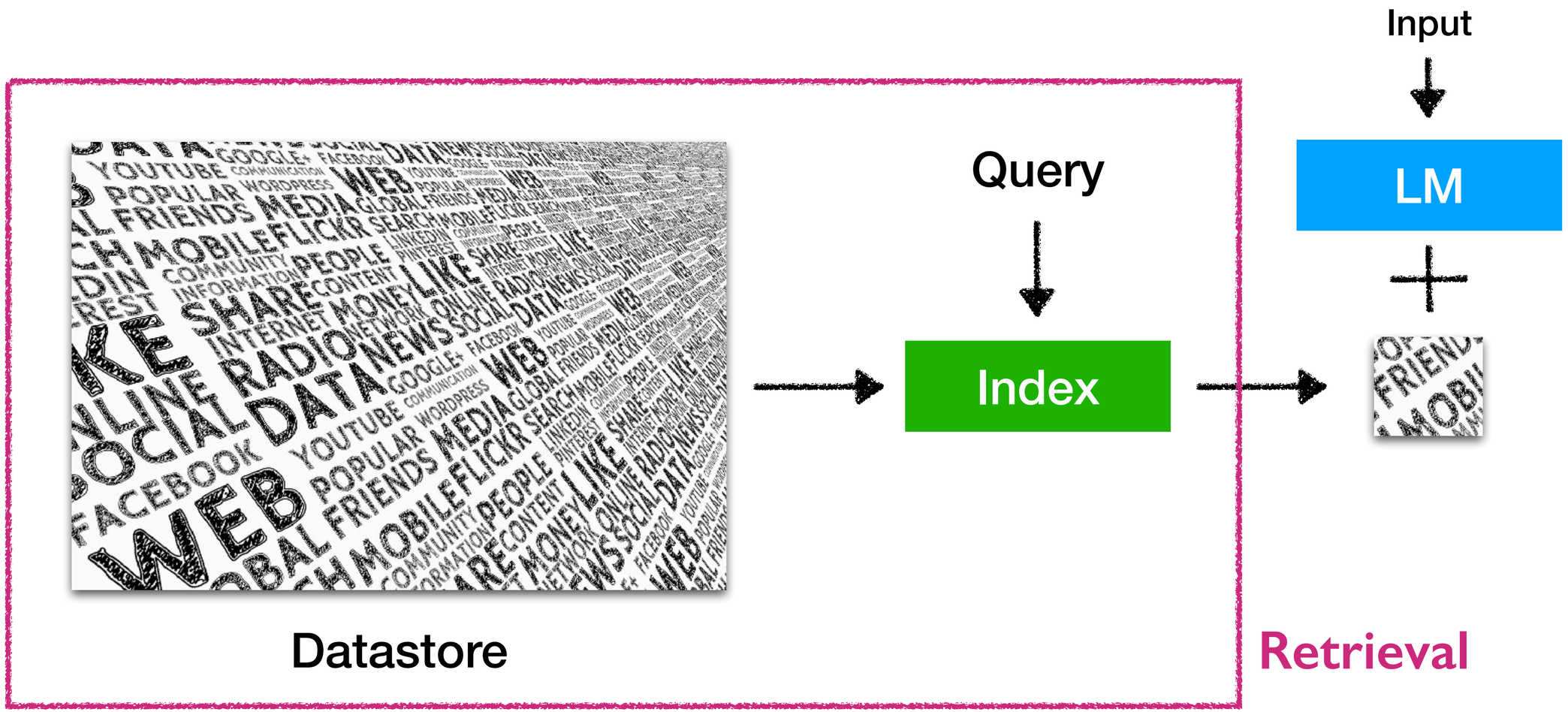


Find a small subset of elements in a datastore that are the most similar to the query

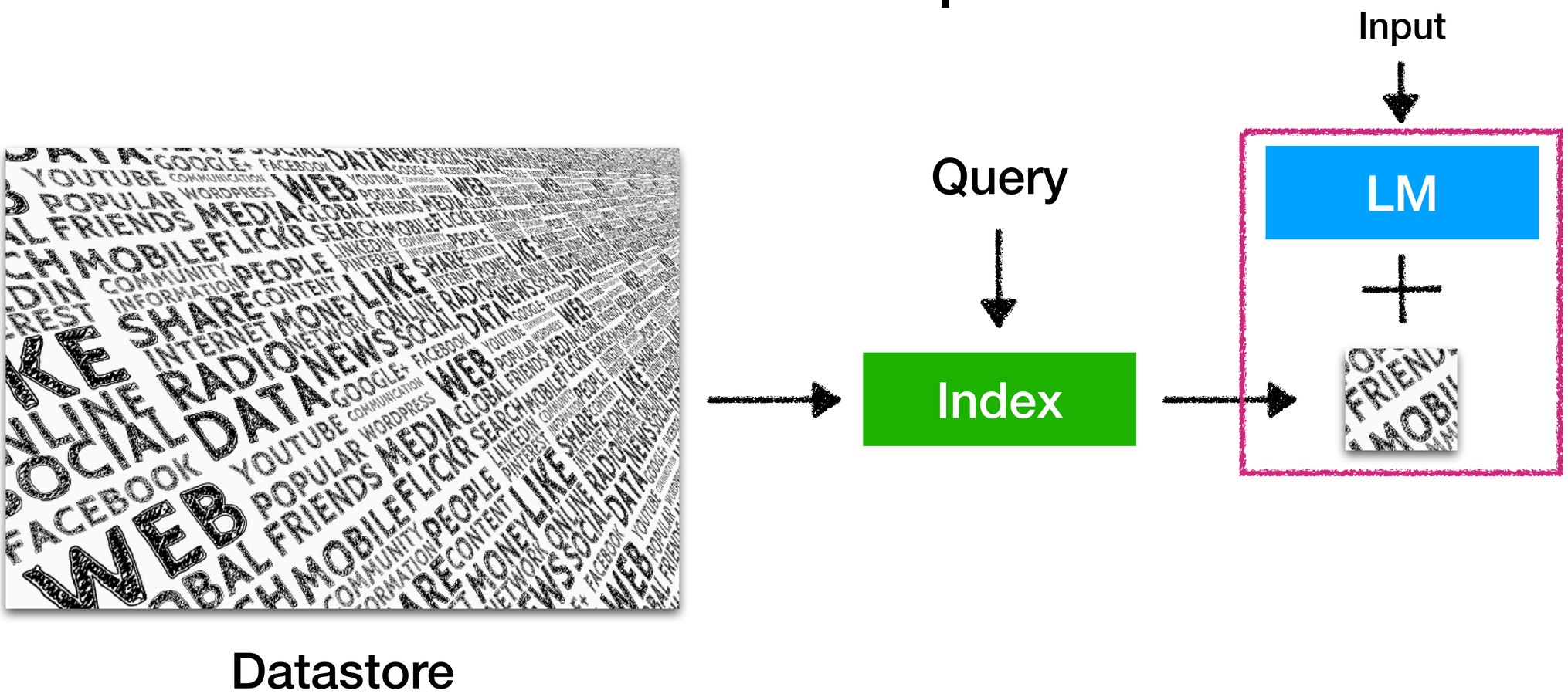
Inference: Search



Inference: Search



Inference: Incorporation



Q: Why should we use retrieval-augmented language models?

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

Problems with Parametric LMs

- **Accuracy issues:**
  - *Knowledge cutoffs*: parameters are usually only updated to a particular time
  - *Private data*: data stored in private text or data repositories not suitable for training
  - *Learning failures*: even for data that the model was trained on, it might not be sufficient to get the right answer
- **Verifiability issues:** It is hard to tell if the answer is correct
  - *Information asymmetry* between user and opaque parameters in an LM

Core limitations of parametric LMs

Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Some Key Papers (and Topics) Associated with David Smith:

"The Oxford Dictionary of Computational Linguistics"
This is a significant contribution to the field of computational linguistics. While Smith might not be the sole author, his involvement in such works has helped define technical terminology and concepts that are essential for the development of language-processing tools and systems.

"Fast, Scalable Construction of Parsing Models"
This paper discusses efficient ways to build computational models for parsing sentences in natural language, helping algorithms understand syntax more effectively and at scale.

"Topic Models: A Survey"
Smith contributed to the survey of topic models, which are critical for understanding large corpora of text by automatically discovering topics within. This has widespread applications, from document clustering to recommendation systems.

"Statistical Approaches to Natural Language Processing"
This paper outlines various statistical models used for language understanding. It highlights the shift from rule-based systems to statistical models and is foundational for modern NLP research.

Core limitations of parametric LMs

Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Explain David Smith, including the list of his famous papers.

↓

Language model

↓

His most famous paper is "*Rust 2021: An Update on the Revision of the Rust Programming Language.*"

Core limitations of parametric LMs

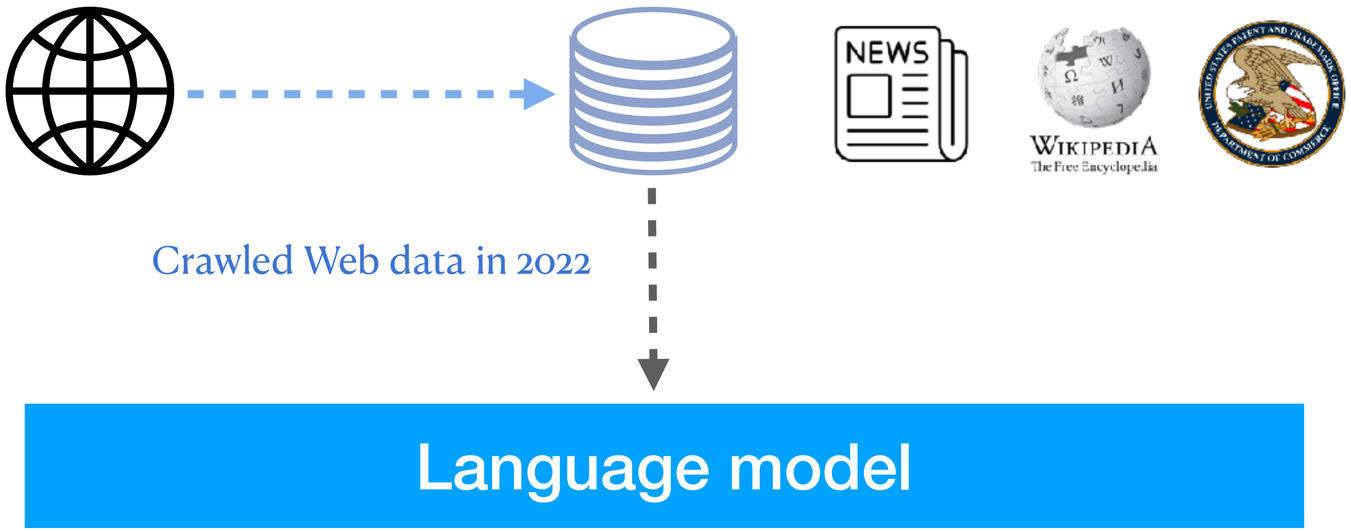
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



Core limitations of parametric LMs

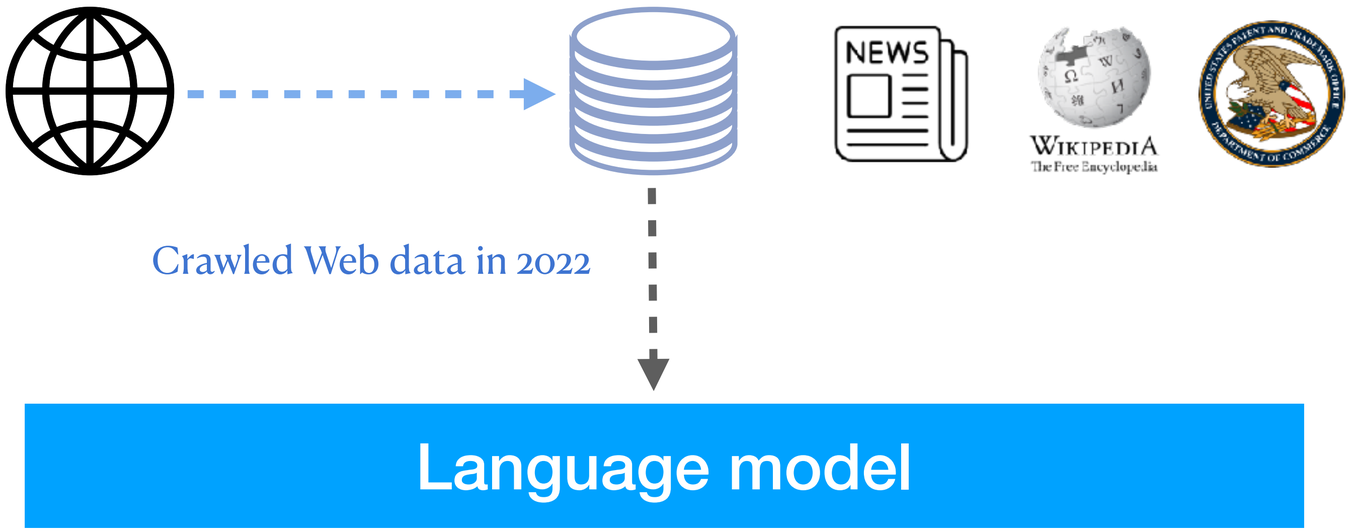
Hallucinations

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

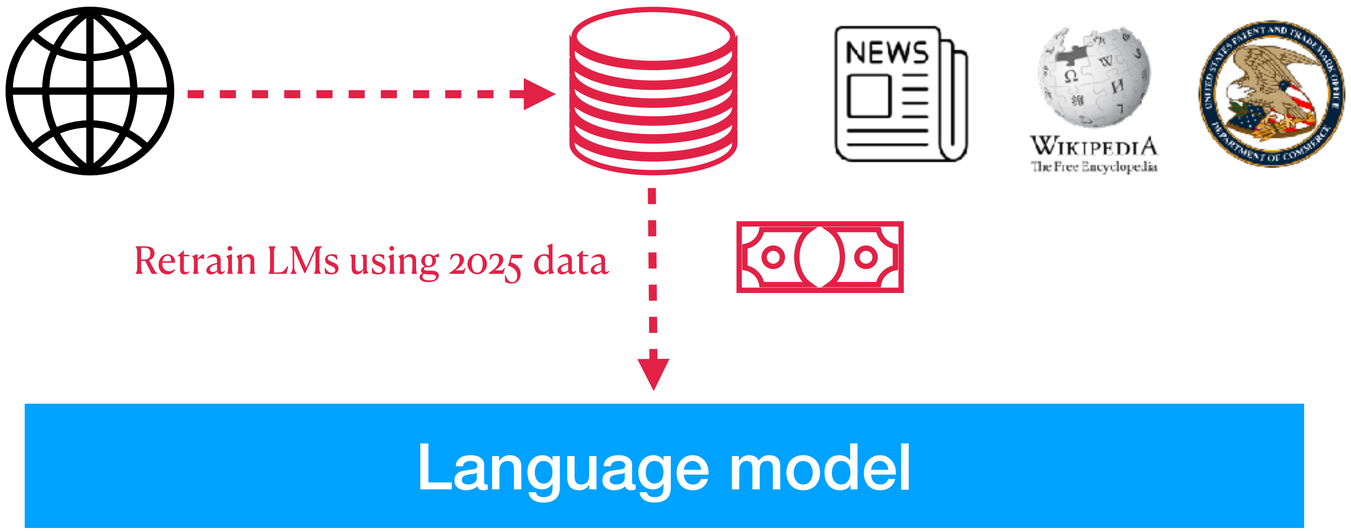
Hallucinations

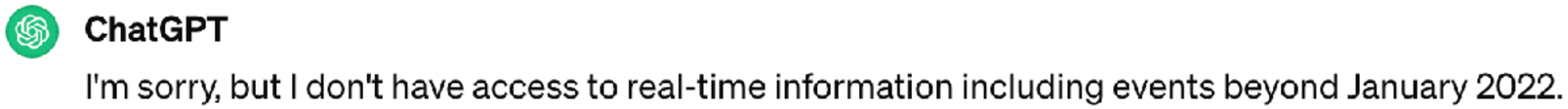
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size





Core limitations of parametric LMs

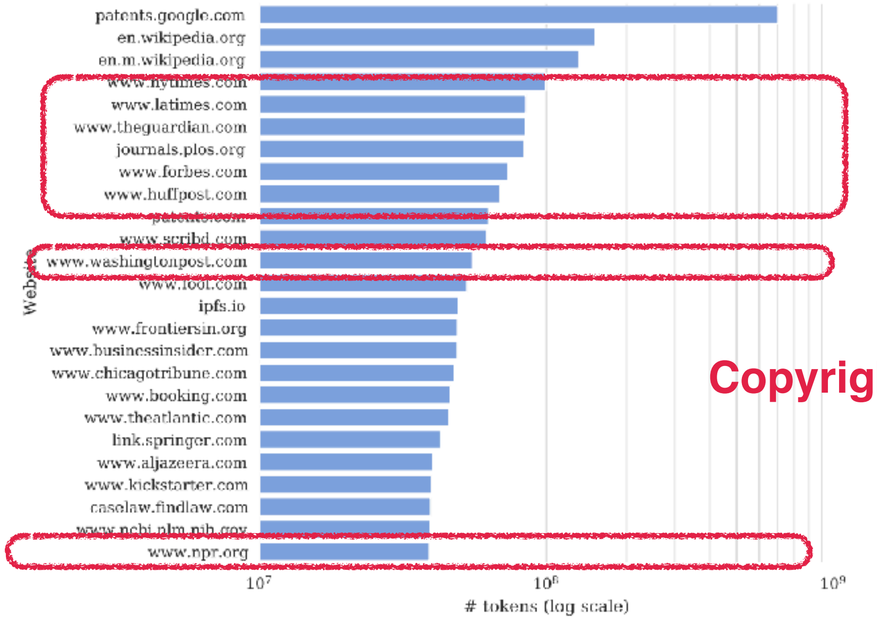
Hallucinations

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

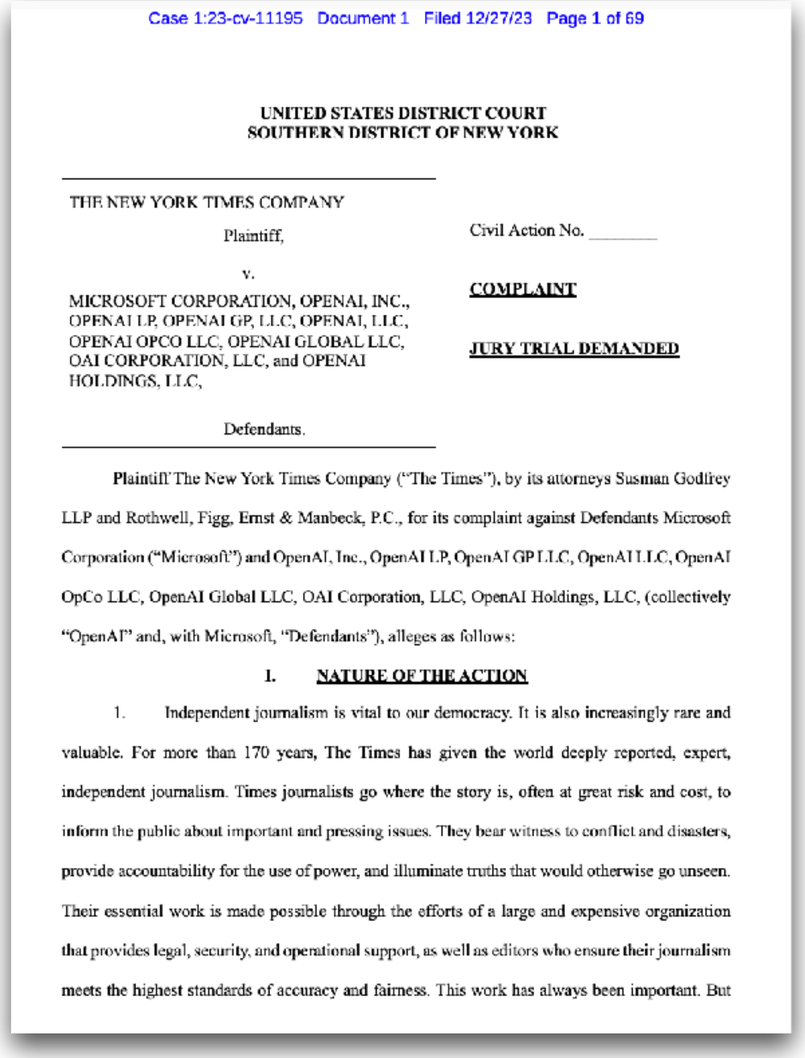
Hallucinations

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

UNITED STATES DISTRICT COURT
SOUTHERN DISTRICT OF NEW YORK

THE NEW YORK TIMES COMPANY

Plaintiff,

v.

MICROSOFT CORPORATION, OPENAI, INC., OPENAI LP, OPENAI GP, LLC, OPENAI, LLC, OPENAI OPCO LLC, OPENAI GLOBAL LLC, OAI CORPORATION, LLC, and OPENAI HOLDINGS, LLC,

Defendants.

Civil Action No. _____

COMPLAINT

JURY TRIAL DEMANDED

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

New York Times lawsuits against OpenAI

Core limitations of parametric LMs

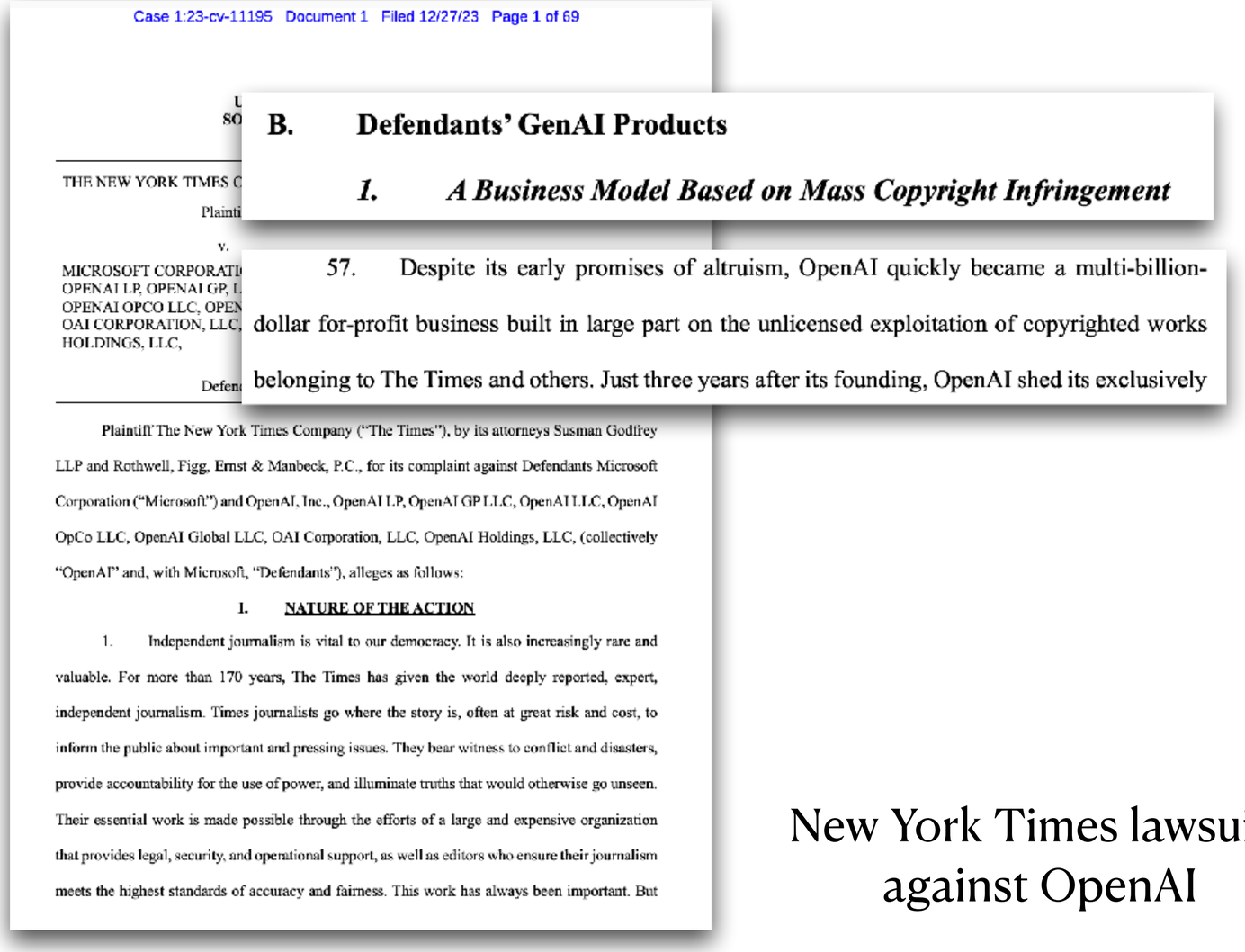
Hallucinations

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

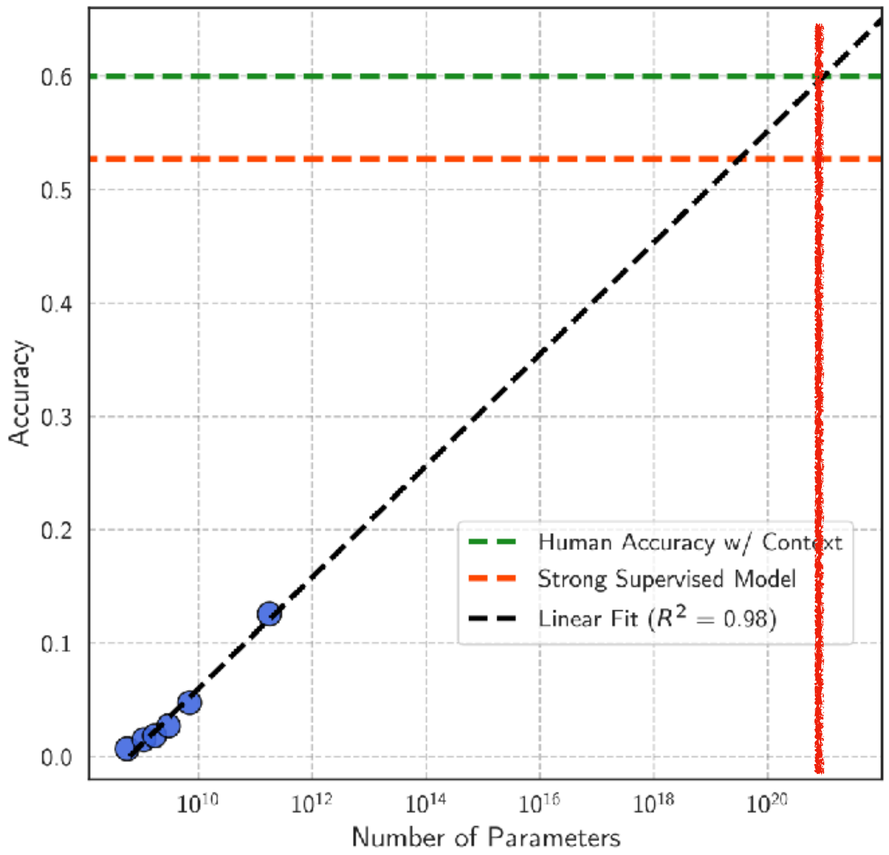
Lack of attributions

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

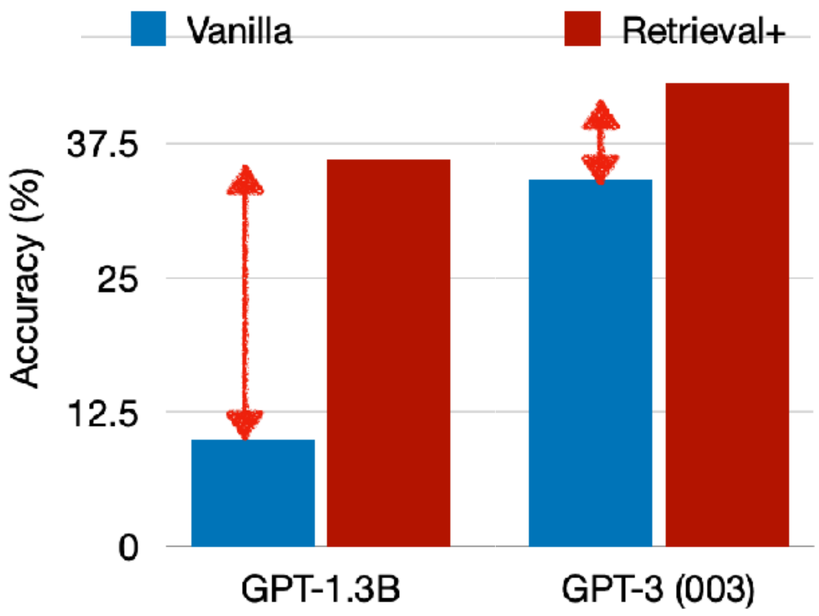
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

QA



Mallen et al. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories 2023.

How retrieval-augmented LMs solve the issues?

Hallucinations

Lack of attributions

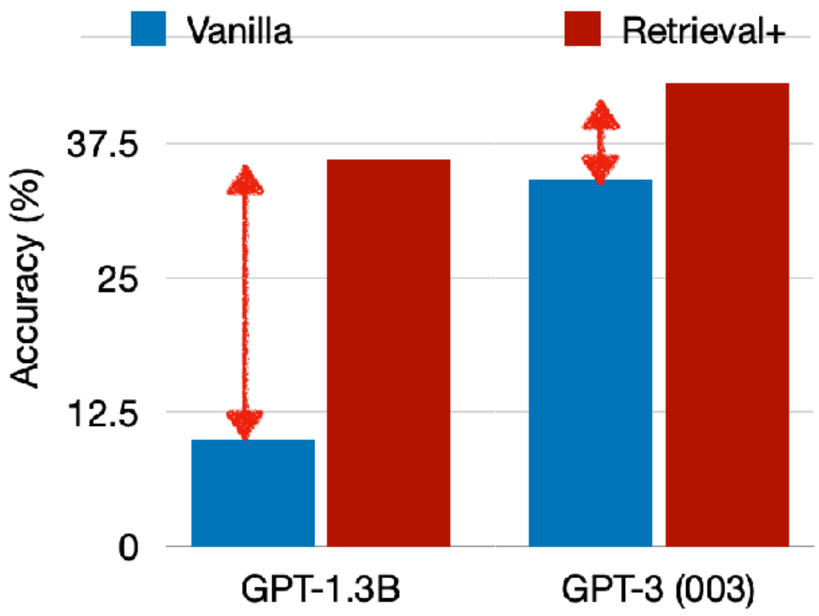
Costs of adaptations

Copyright / privacy

Large parameter size

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

QA



Mallen et al. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories 2023.

How retrieval-augmented LMs solve the issues?

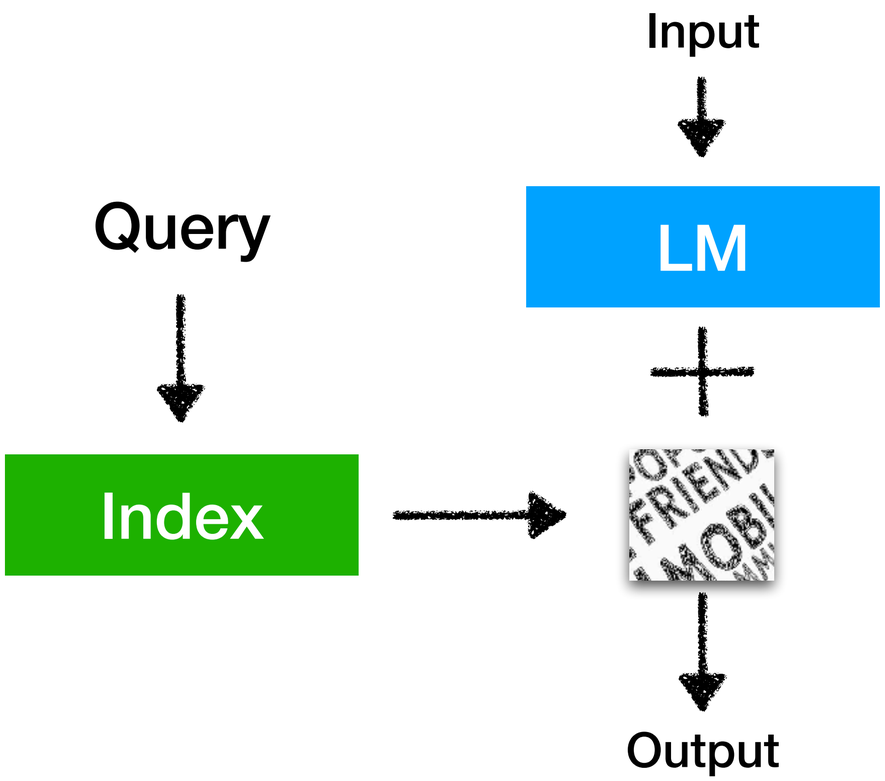
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



How retrieval-augmented LMs solve the issues?

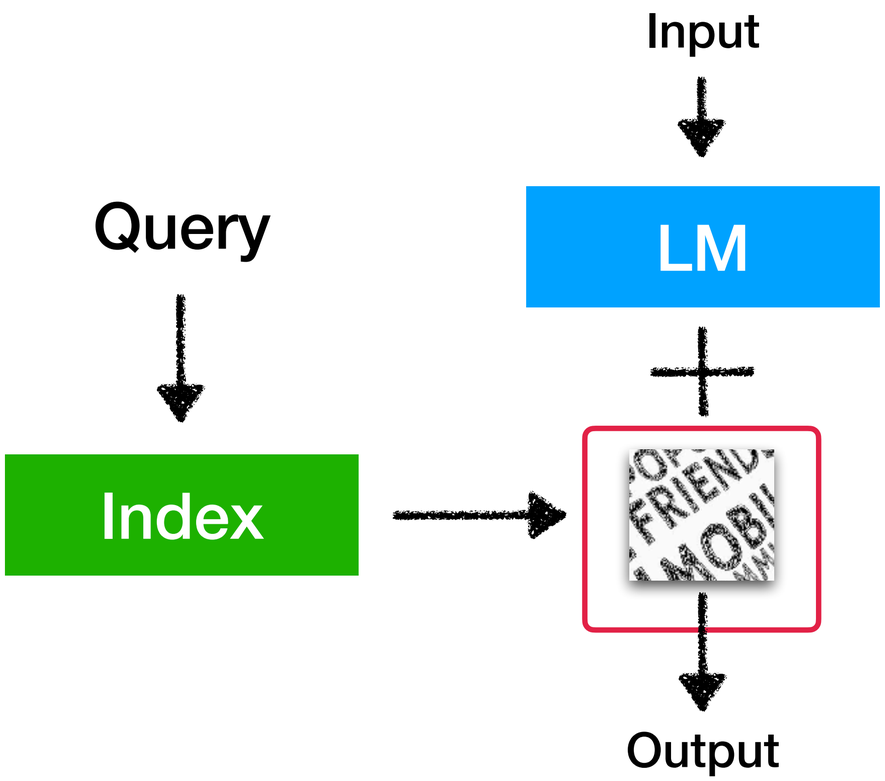
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



How retrieval-augmented LMs solve the issues?

Hallucinations

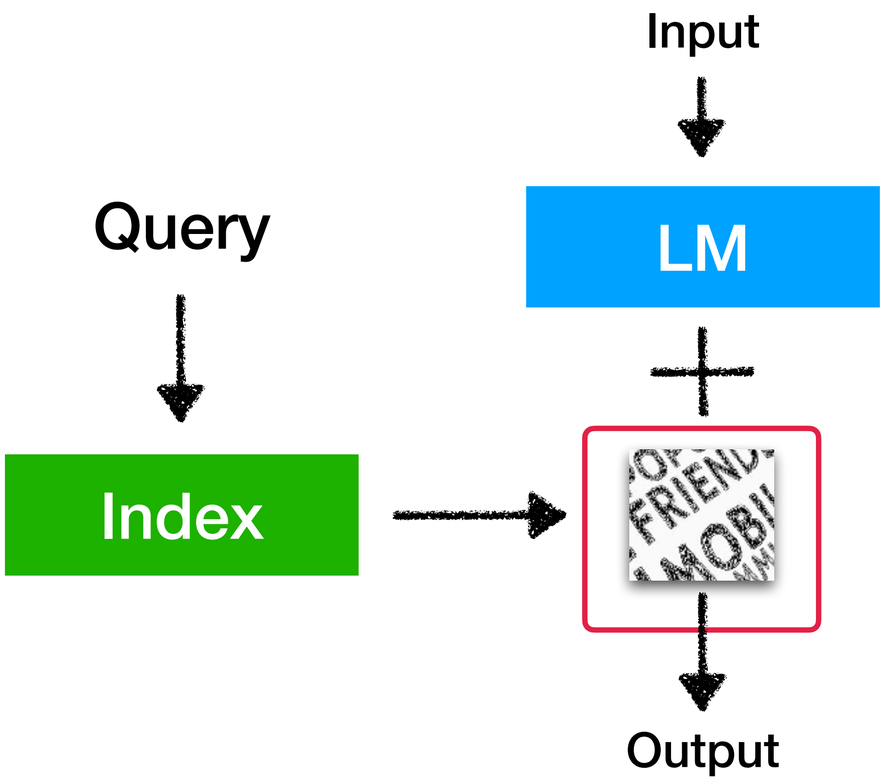
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Retrieved text can be used as attributions



How retrieval-augmented LMs solve the issues?

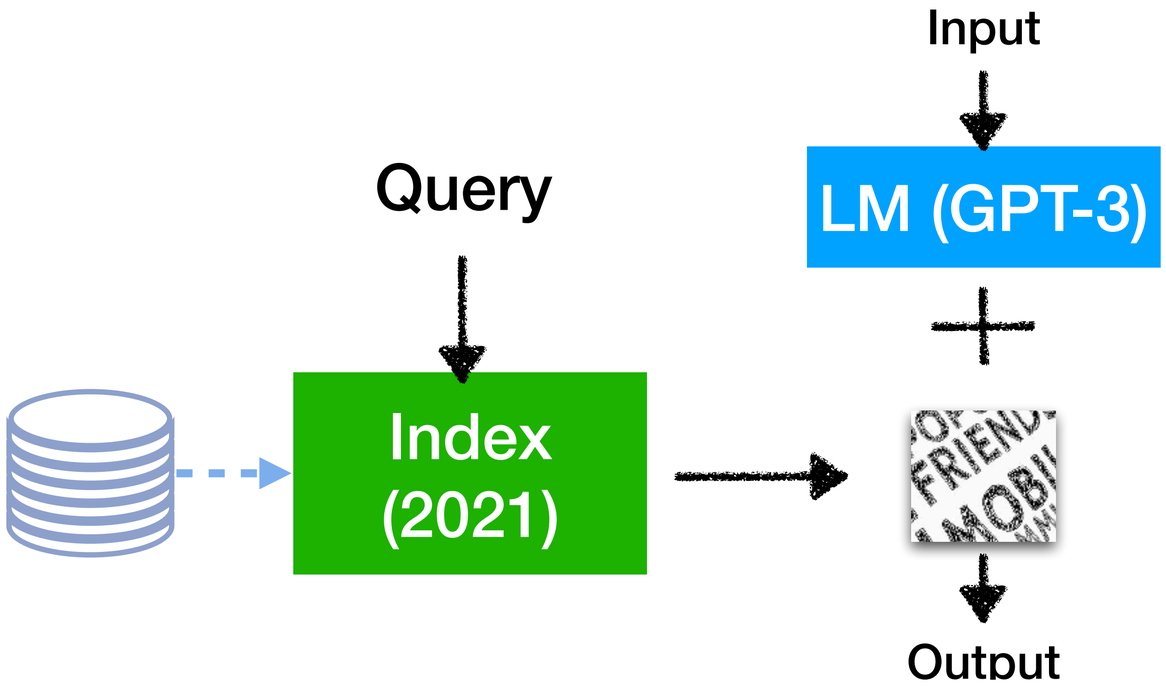
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



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

How retrieval-augmented LMs solve the issues?

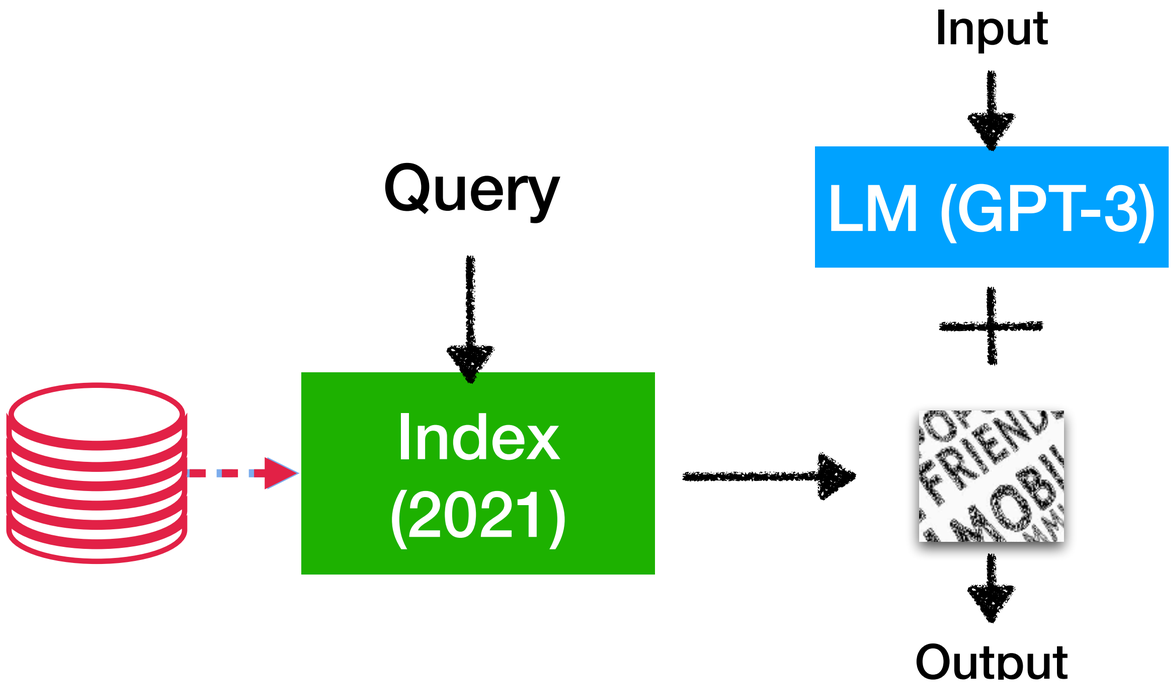
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



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

How retrieval-augmented LMs solve the issues?

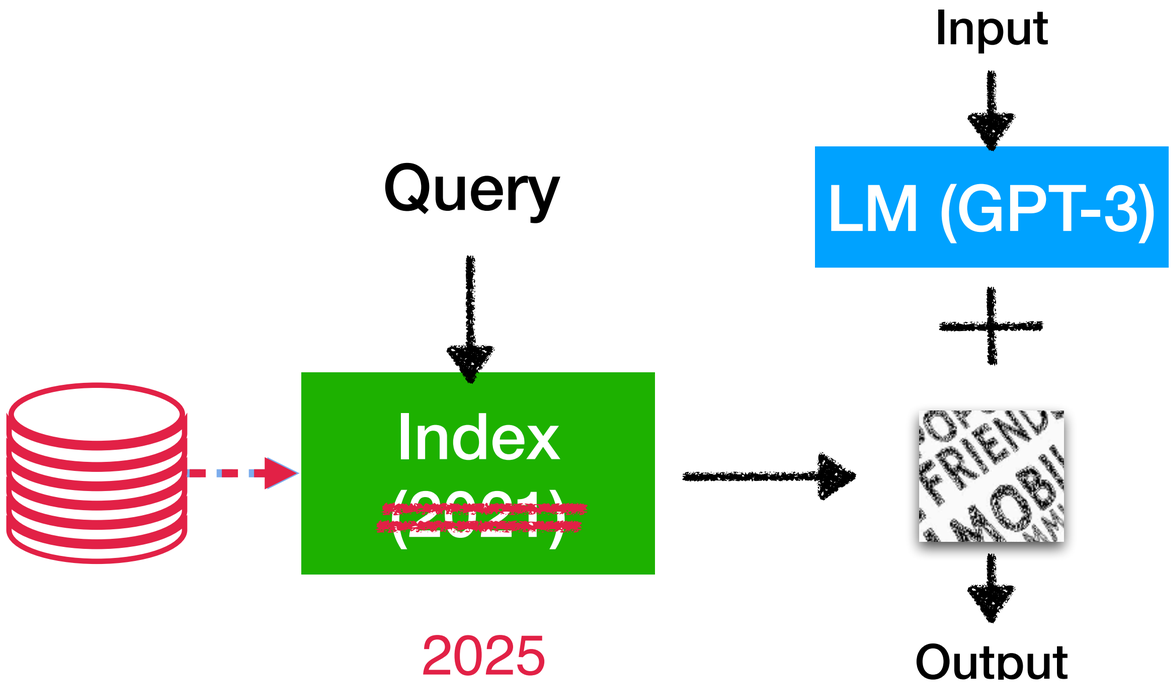
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



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

How retrieval-augmented LMs solve the issues?

Hallucinations

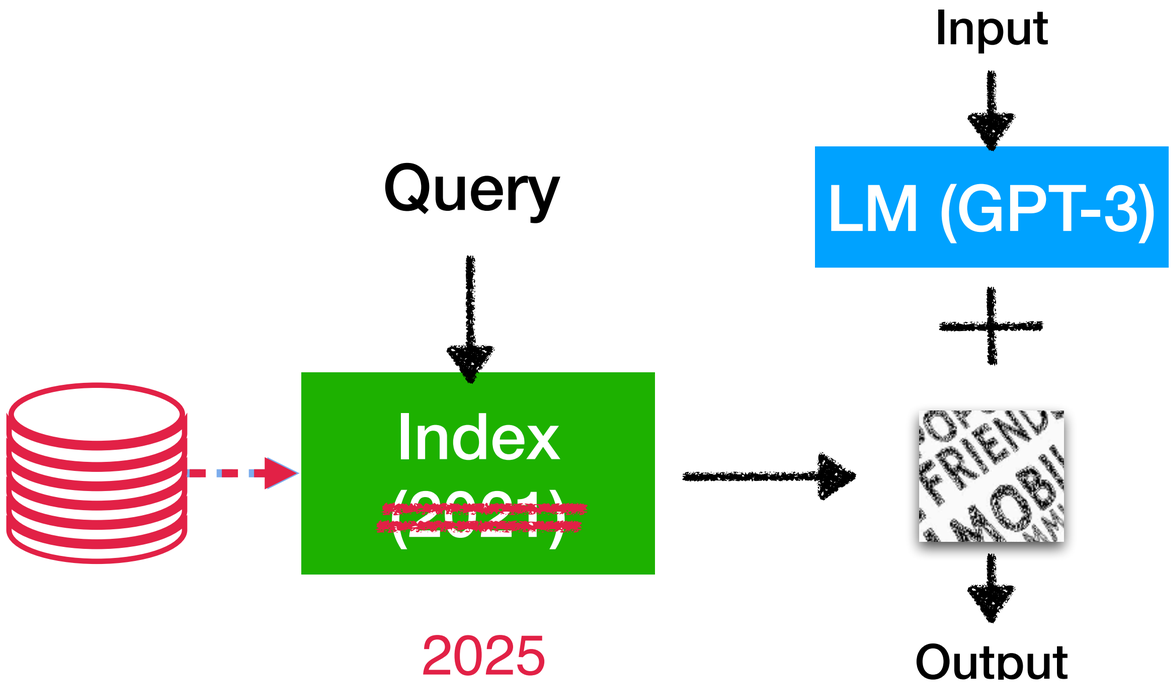
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

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?

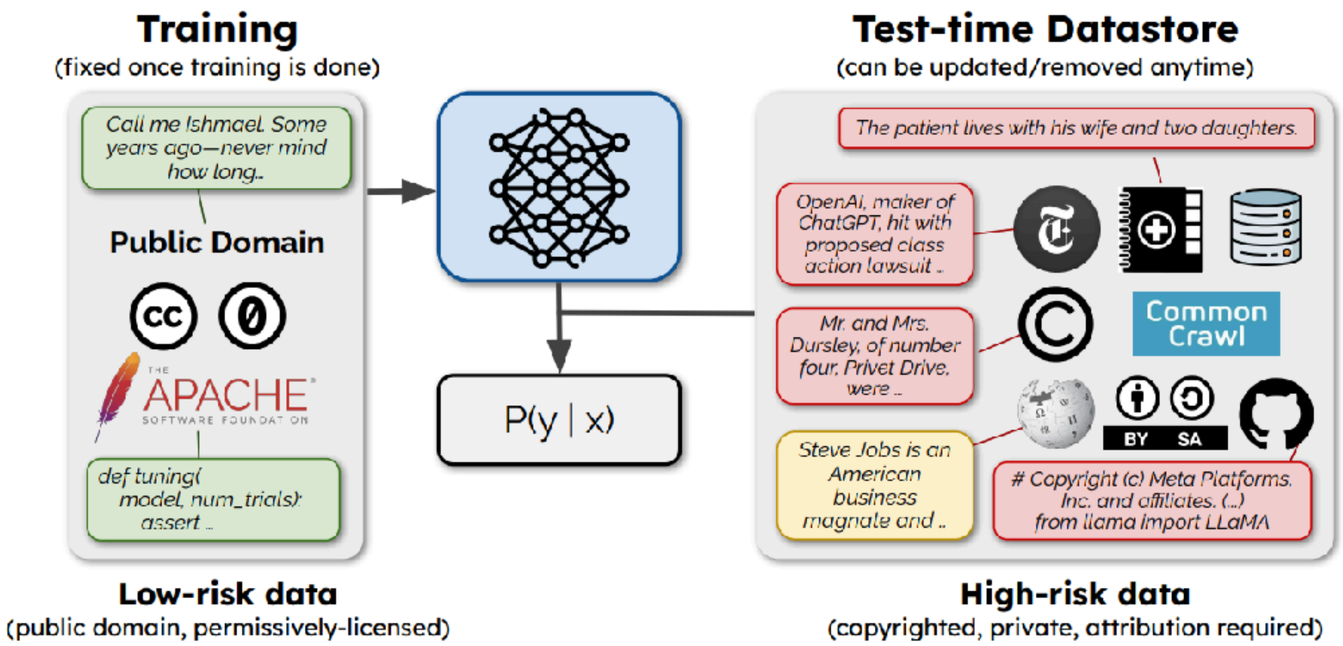
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



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

How retrieval-augmented LMs solve the issues?

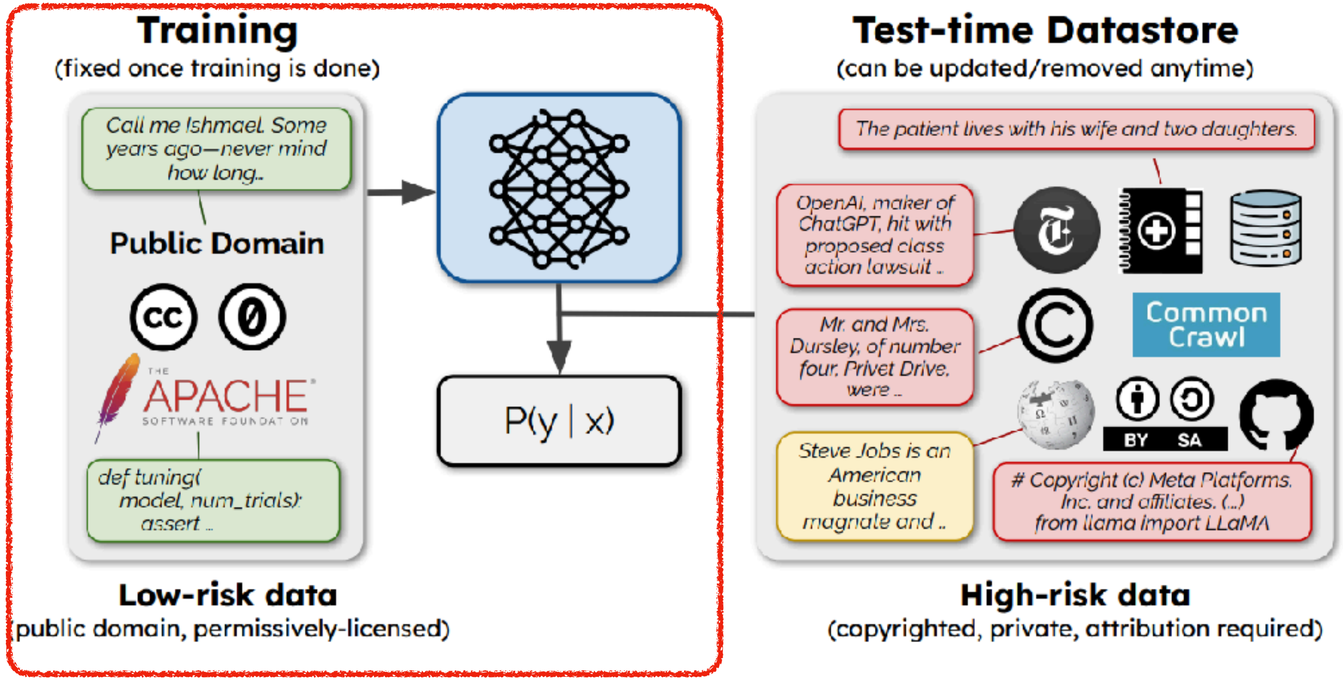
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



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

How retrieval-augmented LMs solve the issues?

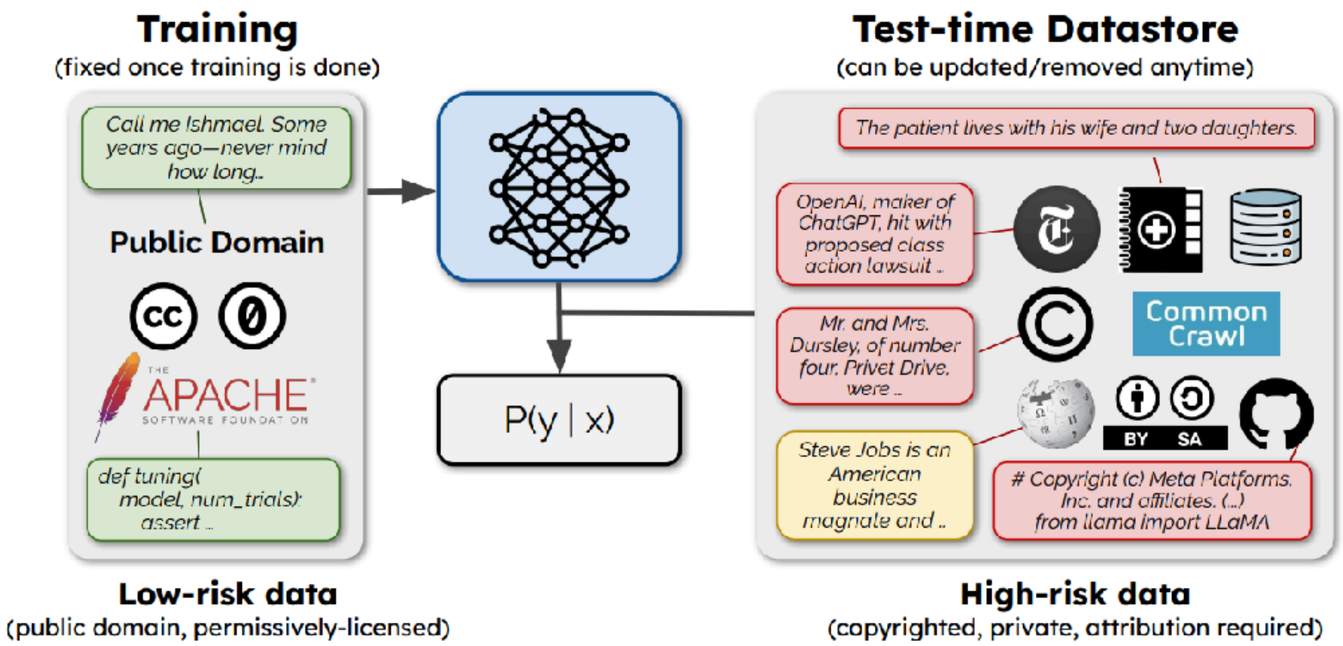
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



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

How retrieval-augmented LMs solve the issues?

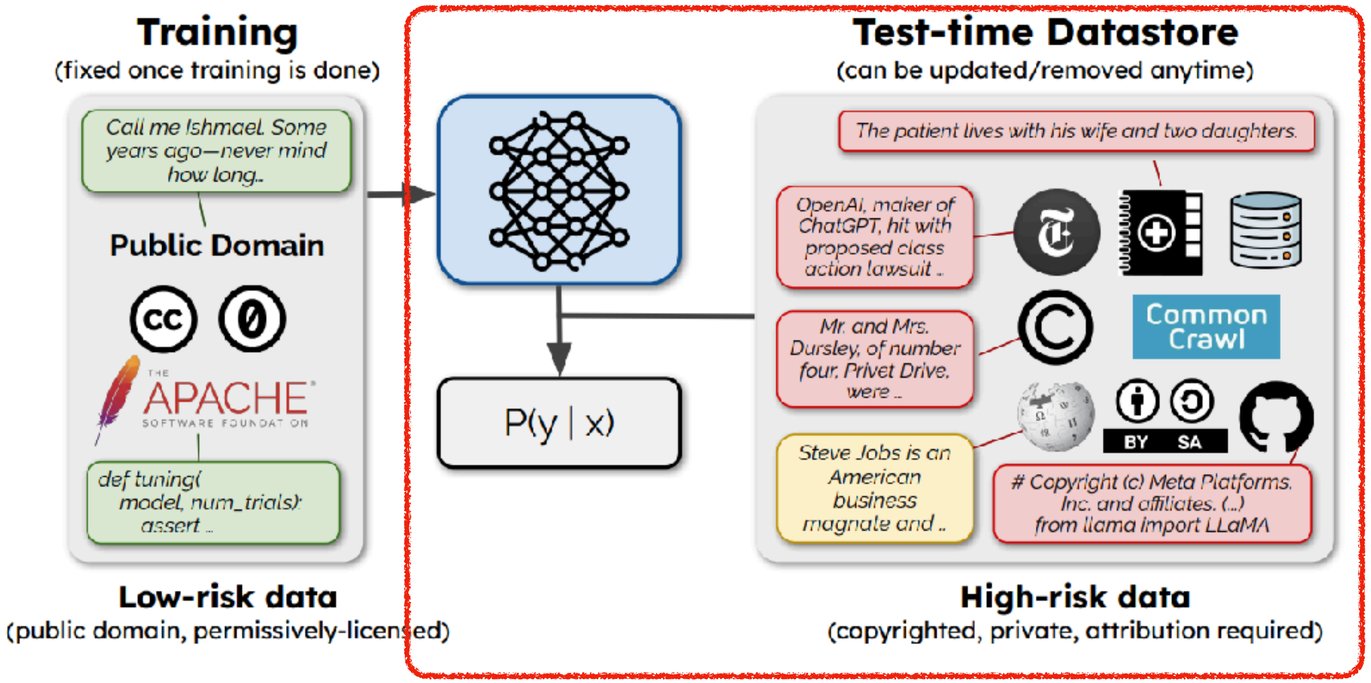
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



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

How retrieval-augmented LMs solve the issues?

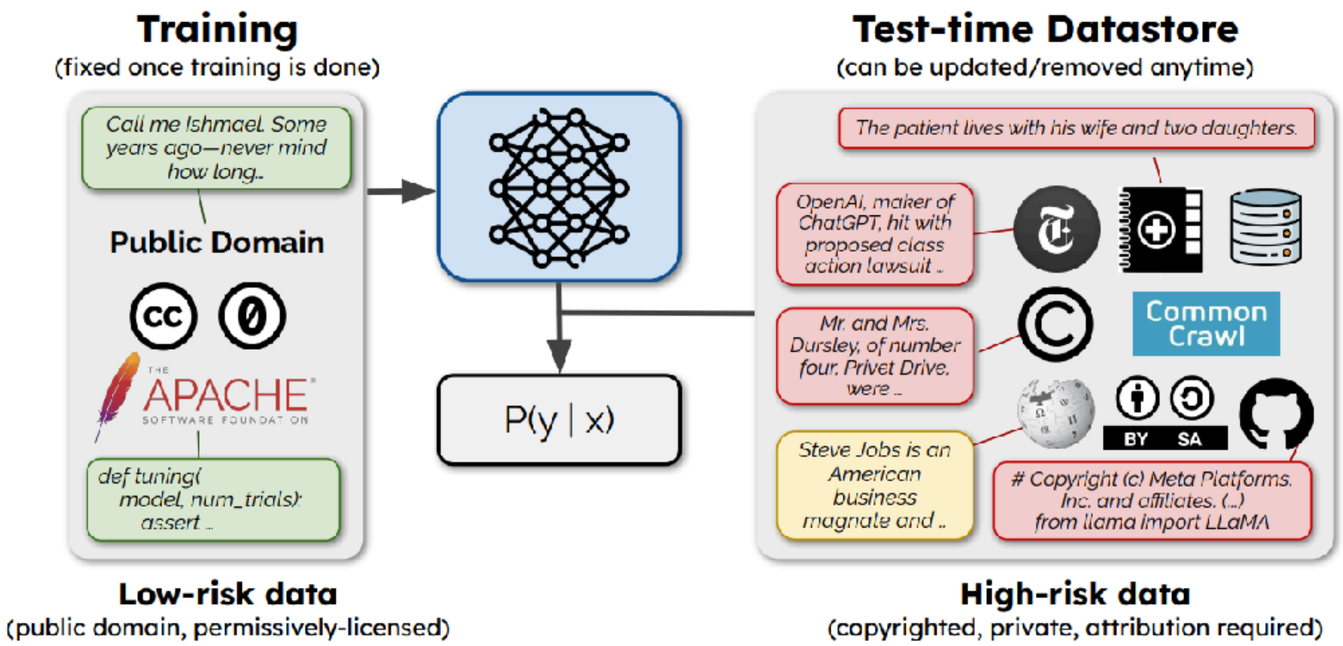
Hallucinations

Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size



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

How retrieval-augmented LMs solve the issues?

Hallucinations

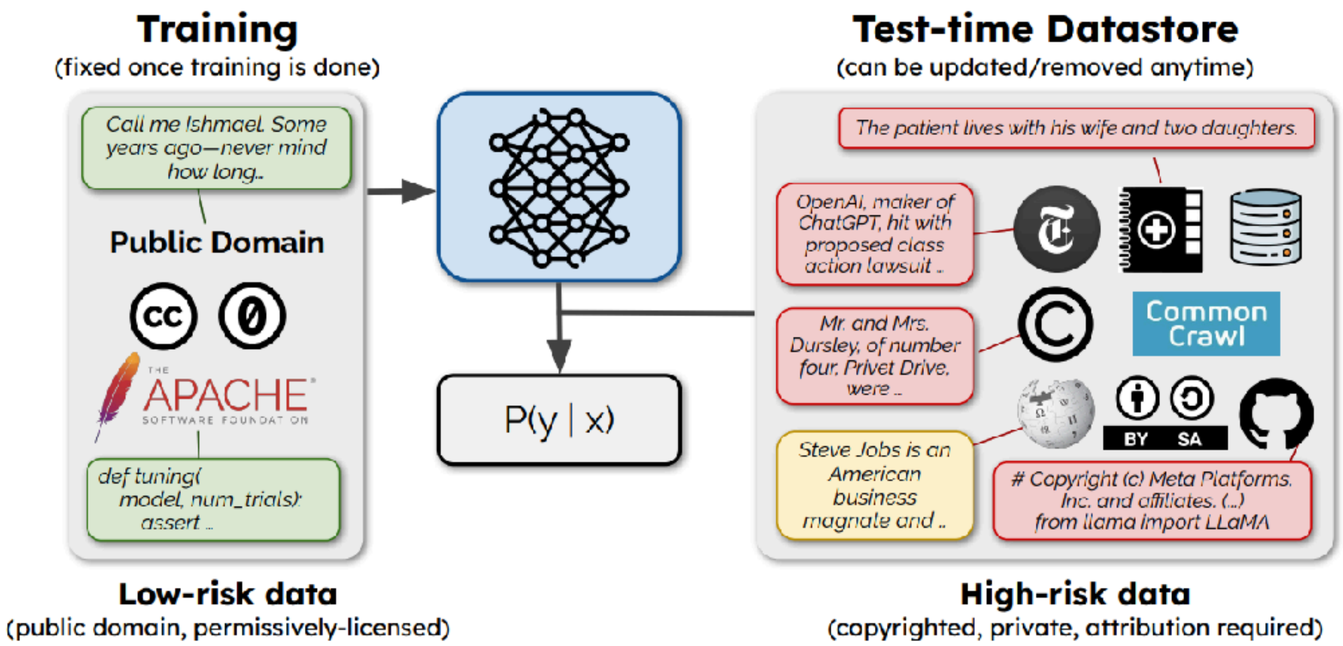
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

Segregating copyright-sensitive data from pre-training data



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

How retrieval-augmented LMs solve the issues?

Hallucinations

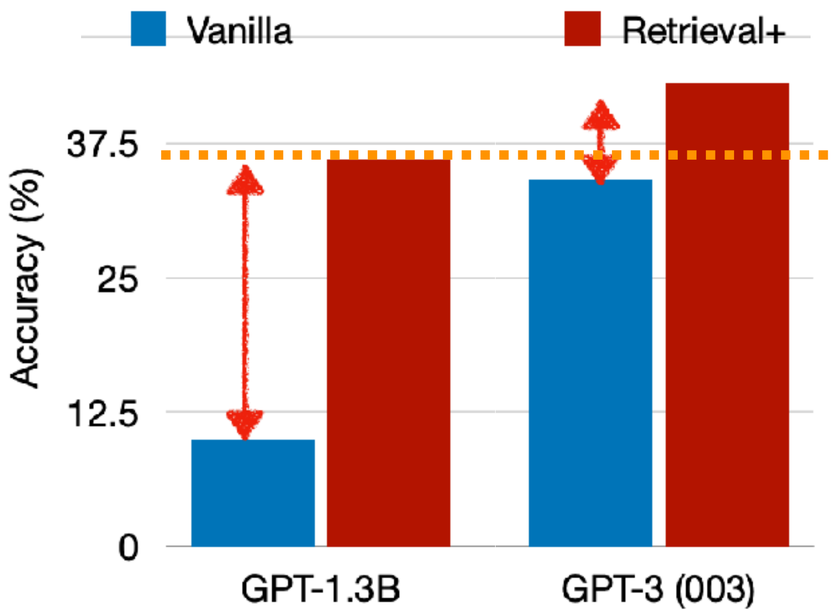
Lack of attributions

Costs of adaptations

Copyright / privacy

Large parameter size

QA



Mallen et al., When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories 2023.

How retrieval-augmented LMs solve the issues?

Hallucinations

Lack of attributions

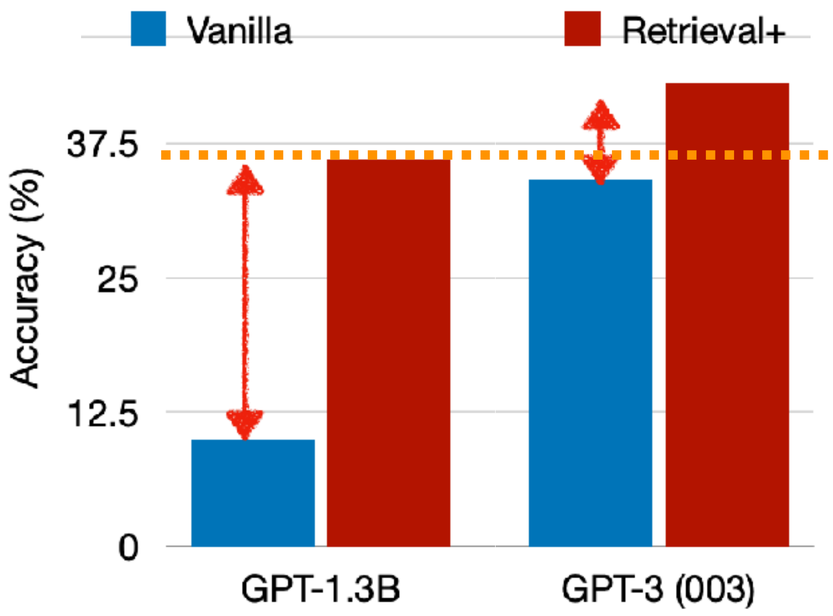
Costs of adaptations

Copyright / privacy

Large parameter size

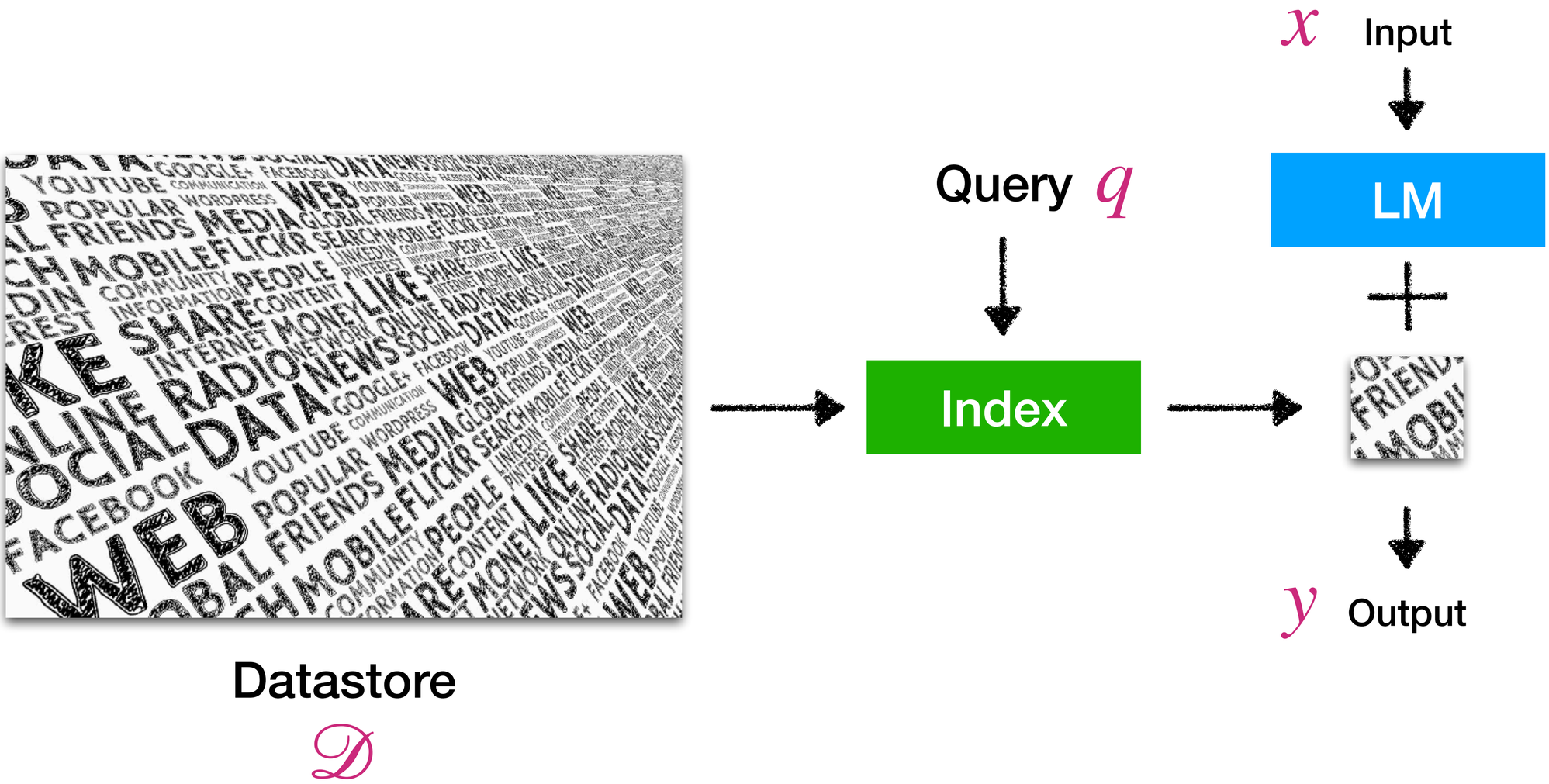
Smaller LMs with retrieval outperform much larger LMs e.g., GPT-3

QA



Mallen et al., When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories 2023.

Notation



Inference: Index

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

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

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

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

N: # of total docs

df_i: # of docs containing i

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

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

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

N: # of total docs

df_i: # of docs containing i

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

Example

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

Maps the text into an h-dimensional vector

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

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

N: # of total docs

df_i: # of docs containing i

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

Example

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

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

sim: a similarity score between two pieces of text

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

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

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

Index: given q, return $\underbrace{\mathrm{argTop}\text{-}k_{d \in \mathcal{D}} \mathrm{sim}(q, d)}_{k \text{ elements from a datastore}}$ through fast nearest neighbor search

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

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

Can be a totally separate research area on
how to do this fast & accurately

Index: given q, return $\underbrace{\mathrm{argTop}\text{-}k_{d \in \mathcal{D}} \mathrm{sim}(q, d)}_{k \text{ elements from a datastore}}$ through fast nearest neighbor search

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

Categorization of retrieval-augmented LMs

Categorization of retrieval-augmented LMs

What to retrieve?

Query

Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?

Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?

Tokens?

Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

Categorization of retrieval-augmented LMs

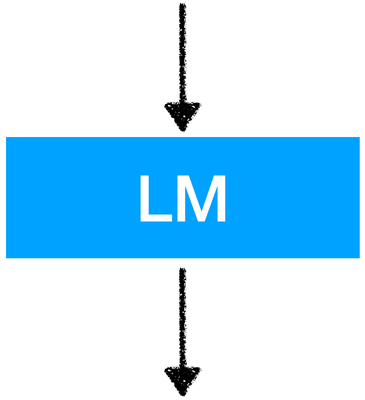
What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?

Input



Output

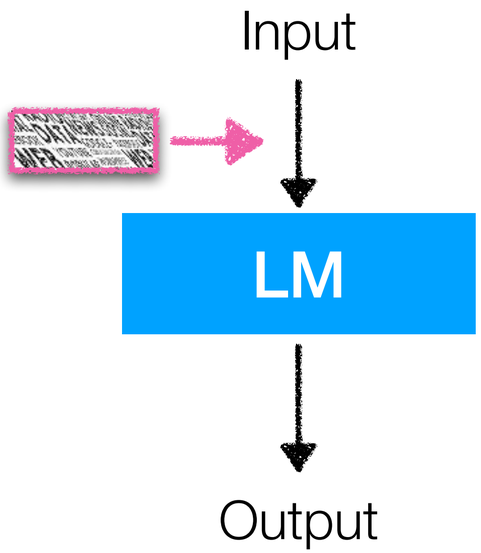
Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



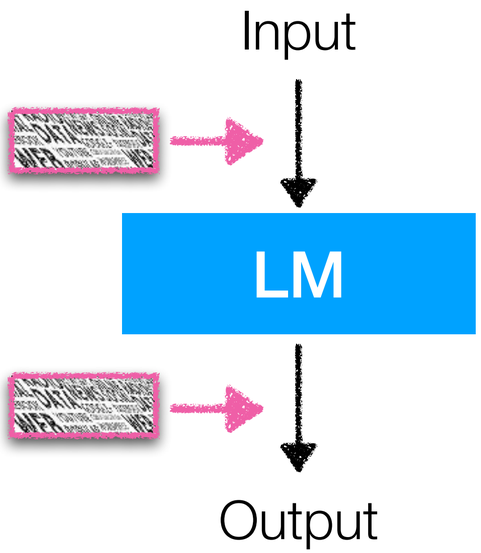
Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



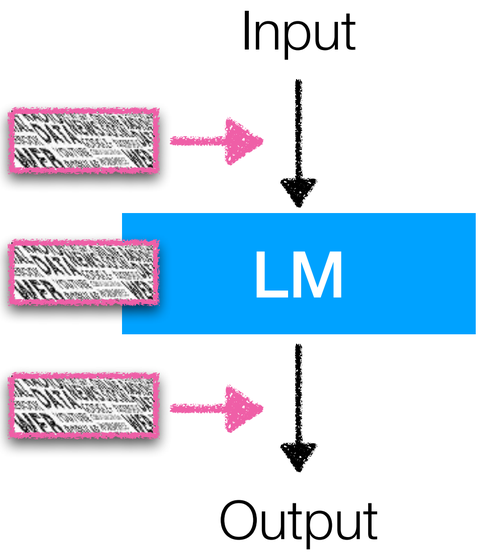
Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



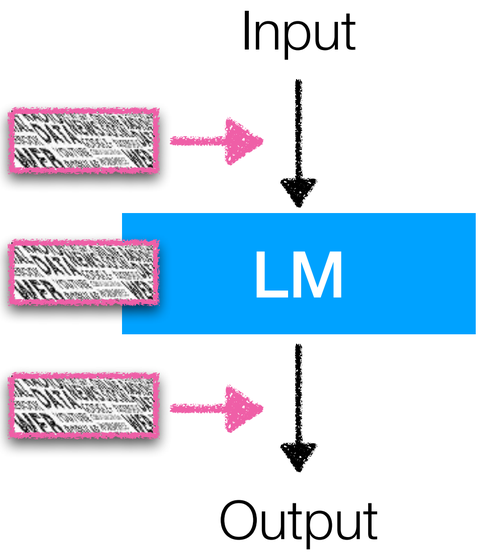
Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



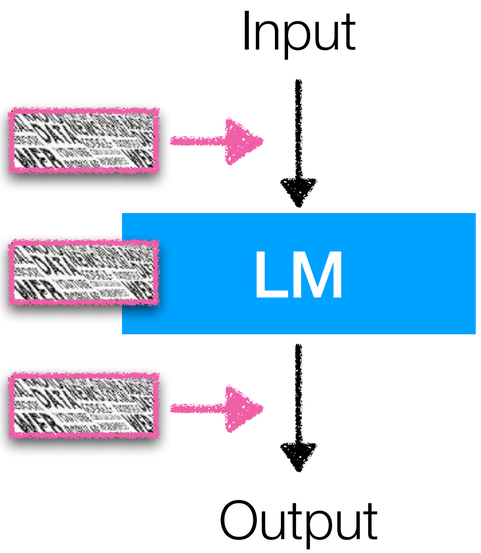
When to retrieve?

Categorization of retrieval-augmented LMs

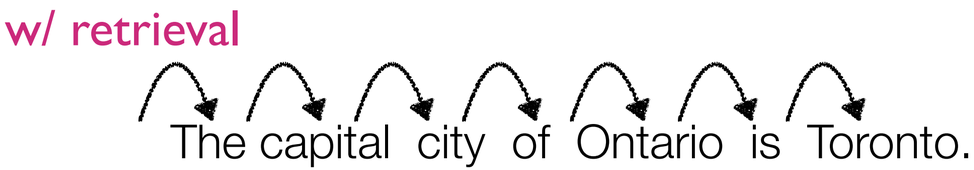
What to retrieve?

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



When to retrieve?



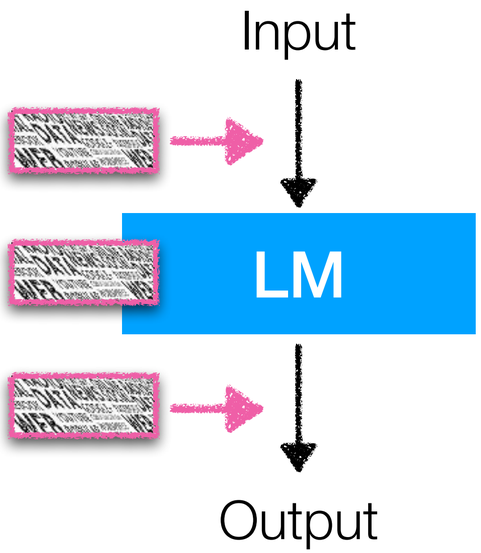
Categorization of retrieval-augmented LMs

What to retrieve?

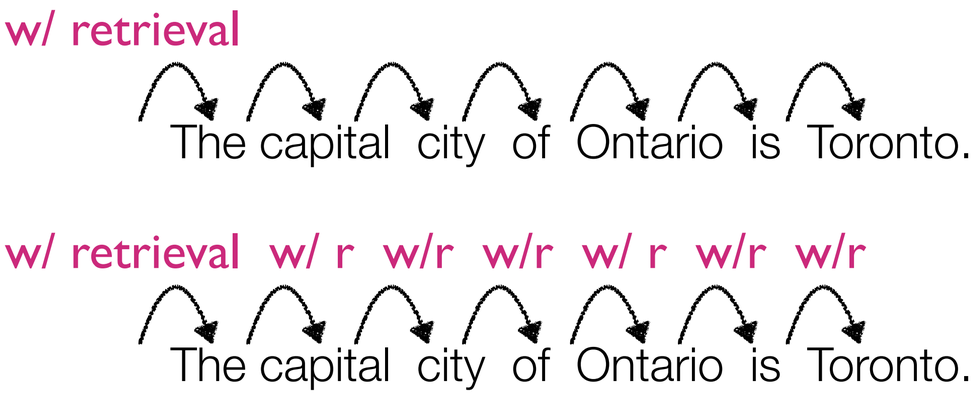
Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



When to retrieve?



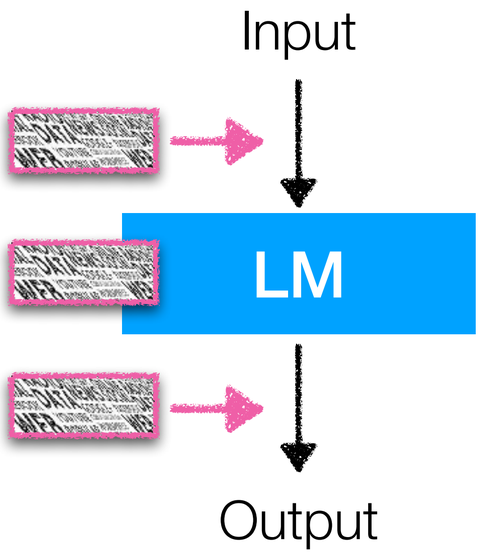
Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



When to retrieve?

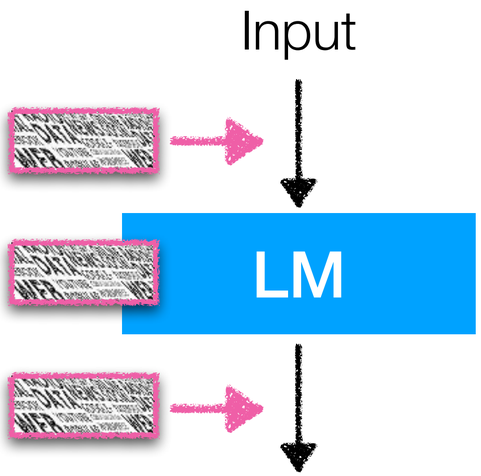
Categorization of retrieval-augmented LMs

What to retrieve?

Query

Text chunks (passages)?
Tokens?
Something else?

How to use retrieval?



When to retrieve?

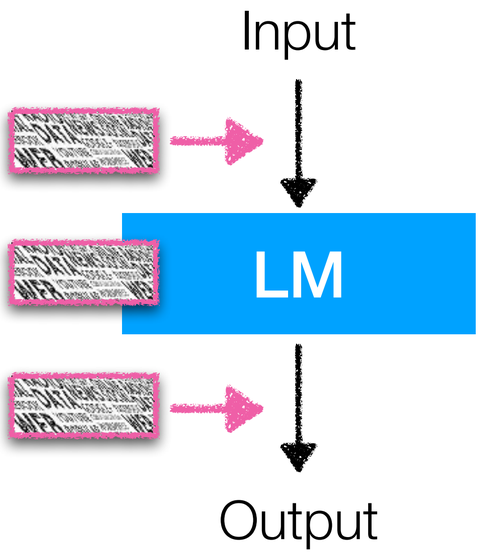
Categorization of retrieval-augmented LMs

What to retrieve?

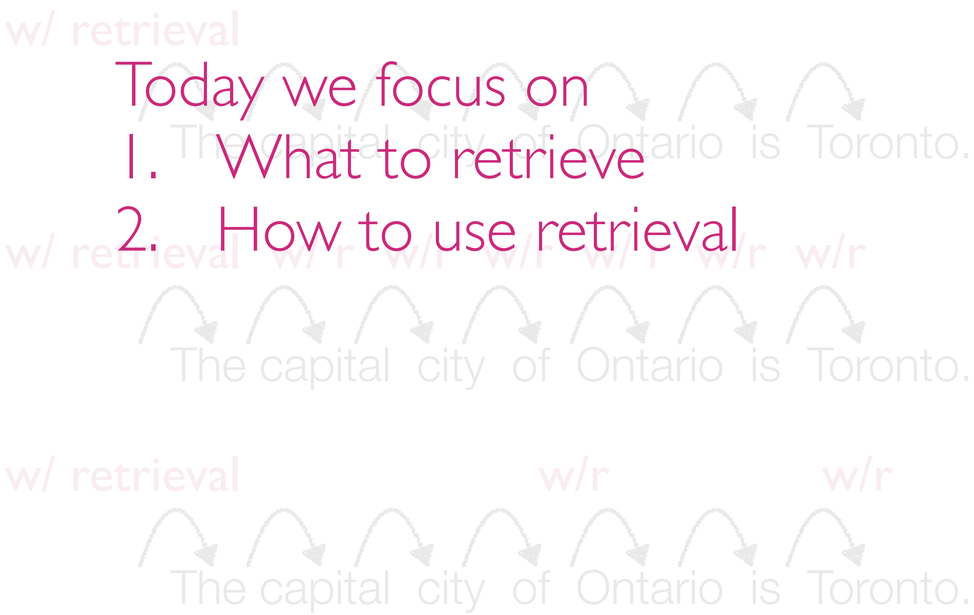
Query

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

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)

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)

REALM (Guu et al 2020)

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.

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.

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

REALM (Guu et al 2020)

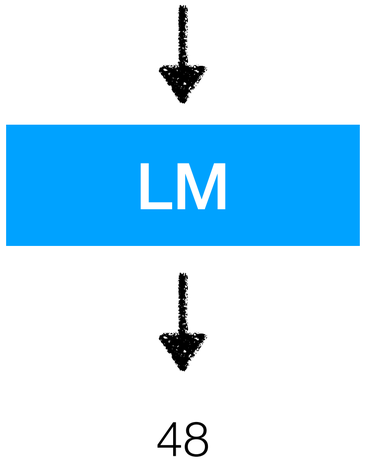
x = World Cup 2022 was the last with 32 teams before the increase to **[MASK]** in 2026.

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.

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.

q (=x)

↓

Retrieval

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

↓

LM

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.

q (=x)

↓

Retrieval

↓

k chunks of text (passages)

FIFA World Cup 2026 will expand to 48 teams.

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

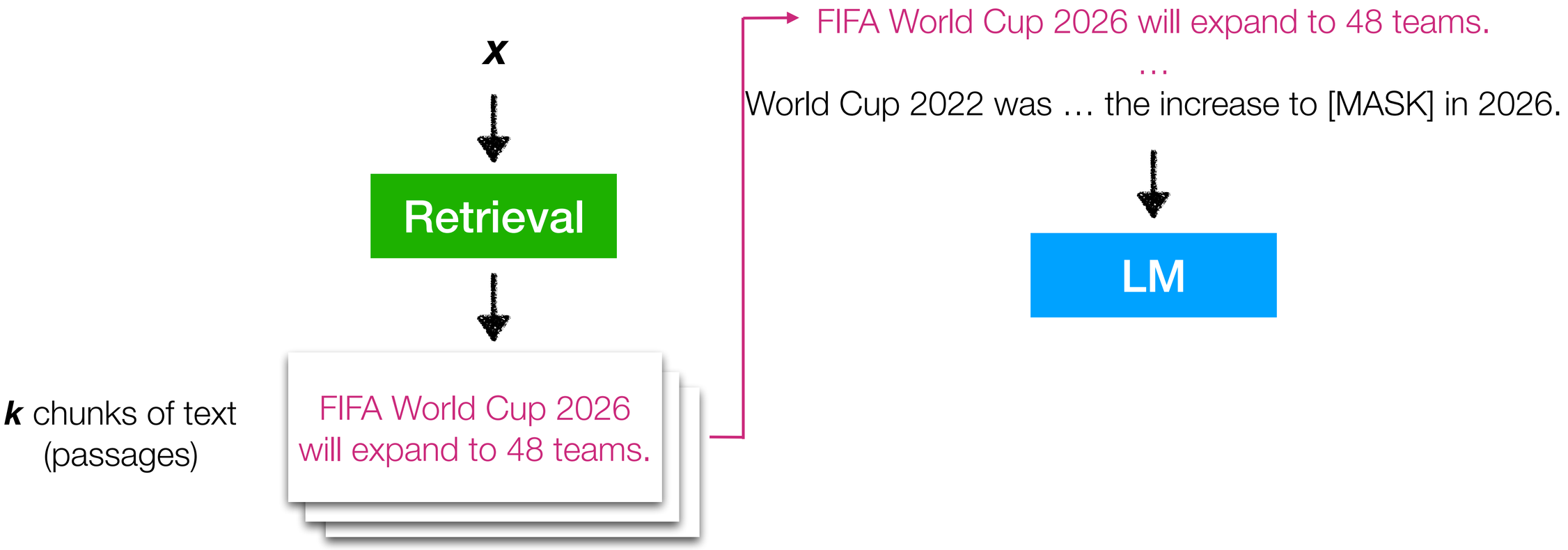
↓

LM

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

REALM (Guu et al 2020)

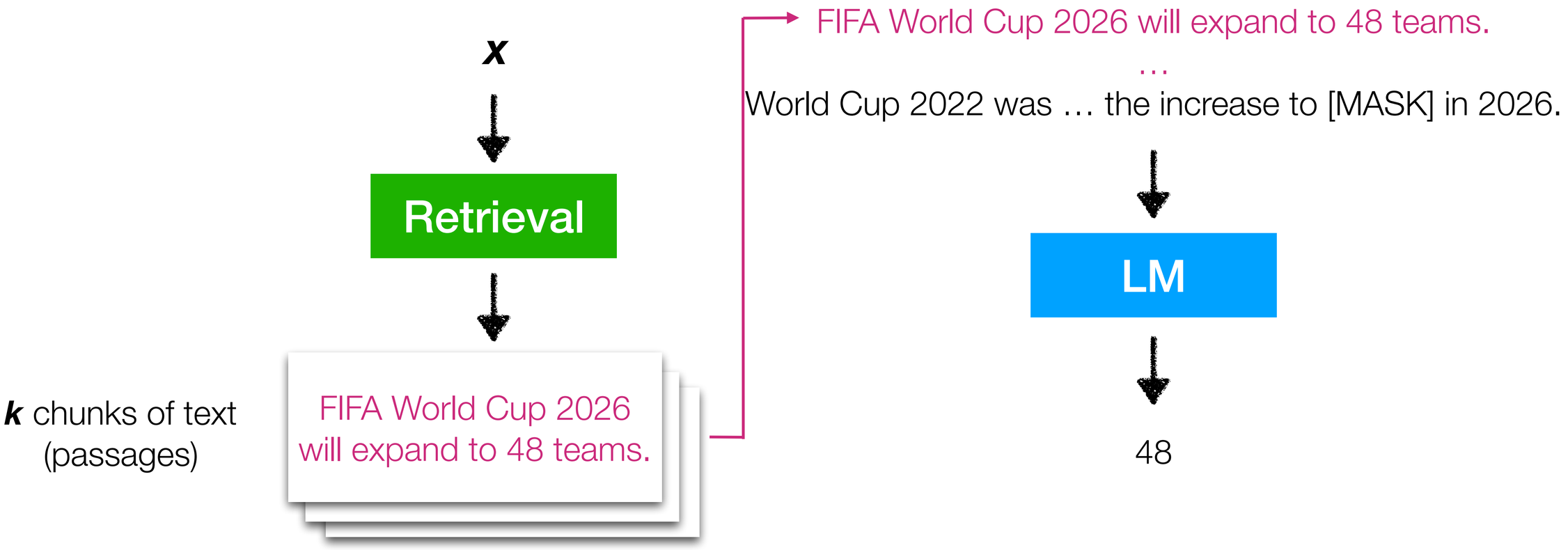
x = World Cup 2022 was the last before the increase to [MASK] in the 2026 tournament.



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

REALM (Guu et al 2020)

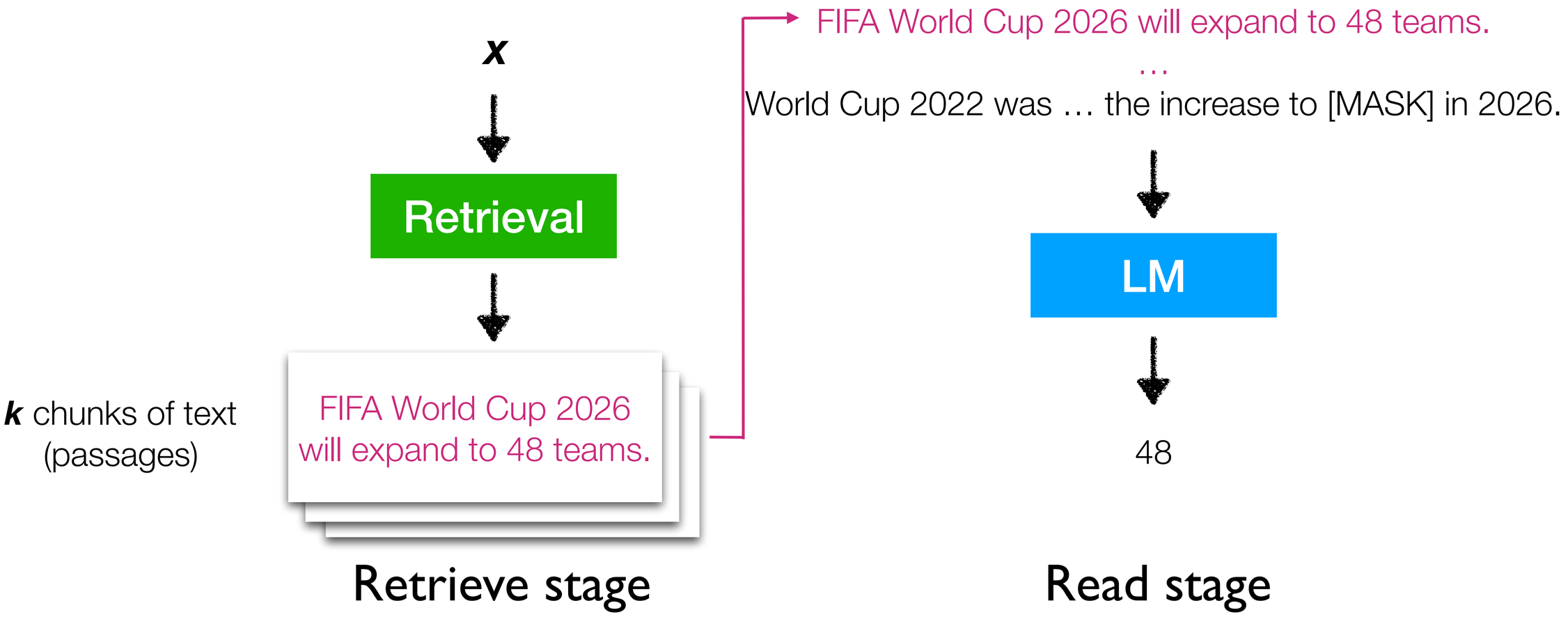
x = World Cup 2022 was the last before the increase to [MASK] in the 2026 tournament.



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

REALM (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

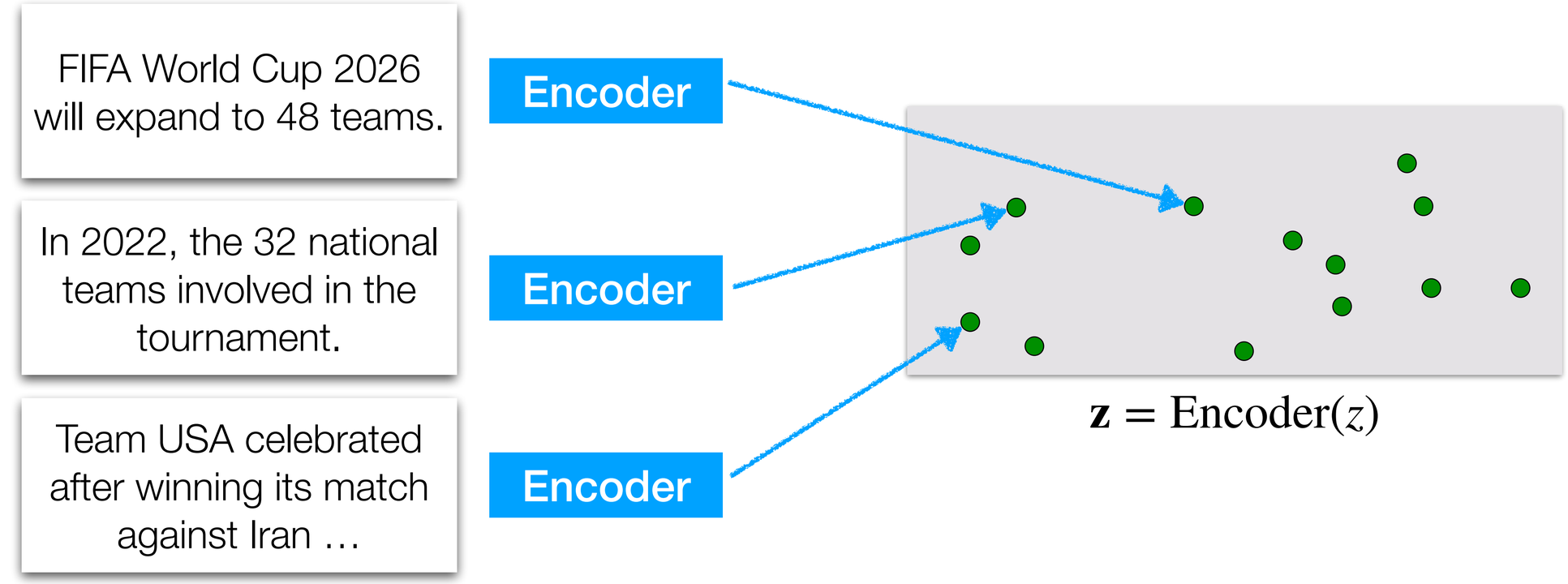
FIFA World Cup 2026 will expand to 48 teams.

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)

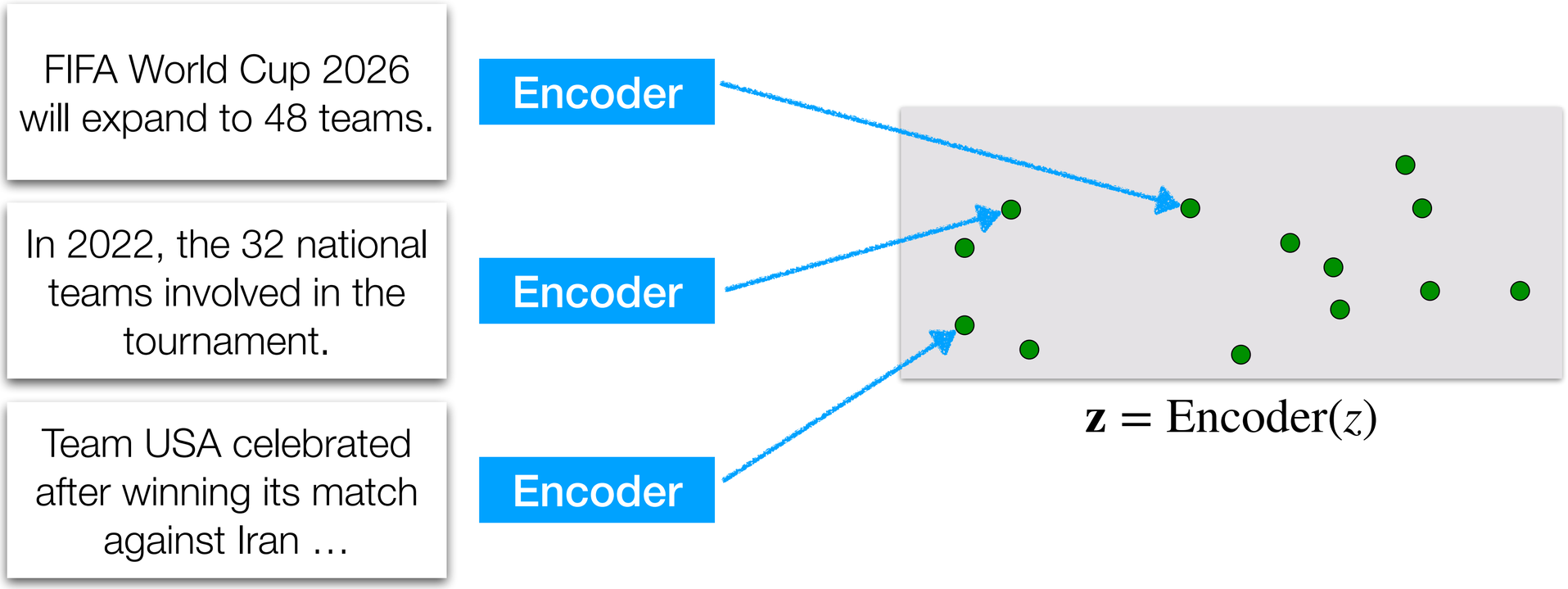
REALM: (1) Retrieve stage



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

REALM: (1) Retrieve stage

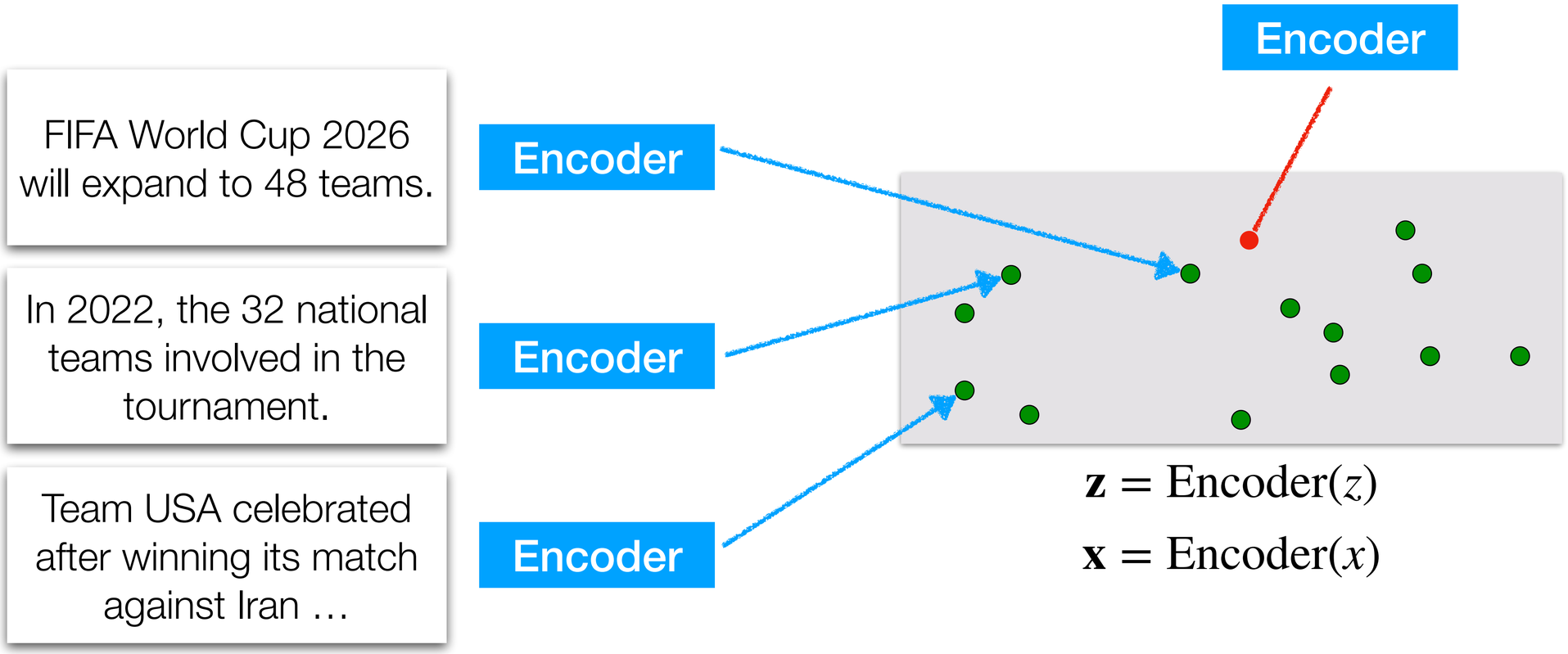
x = World Cup 2022 was … the increase to [MASK] in 2026.



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

REALM: (1) Retrieve stage

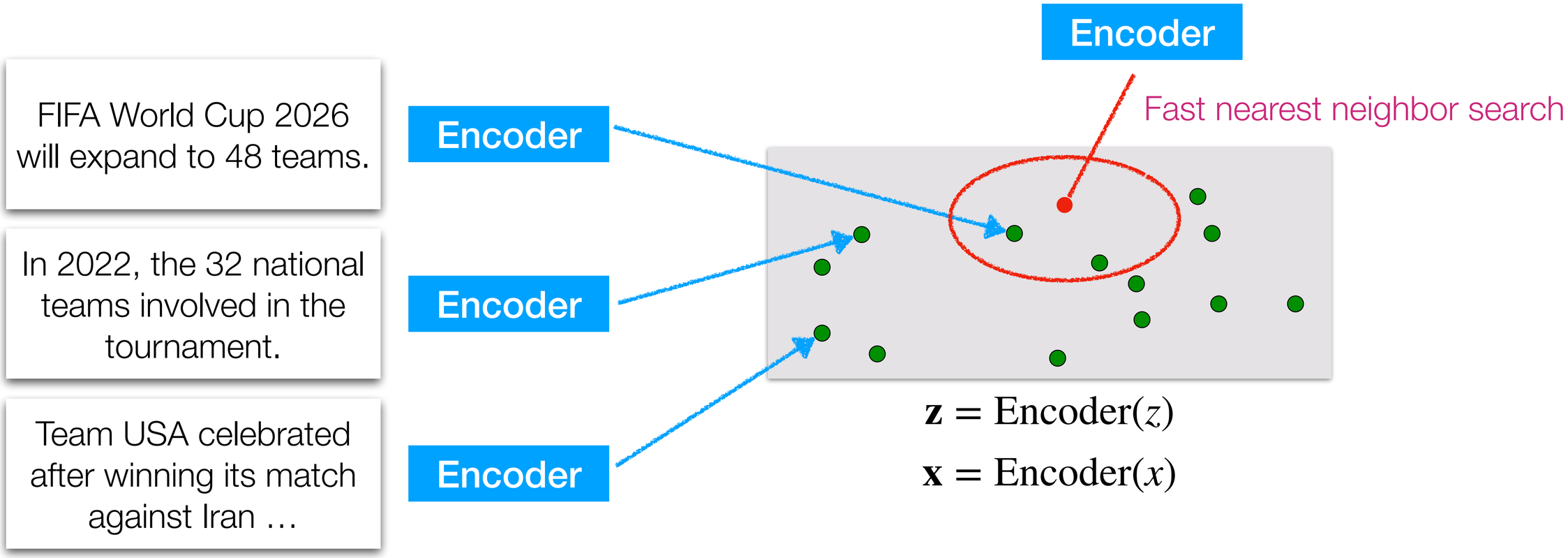
x = World Cup 2022 was … the increase to [MASK] in 2026.



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

REALM: (1) Retrieve stage

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



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

REALM: (1) Retrieve stage

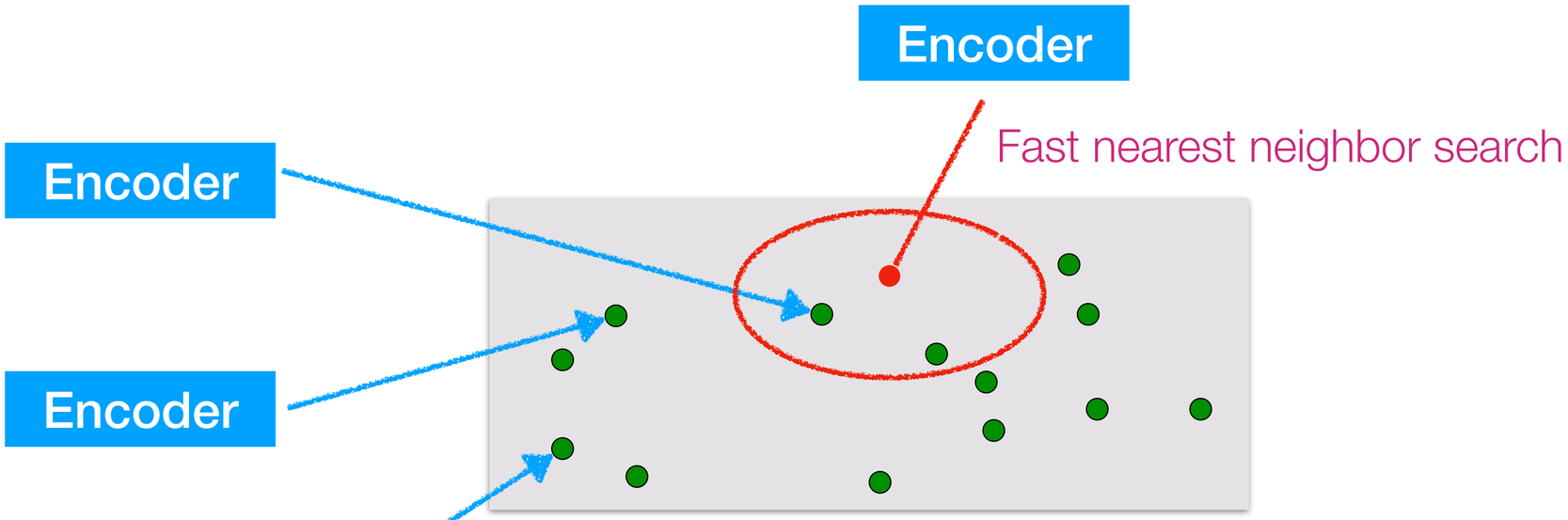
x = World Cup 2022 was … the increase to [MASK] in 2026.

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)



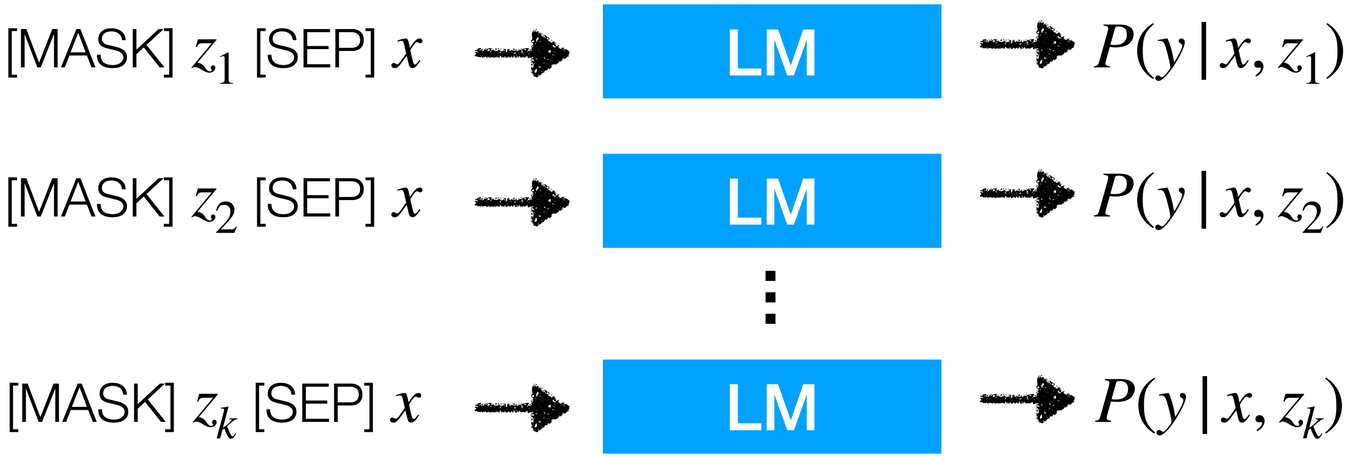
$$\mathbf{z} = \text{Encoder}(z)$$

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

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

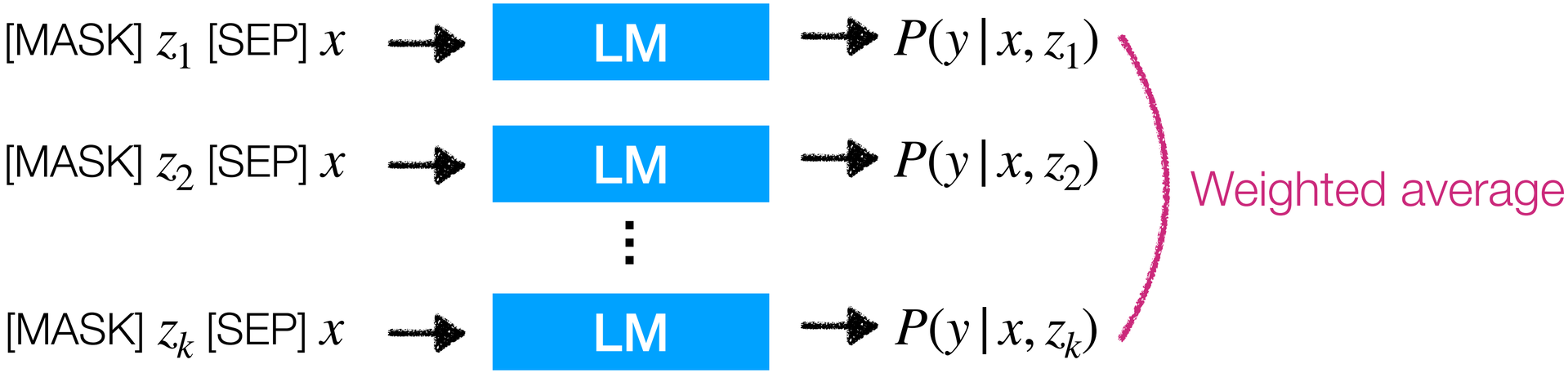
k retrieved chunks

REALM: (2) Read stage



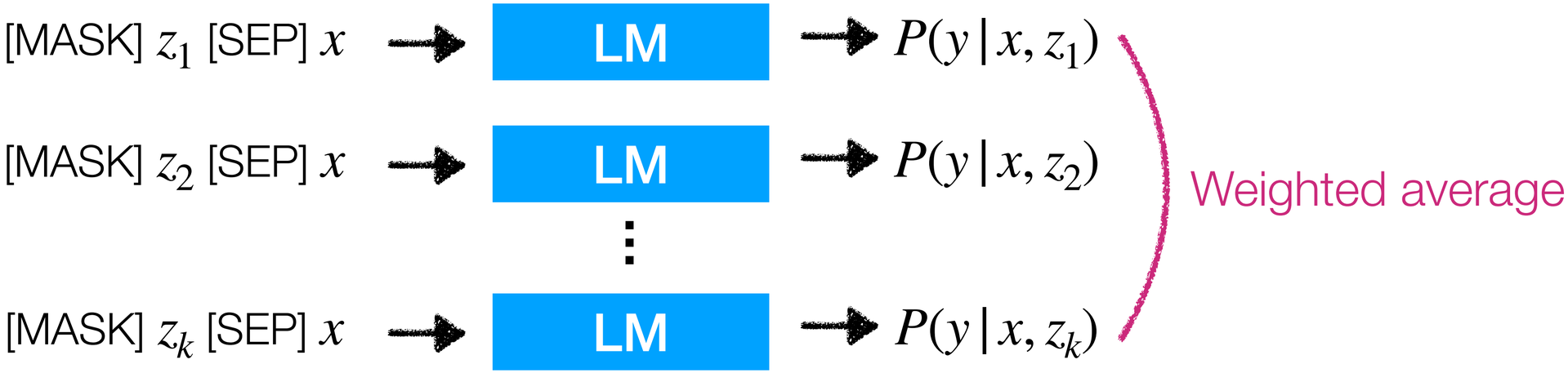
Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

REALM: (2) Read stage



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

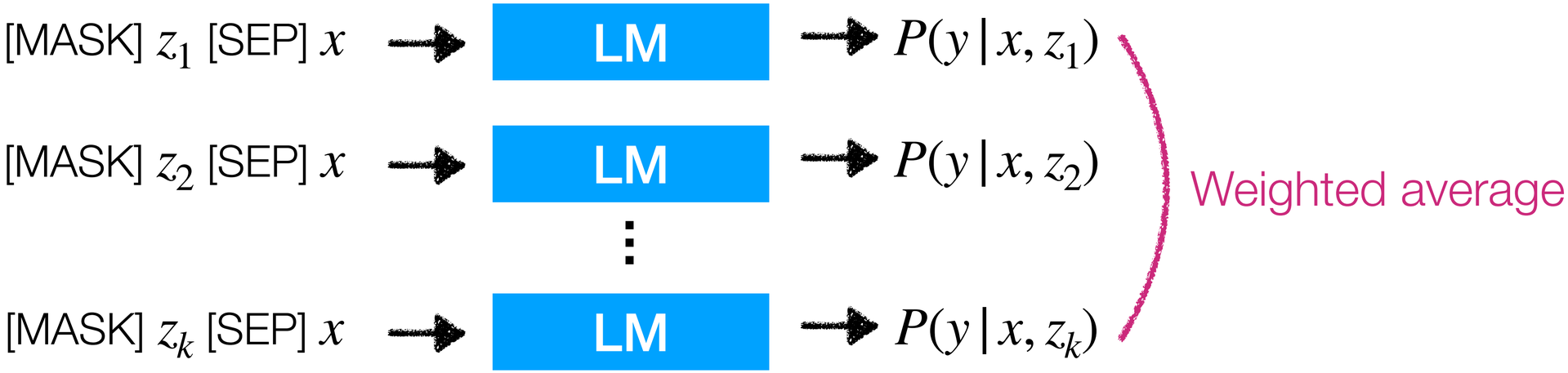
REALM: (2) Read stage



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

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

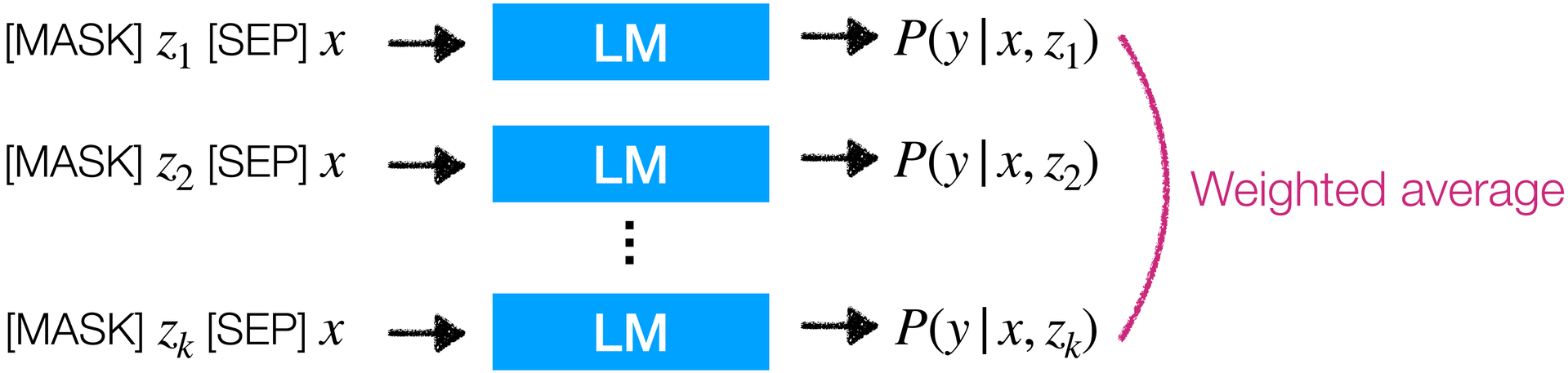
REALM: (2) Read stage



$$\sum_{z \in \mathcal{D}} \underbrace{P(z \mid x)}_{\text{from the retrieve stage}} P(y \mid x, z)$$

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

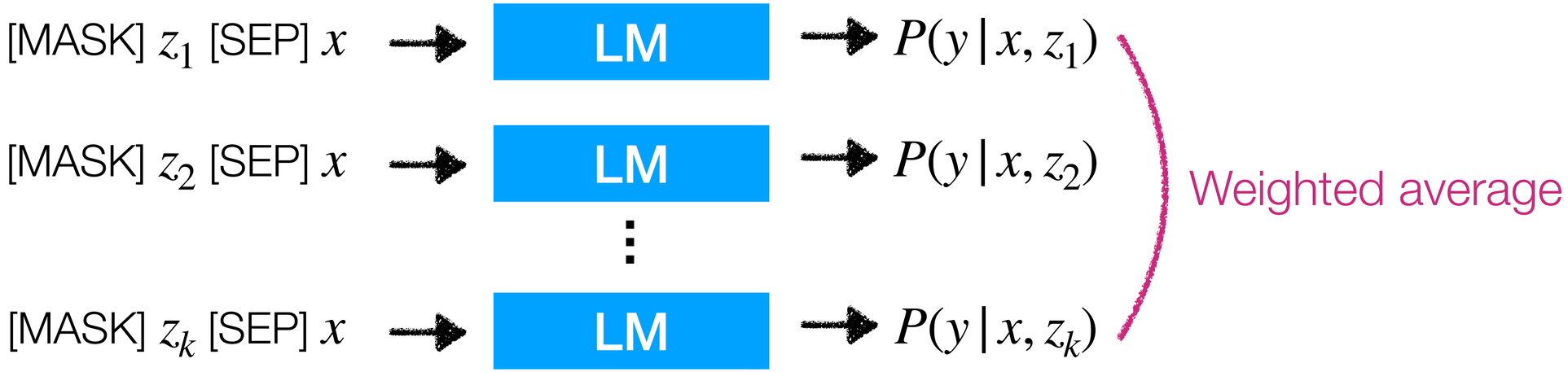
REALM: (2) Read stage



$$\sum_{z \in \mathcal{D}} \underbrace{P(z \mid x)}_{\text{from the retrieve stage}} \underbrace{P(y \mid x, z)}_{\text{from the read stage}}$$

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

REALM: (2) Read stage

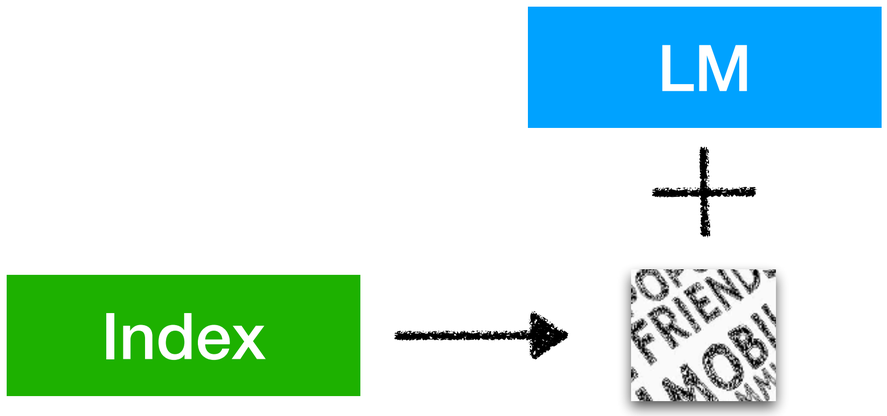


$$\sum_{z \in \mathcal{D}} \underbrace{P(z \mid x)}_{\text{from the retrieve stage}} \underbrace{P(y \mid x, z)}_{\text{from the read stage}}$$

Need to approximate
→ Consider top *k* chunks only

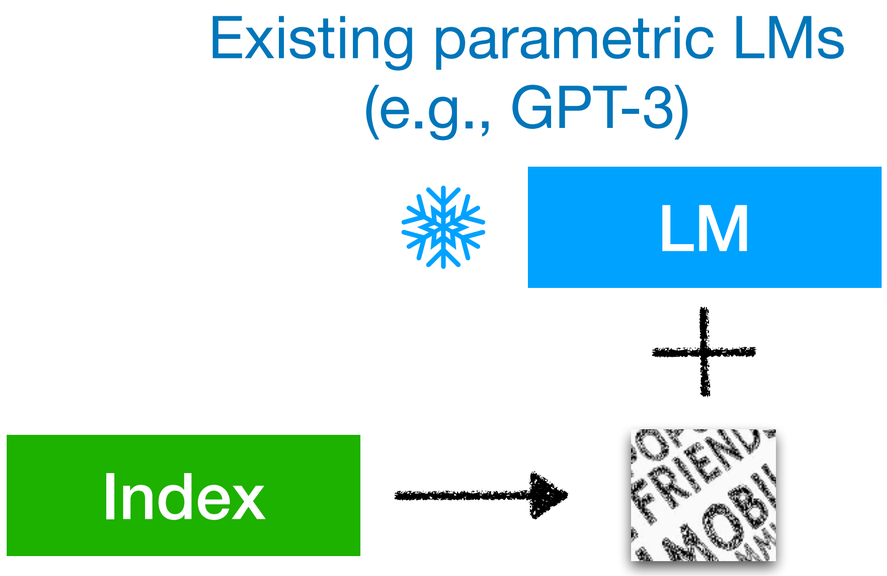
Guu et al. REALM: Retrieval-Augmented Language Model Pre-Training. ICML 2020.

Recent trend: RAG with LLMs



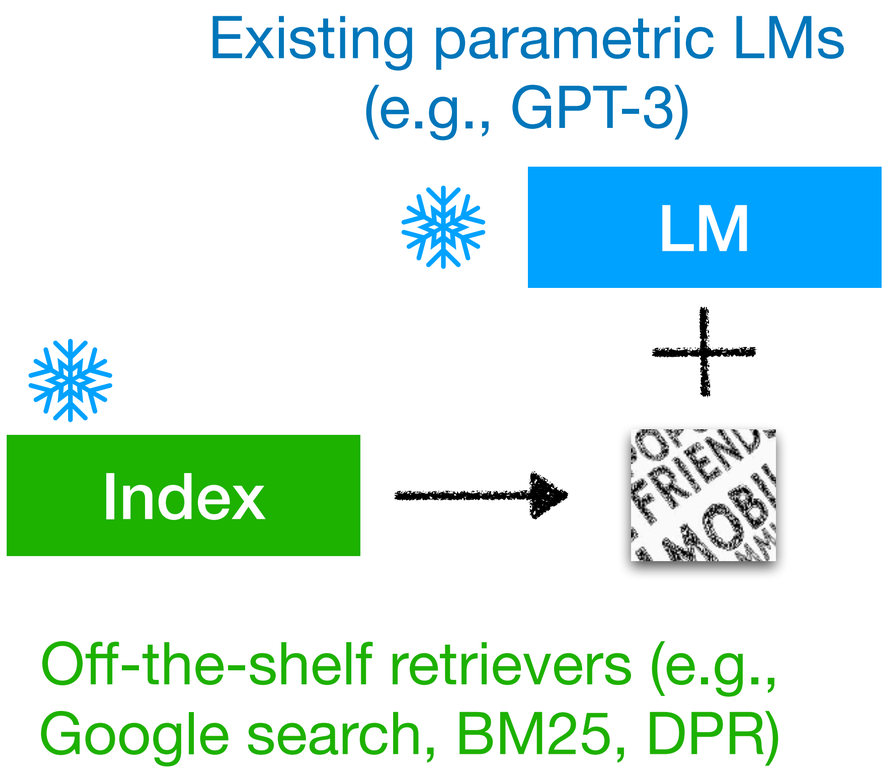
Shi et al. REPLUG: Retrieval-Augmented Black-Box Language Models. Arxiv 2023.
Ram et al. In-Context Retrieval-Augmented Language Models. TACL 2023.

Recent trend: RAG with LLMs



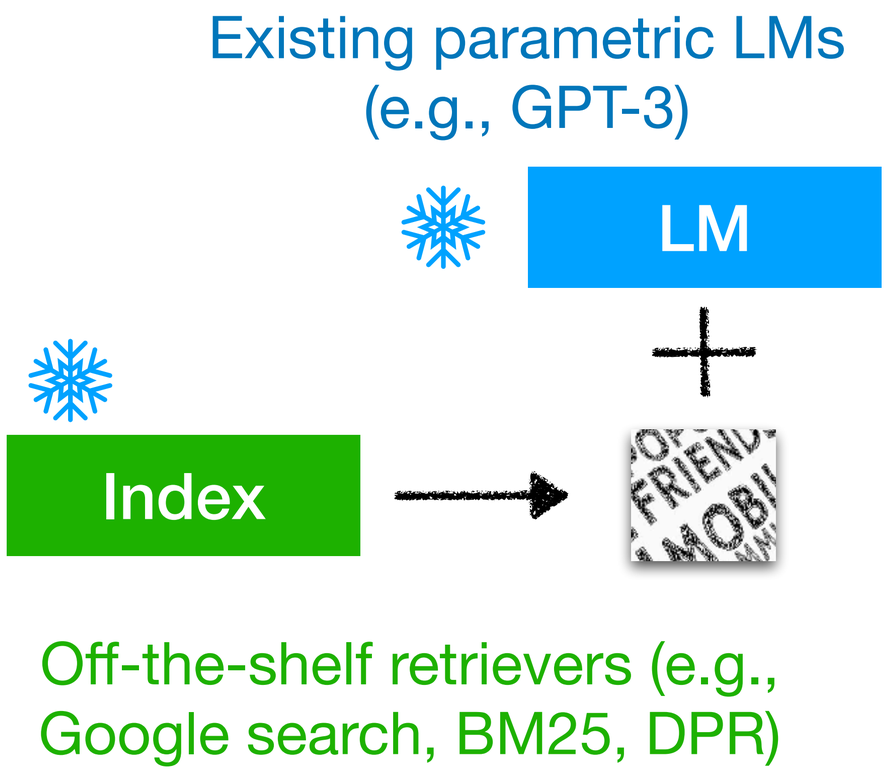
Shi et al. REPLUG: Retrieval-Augmented Black-Box Language Models. Arxiv 2023.
Ram et al. In-Context Retrieval-Augmented Language Models. TACL 2023.

Recent trend: RAG with LLMs



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

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)

RETRO (Borgeaud et al. 2022)

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

RETRO (Borgeaud et al. 2022)

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

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

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)

x = World Cup 2022 was the last with 32 teams, before the increase to

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

RETRO (Borgeaud et al. 2021)

x = World Cup 2022 was the last with 32 teams, before the increase to

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

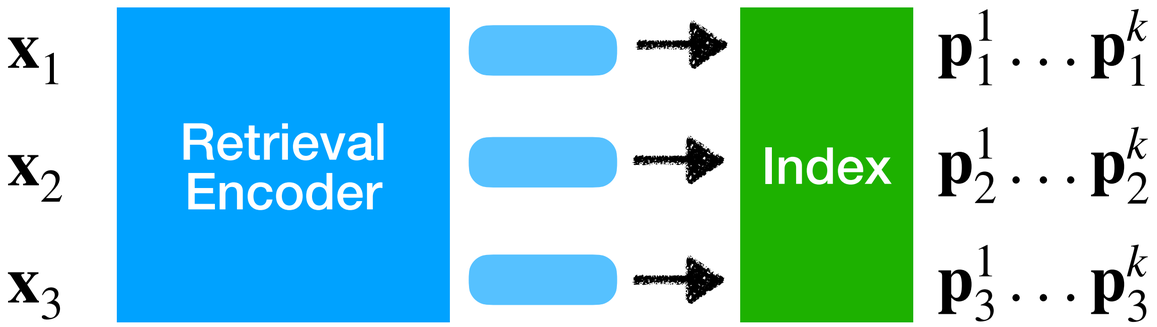
Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

RETRO (Borgeaud et al. 2021)

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)



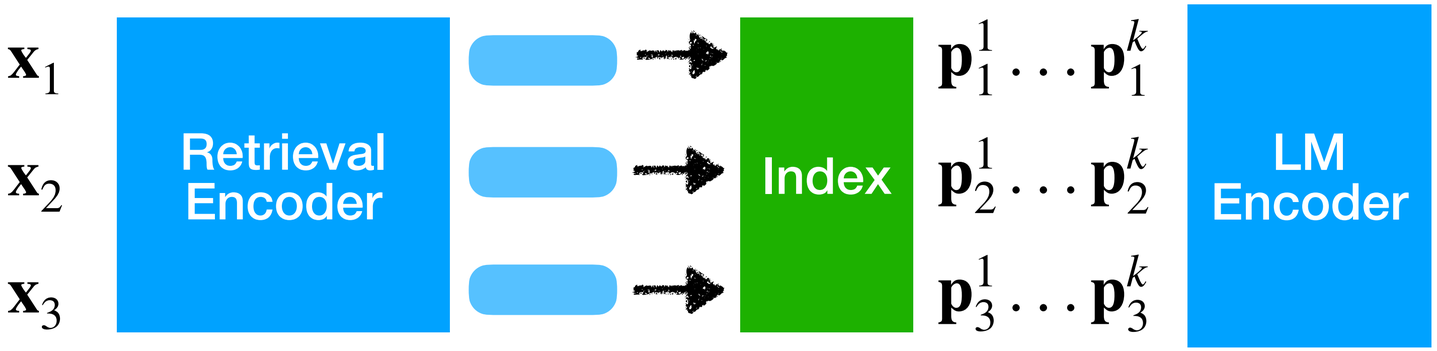
Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

RETRO (Borgeaud et al. 2021)

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)



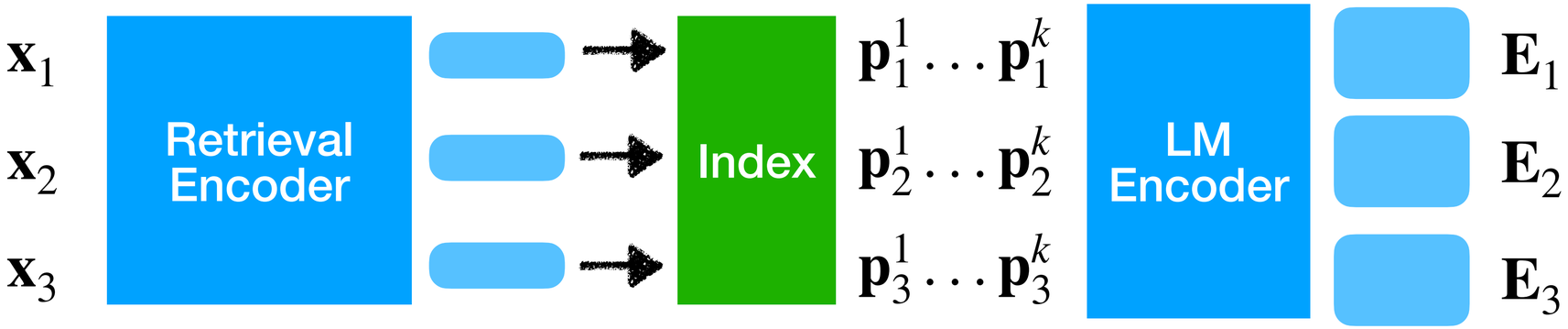
Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

RETRO (Borgeaud et al. 2021)

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)



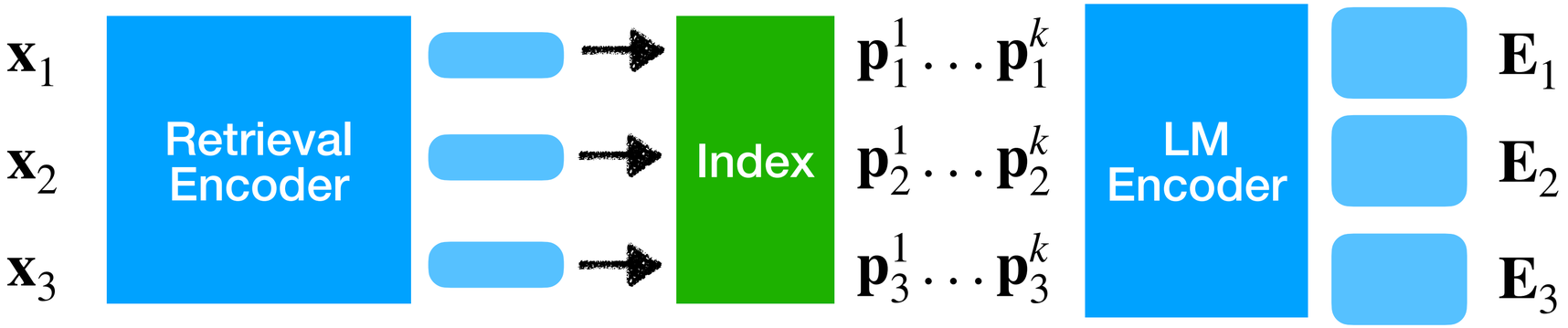
Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

RETRO (Borgeaud et al. 2021)

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)

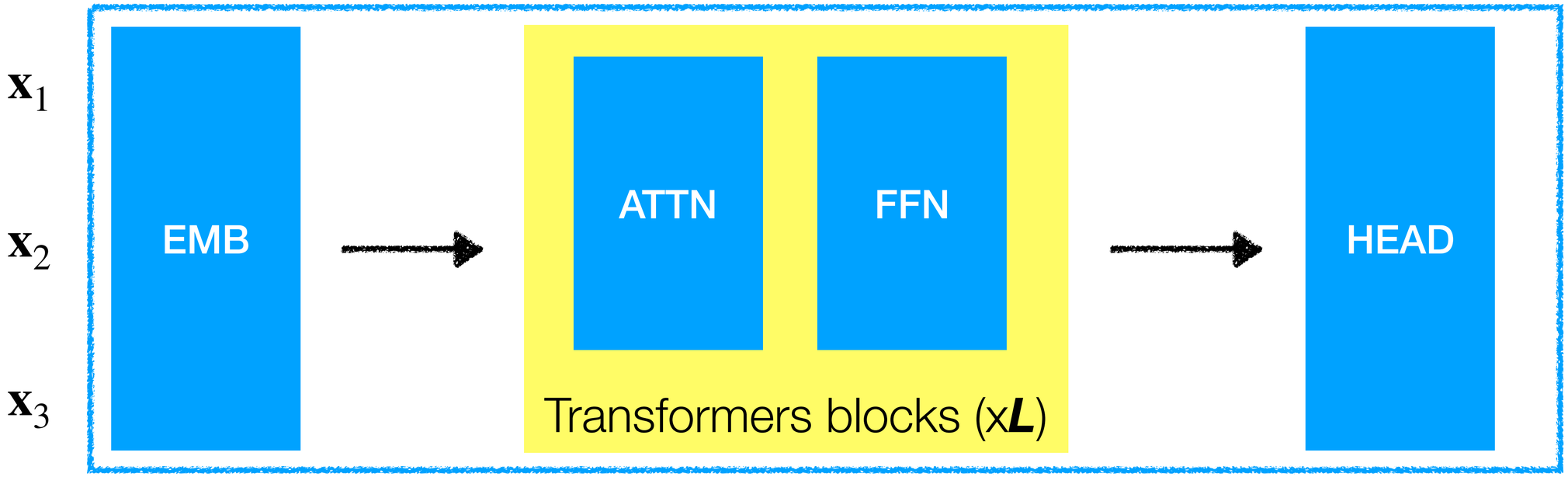


(A $r \times k \times d$ matrix)

(r = # tokens per text chunk)
(d = hidden dimension)
(k = # retrieved chunks per split)

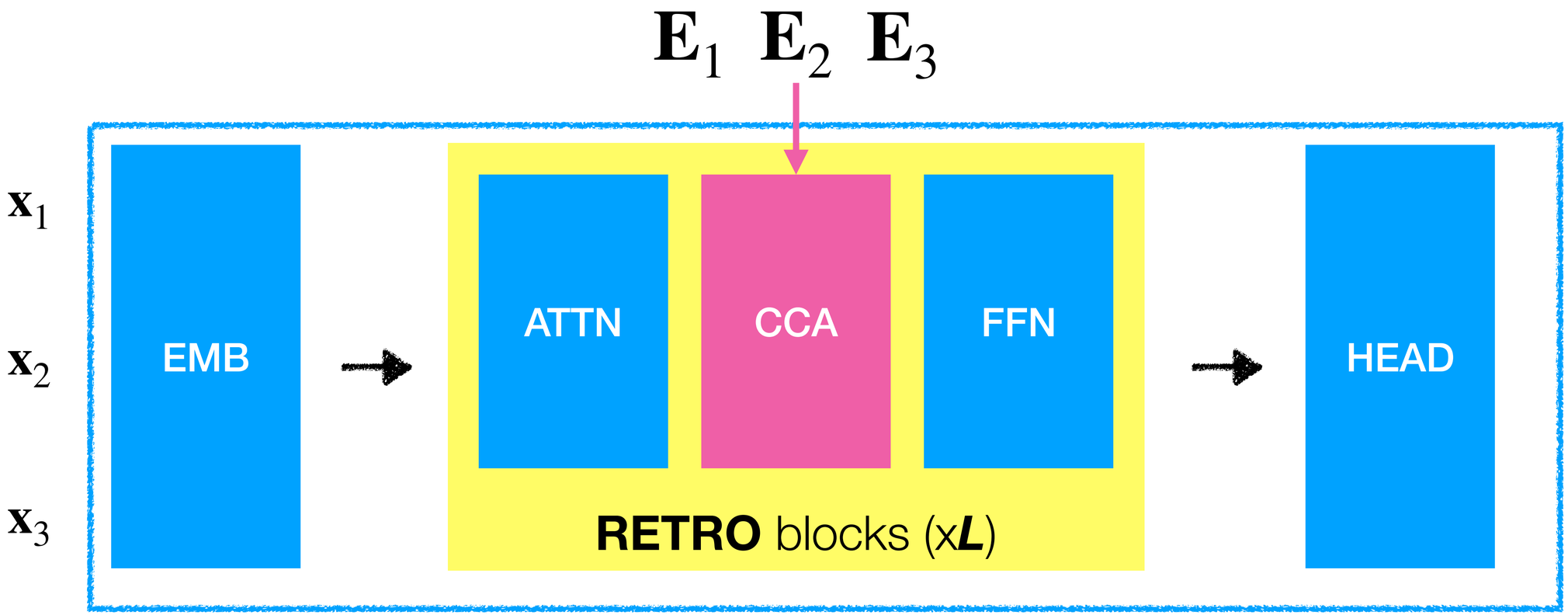
Borgeaud et al. Improving language models by retrieving from trillions of tokens. ICML 2021.

Regular decoder



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

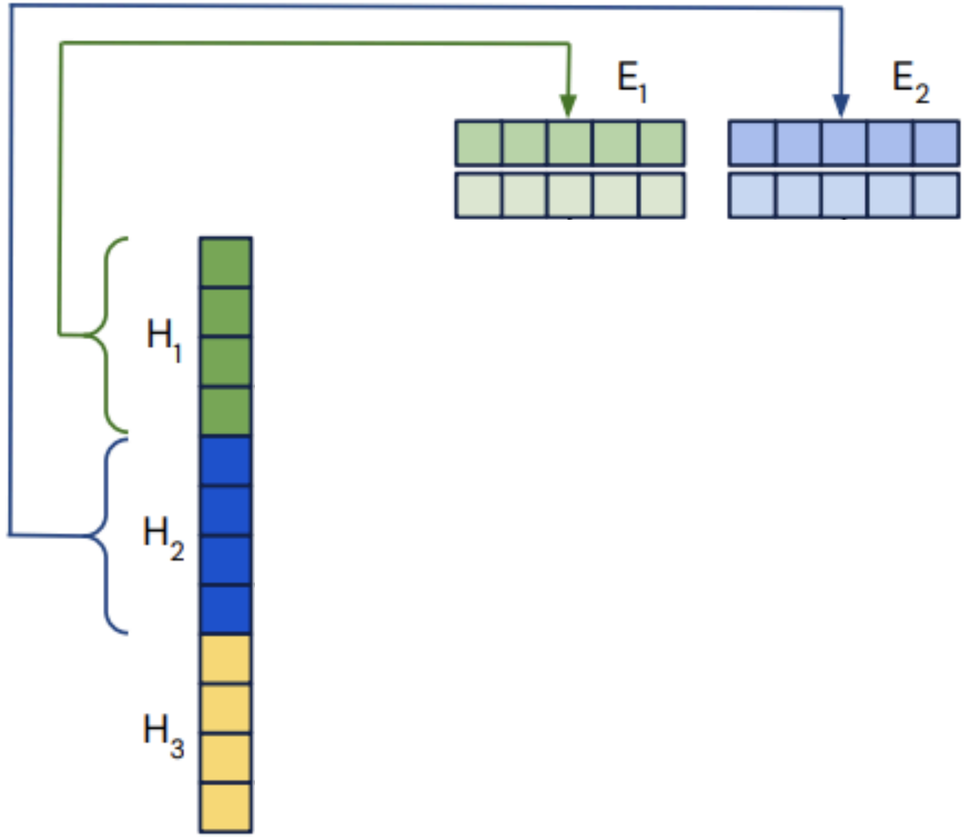
Decoder in RETRO



Chunked Cross Attention (CCA)

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

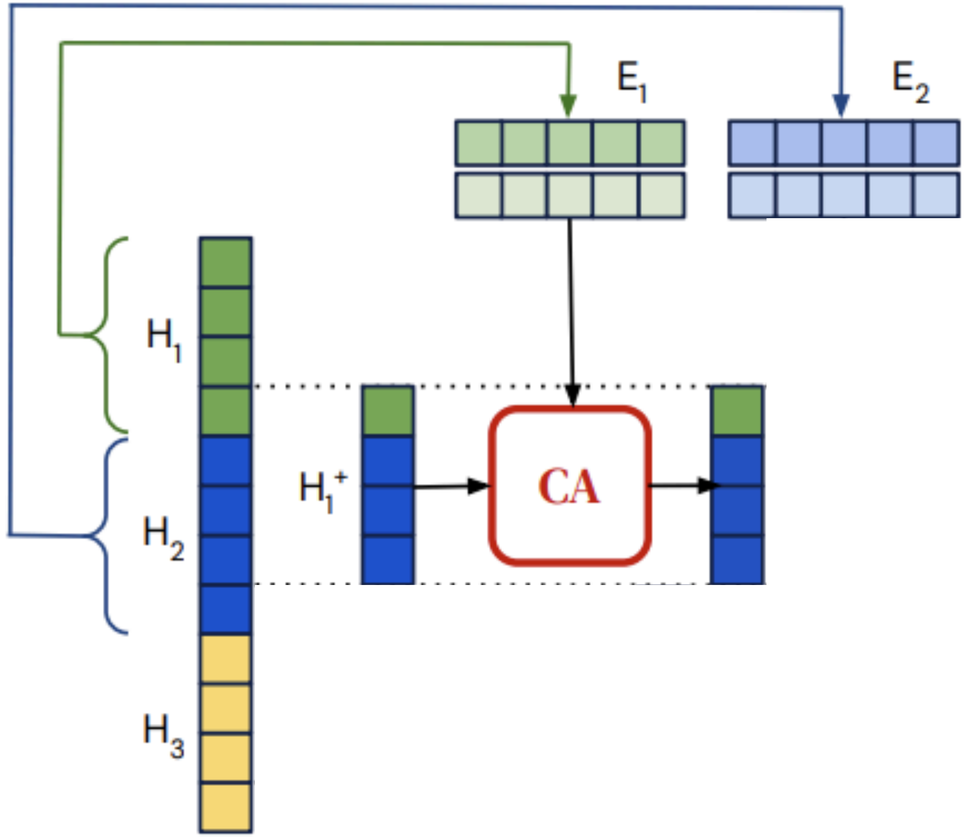
Chunked Cross Attention



Outputs from the previous layer H

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

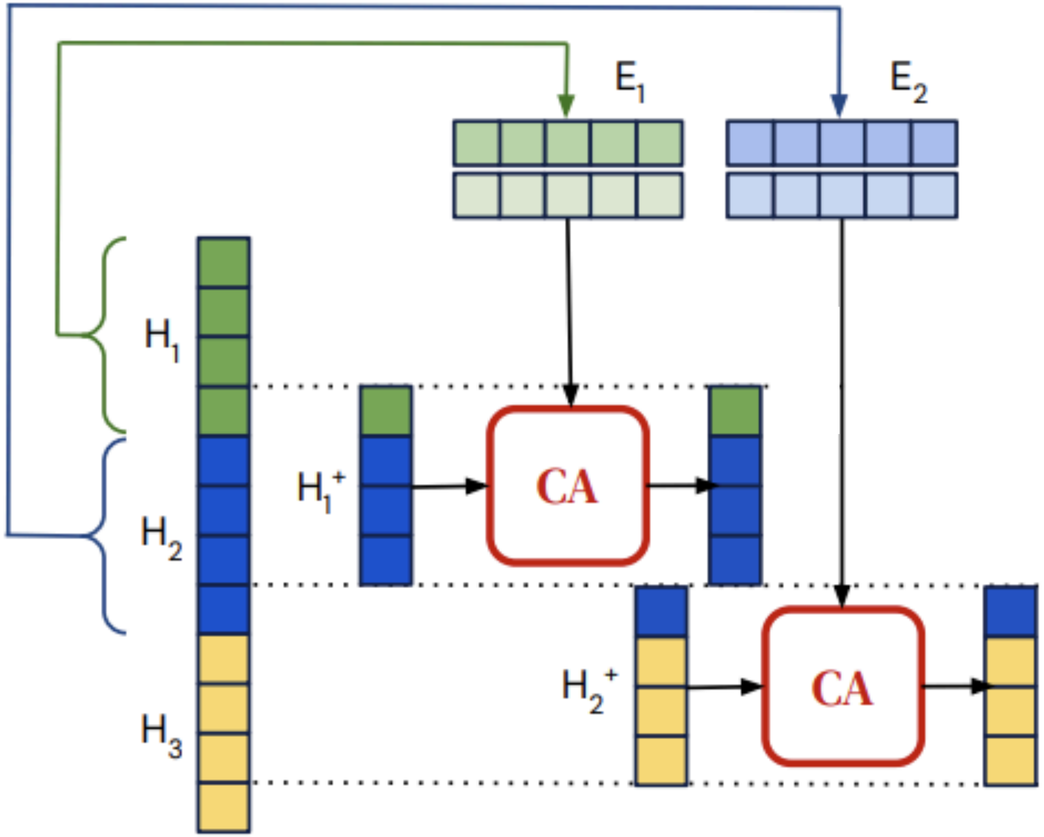
Chunked Cross Attention



Outputs from the previous layer H

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

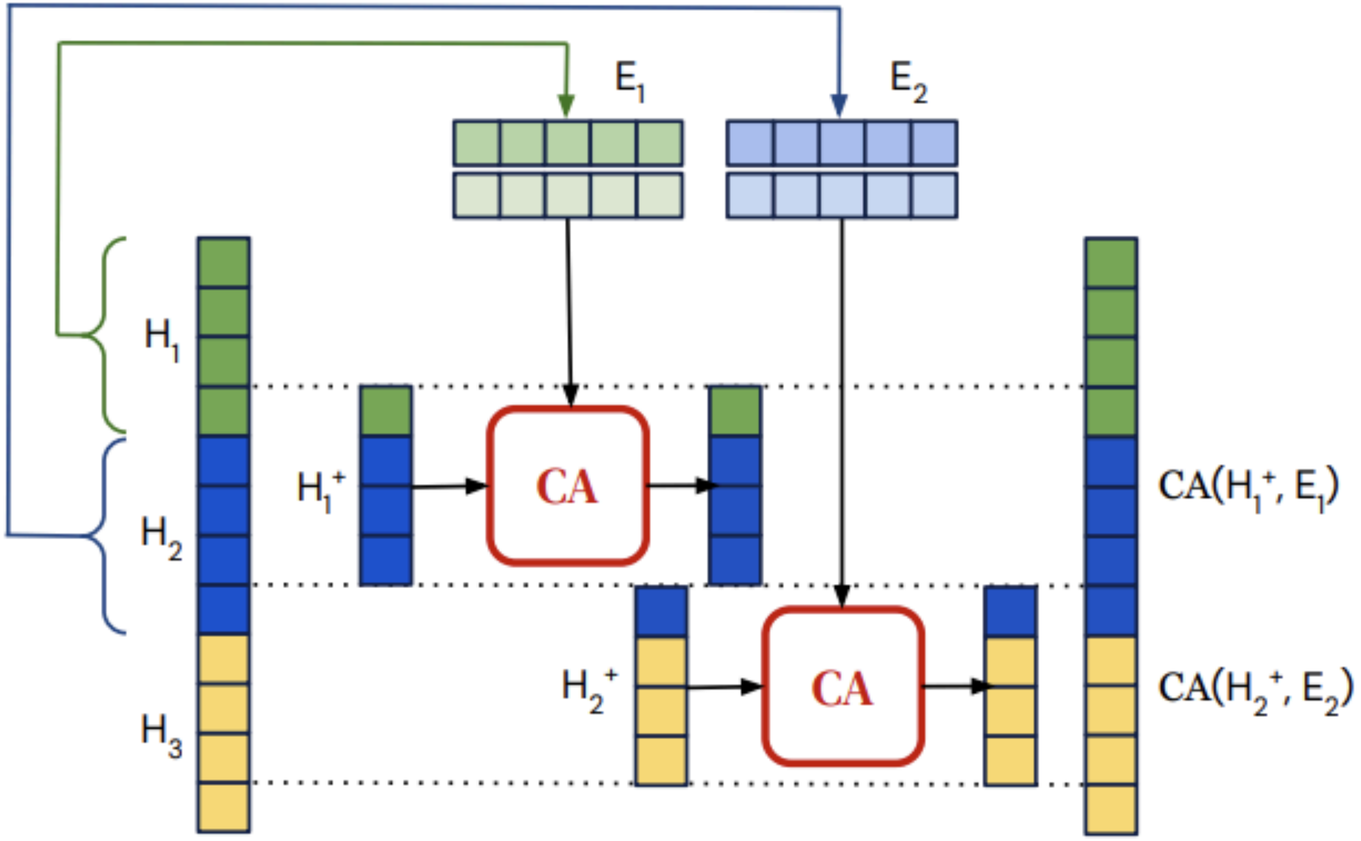
Chunked Cross Attention



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
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

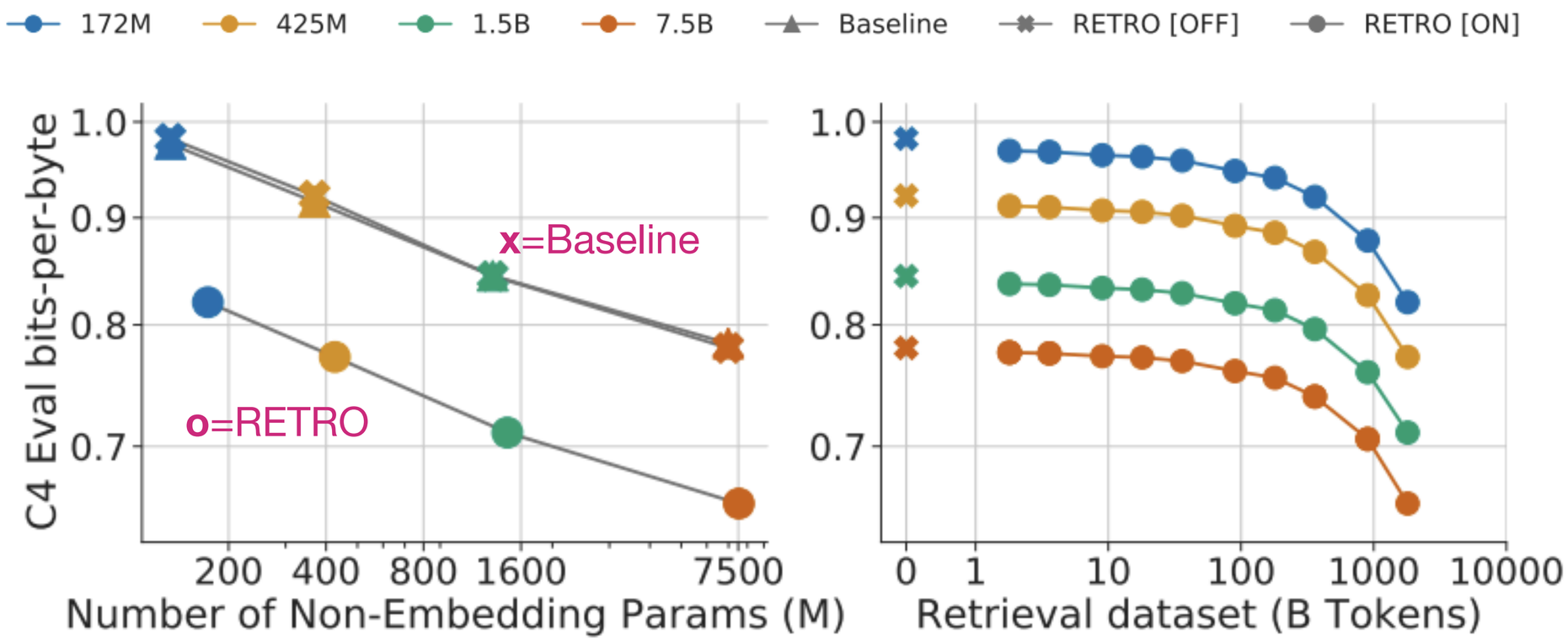
Perplexity: The lower the better

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)

kNN-LM (Khandelwal et al. 2020)

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

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.

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

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

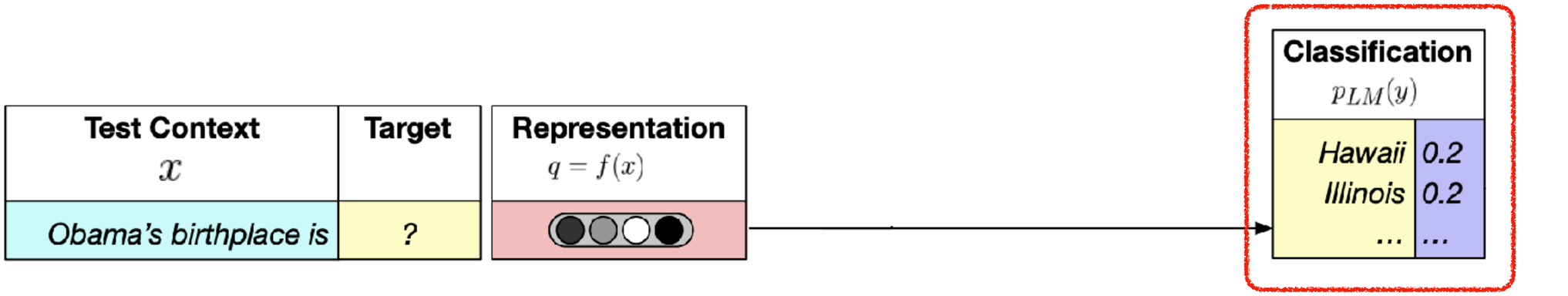
Khandelwal et al. Generalization through Memorization: Nearest Neighbor Language Models. ICLR 2020.

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)



Parametric distribution

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

kNN-LM (Khandelwal et al. 2020)

... 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)

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.

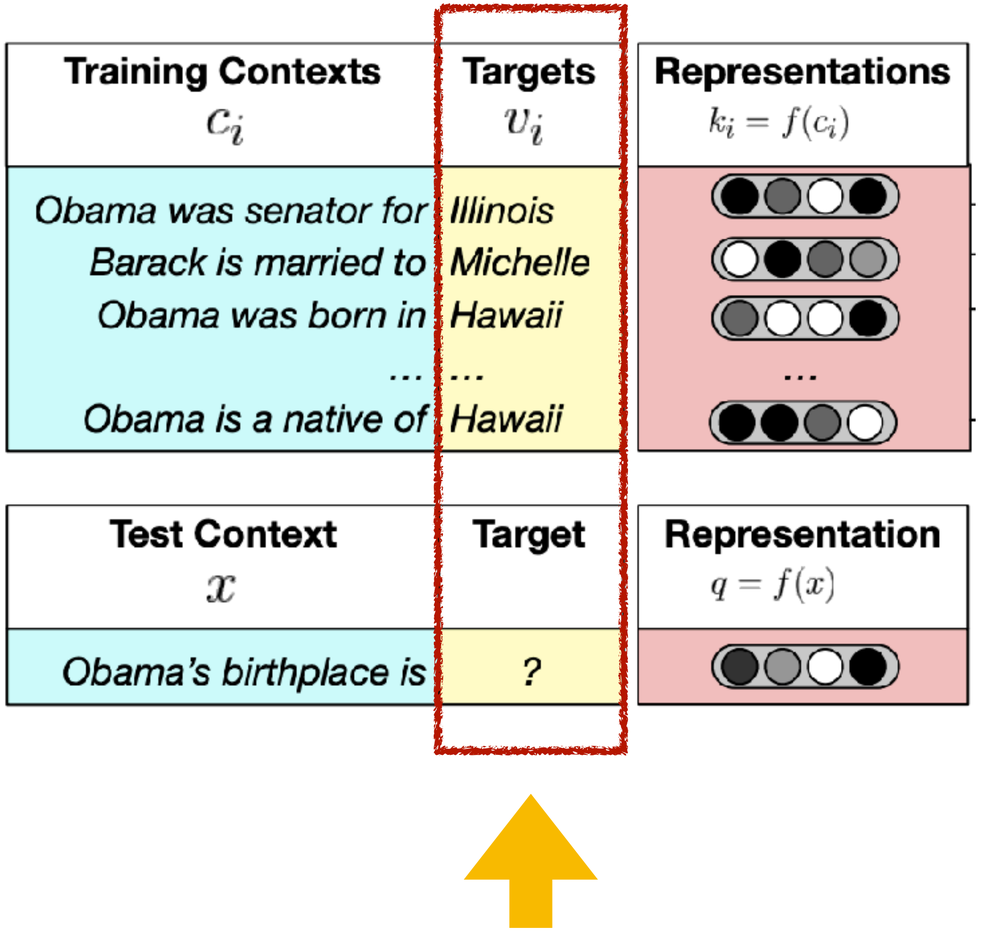
kNN-LM (Khandelwal et al. 2020)

Training Contexts c_i	Targets v_i	Representations $k_i = f(c_i)$
Obama was senator for	*Illinois*	
Barack is married to	*Michelle*	
Obama was born in	*Hawaii*	
...	...	...
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.

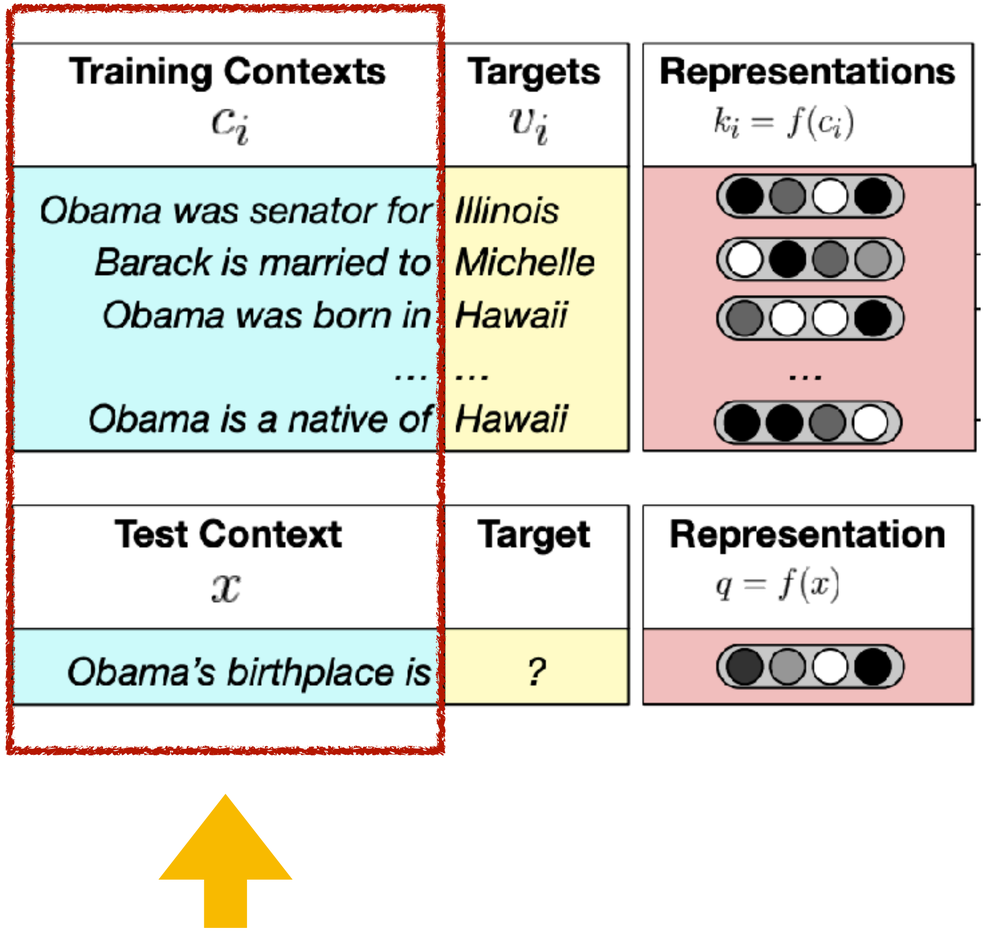
kNN-LM (Khandelwal et al. 2020)



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

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

kNN-LM (Khandelwal et al. 2020)



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

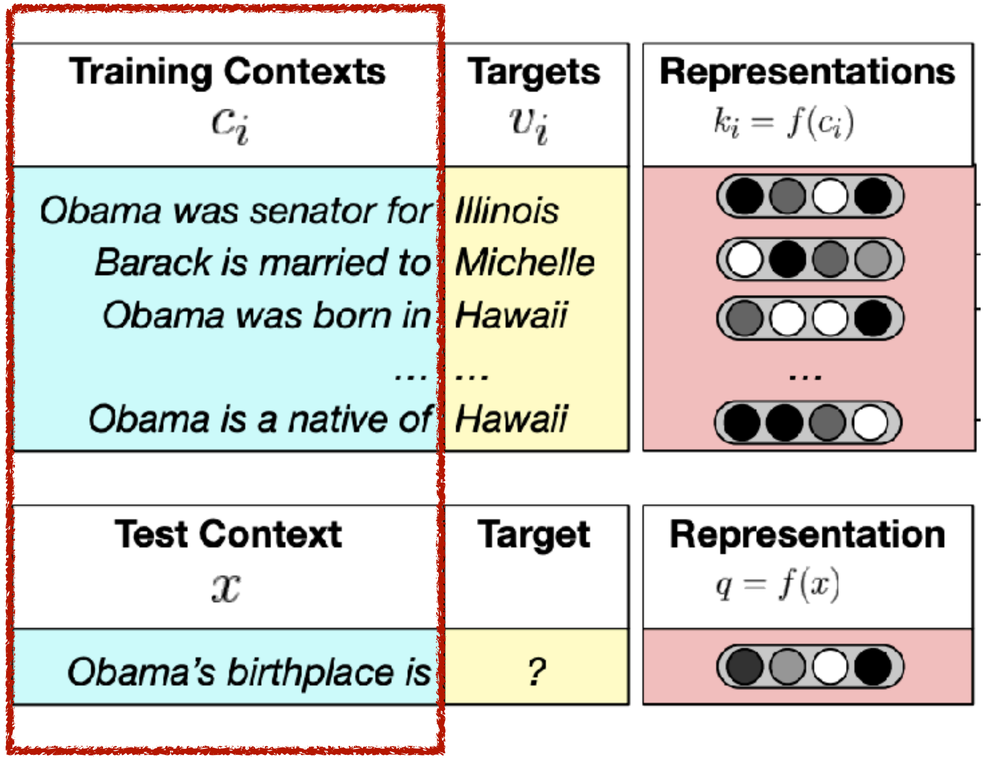
=

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)

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



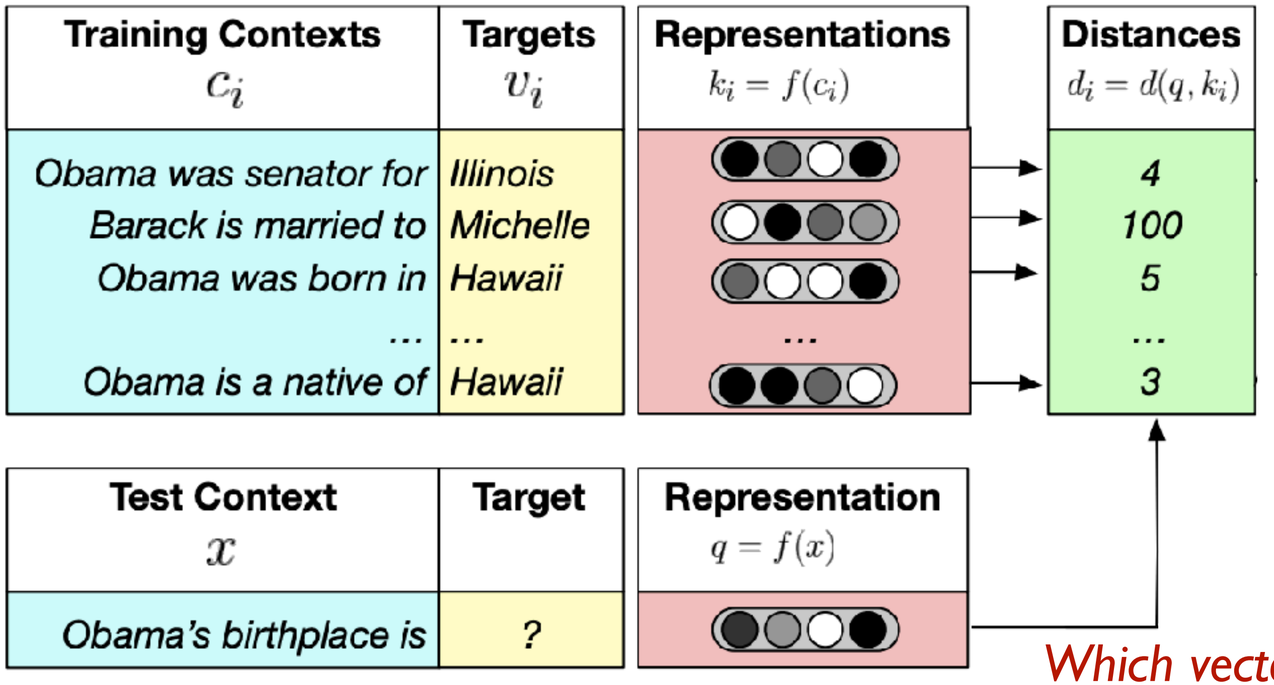
Which tokens <u>in a datastore</u> are close to the next token?

=

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

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

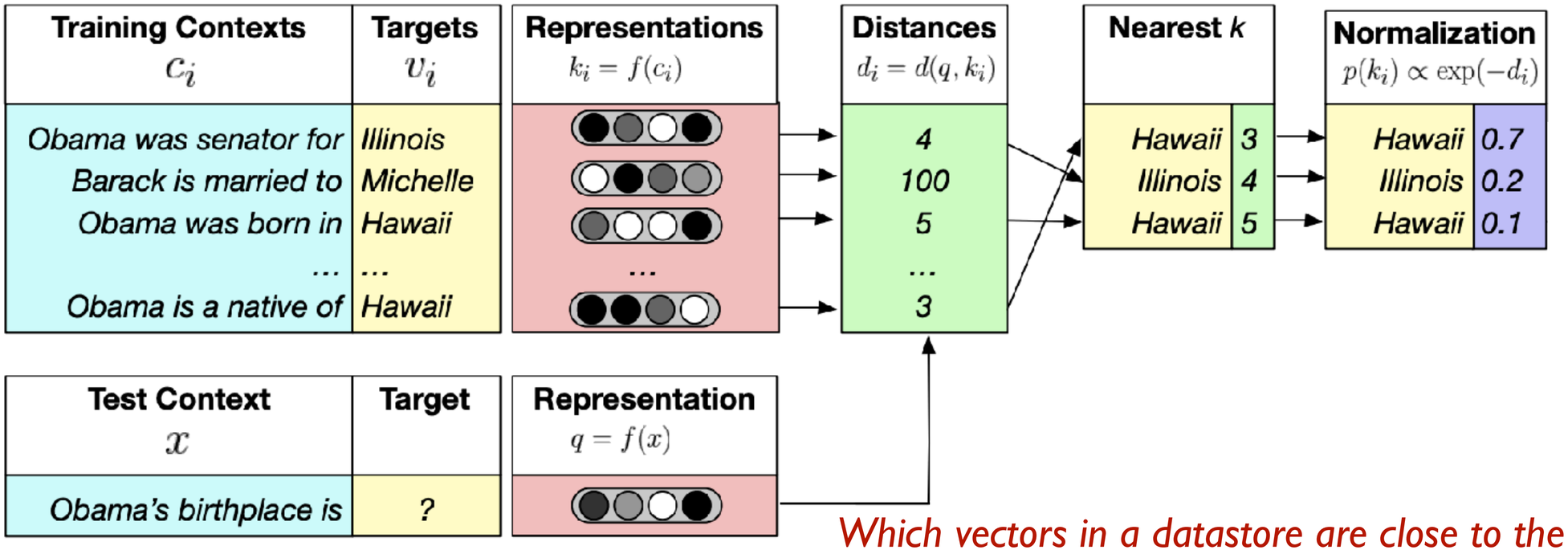
kNN-LM (Khandelwal et al. 2020)



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

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

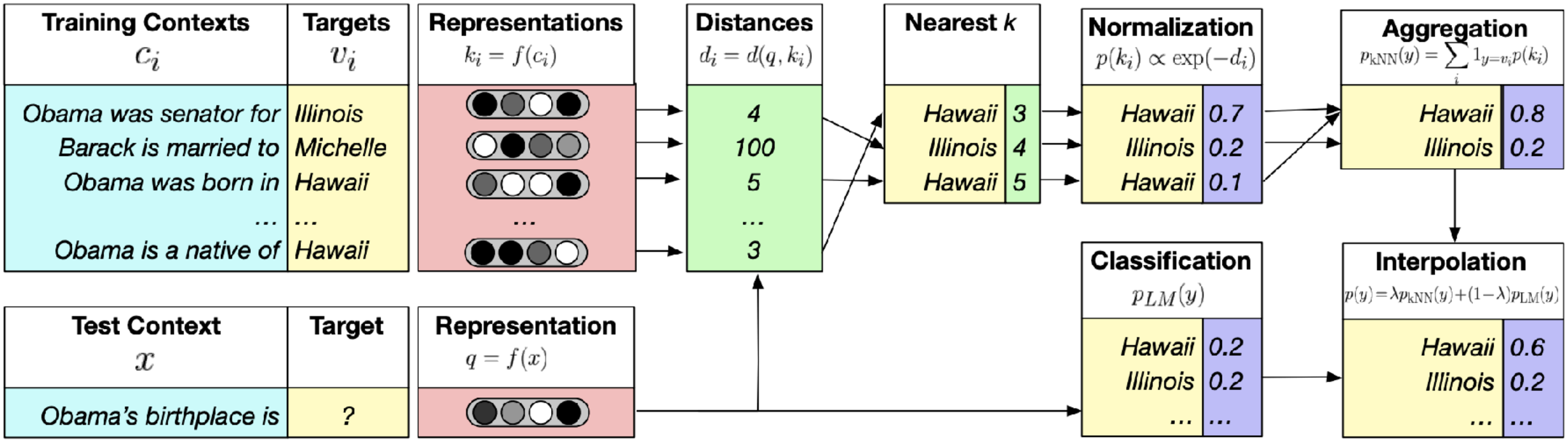
kNN-LM (Khandelwal et al. 2020)



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

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

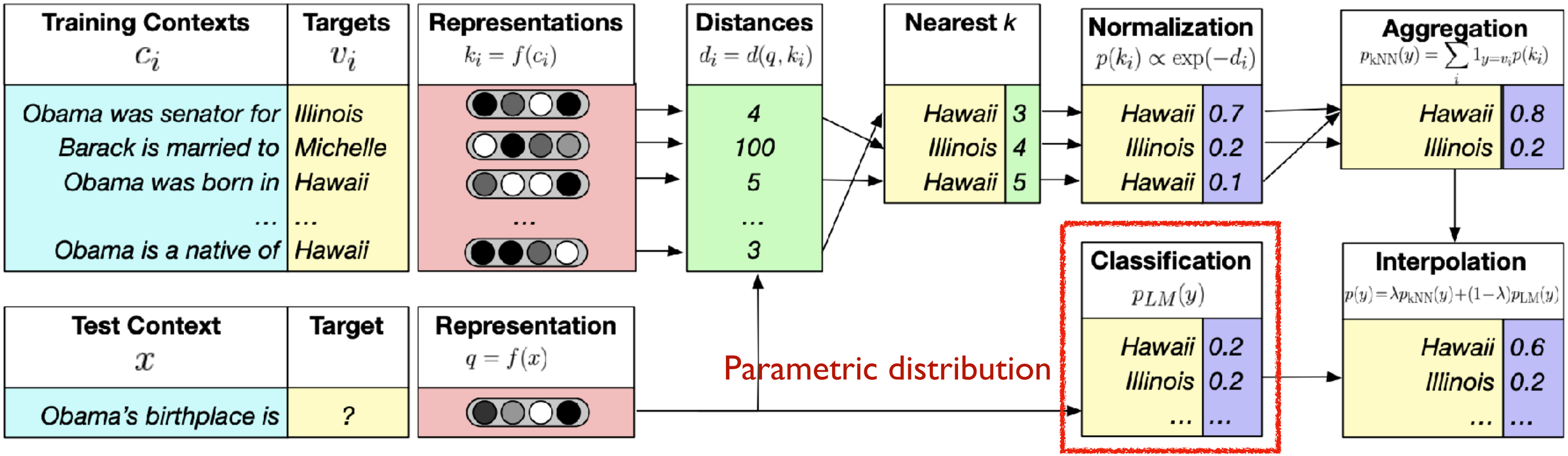
kNN-LM (Khandelwal et al. 2020)



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

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

kNN-LM (Khandelwal et al. 2020)

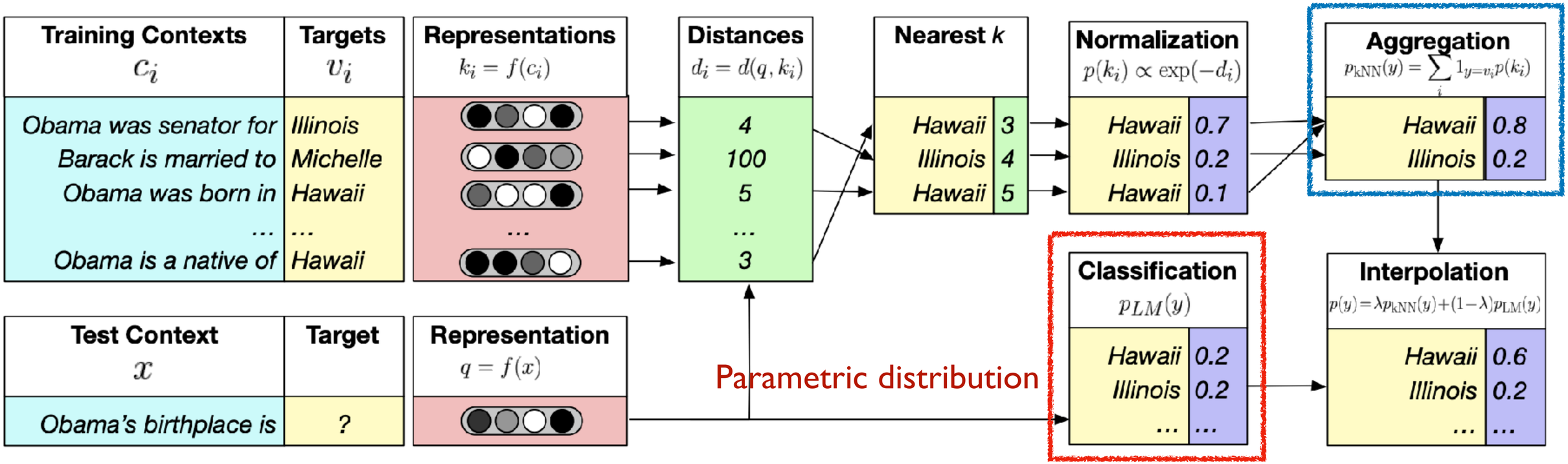


$$P_{k\text{NN-LM}}(y \mid x) = (1 - \lambda)\underline{P_{\text{LM}}(y \mid x)} + \lambda P_{k\text{NN}}(y \mid x)$$

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

kNN-LM (Khandelwal et al. 2020)

Nonparametric distribution

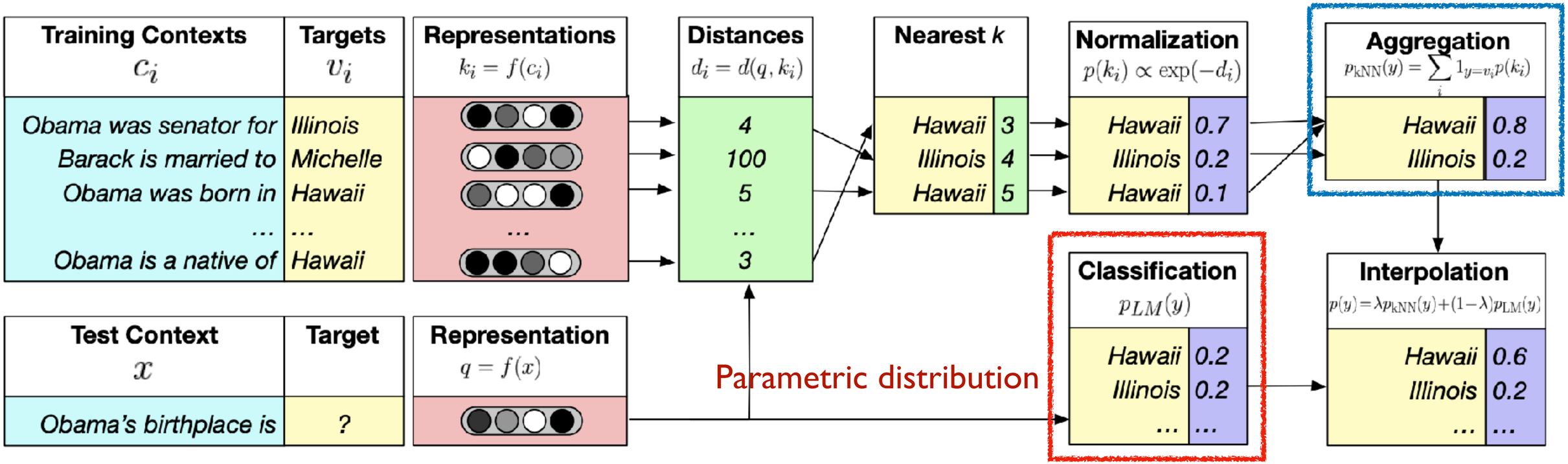


$$P_{k\text{NN-LM}}(y \mid x) = (1 - \lambda) P_{\text{LM}}(y \mid x) + \lambda P_{k\text{NN}}(y \mid x)$$

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

kNN-LM (Khandelwal et al. 2020)

Nonparametric distribution



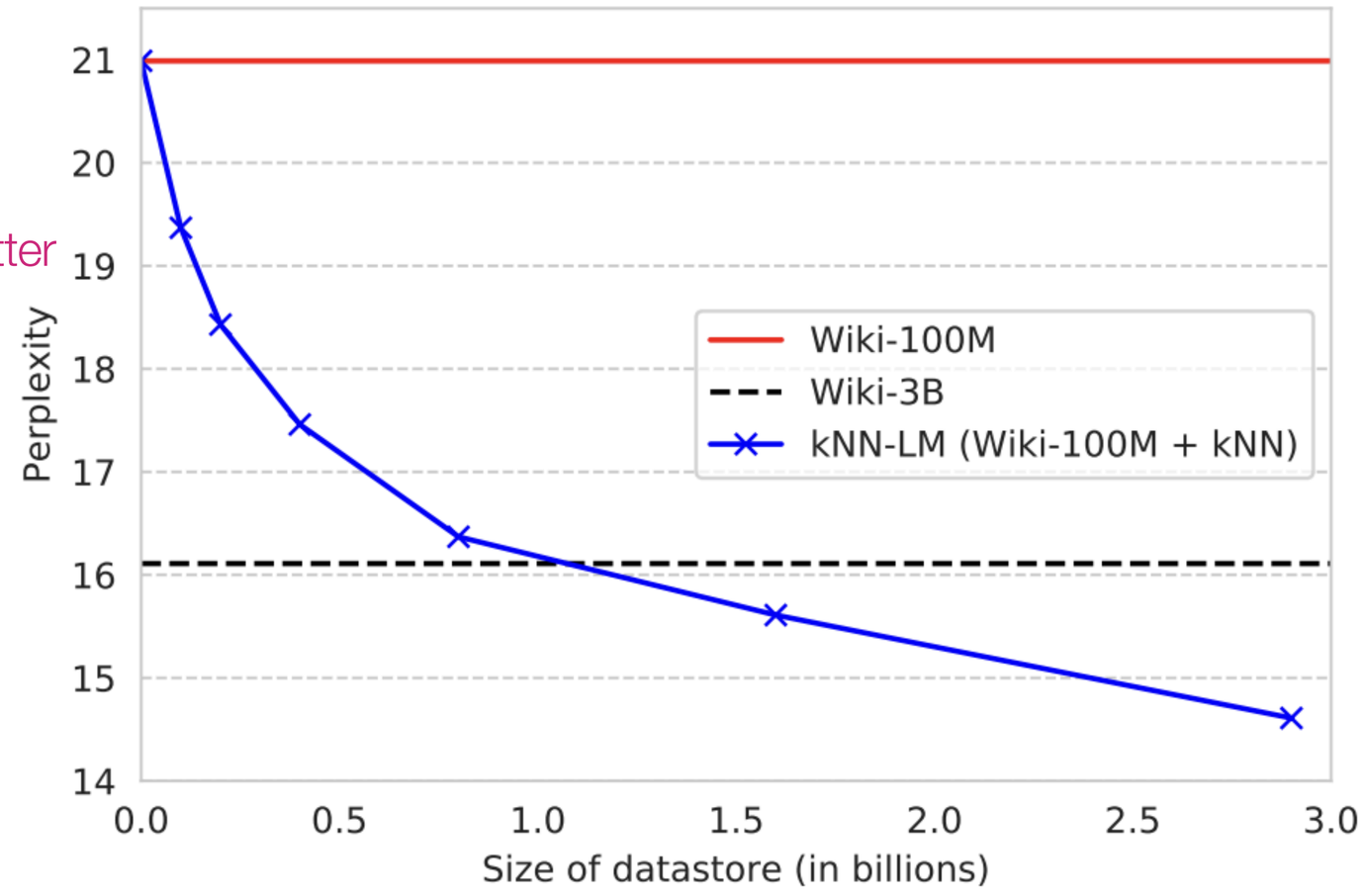
λ: hyperparameter

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

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

kNN-LM - results

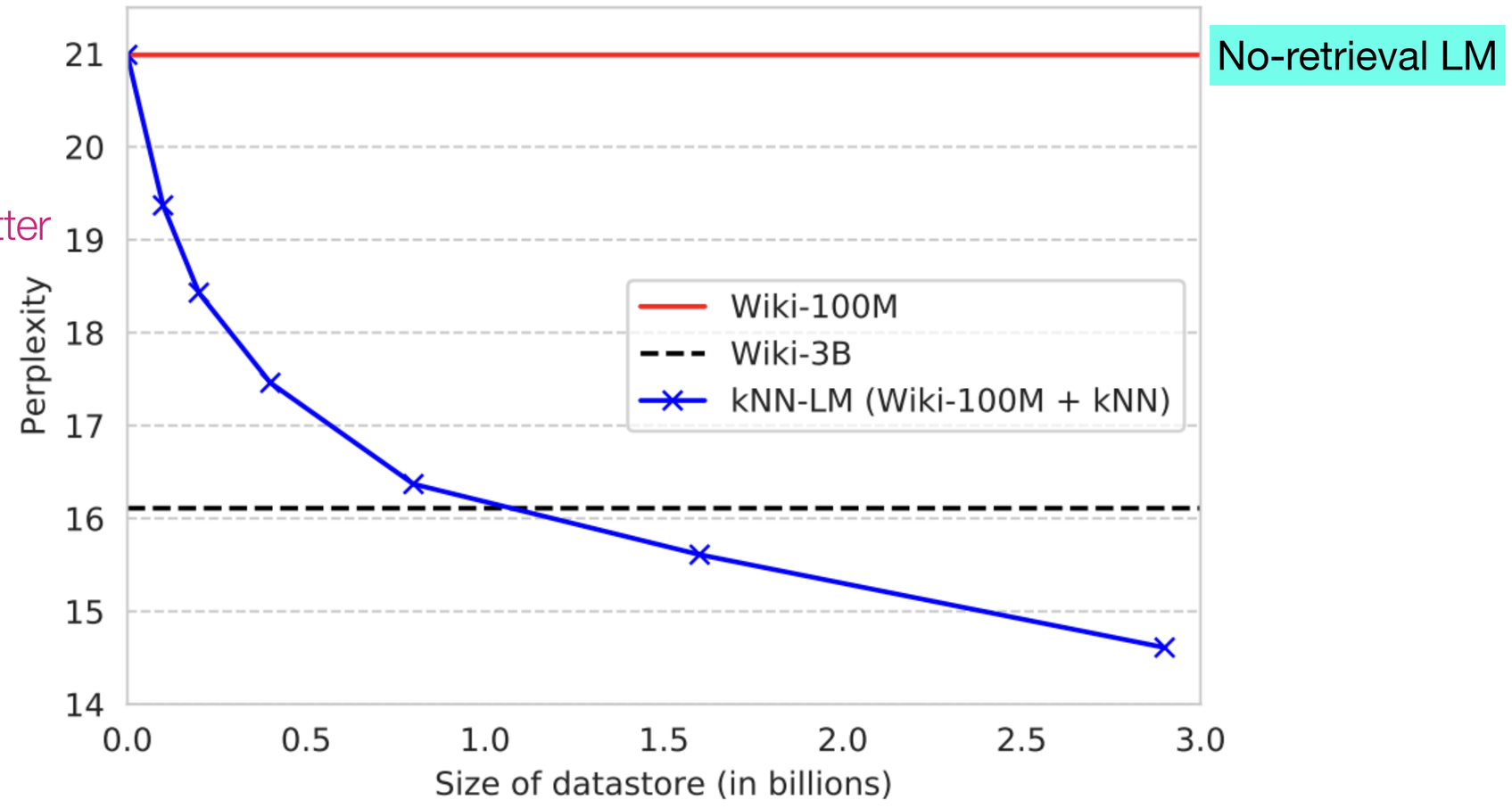
The lower the better



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

kNN-LM - results

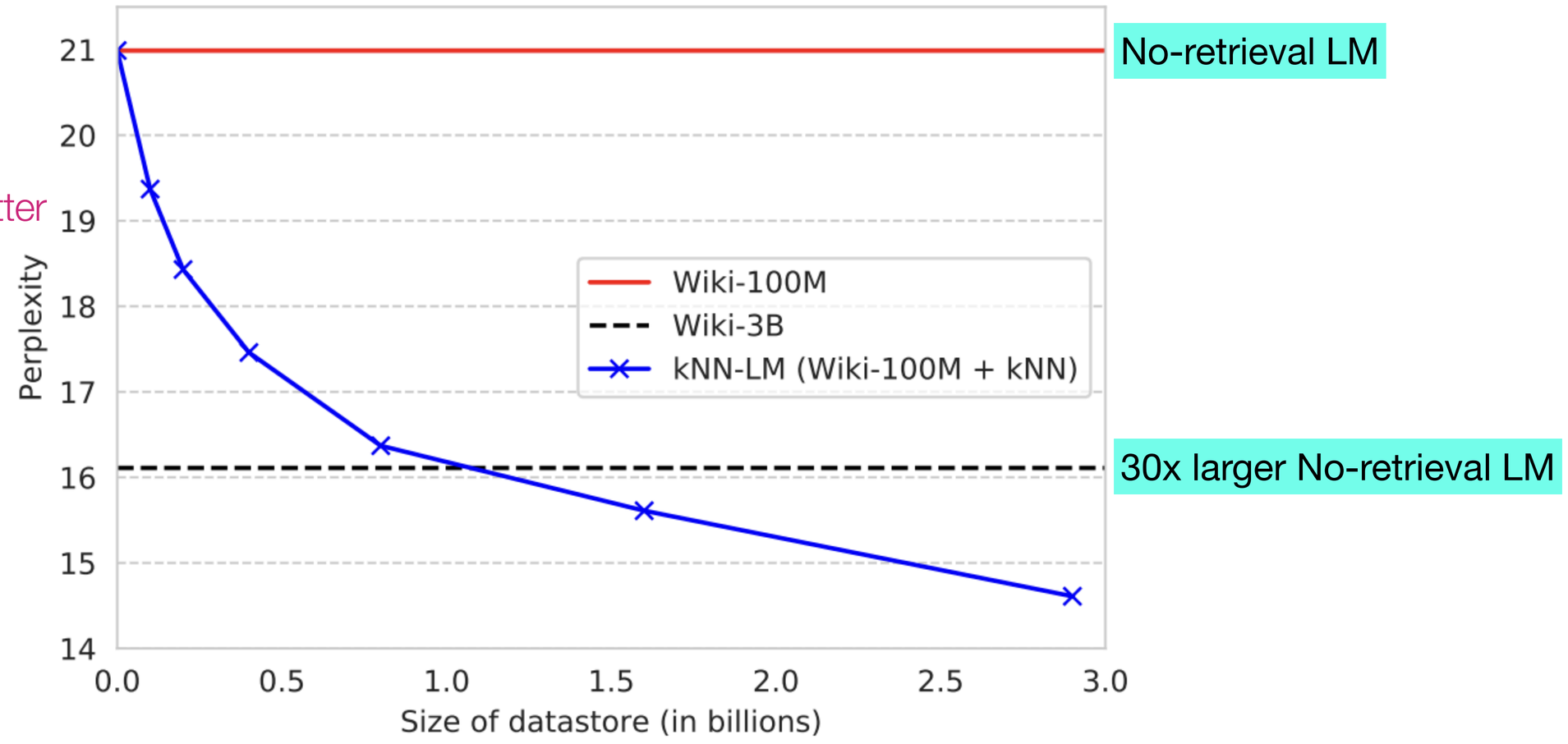
The lower the better



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

kNN-LM - results

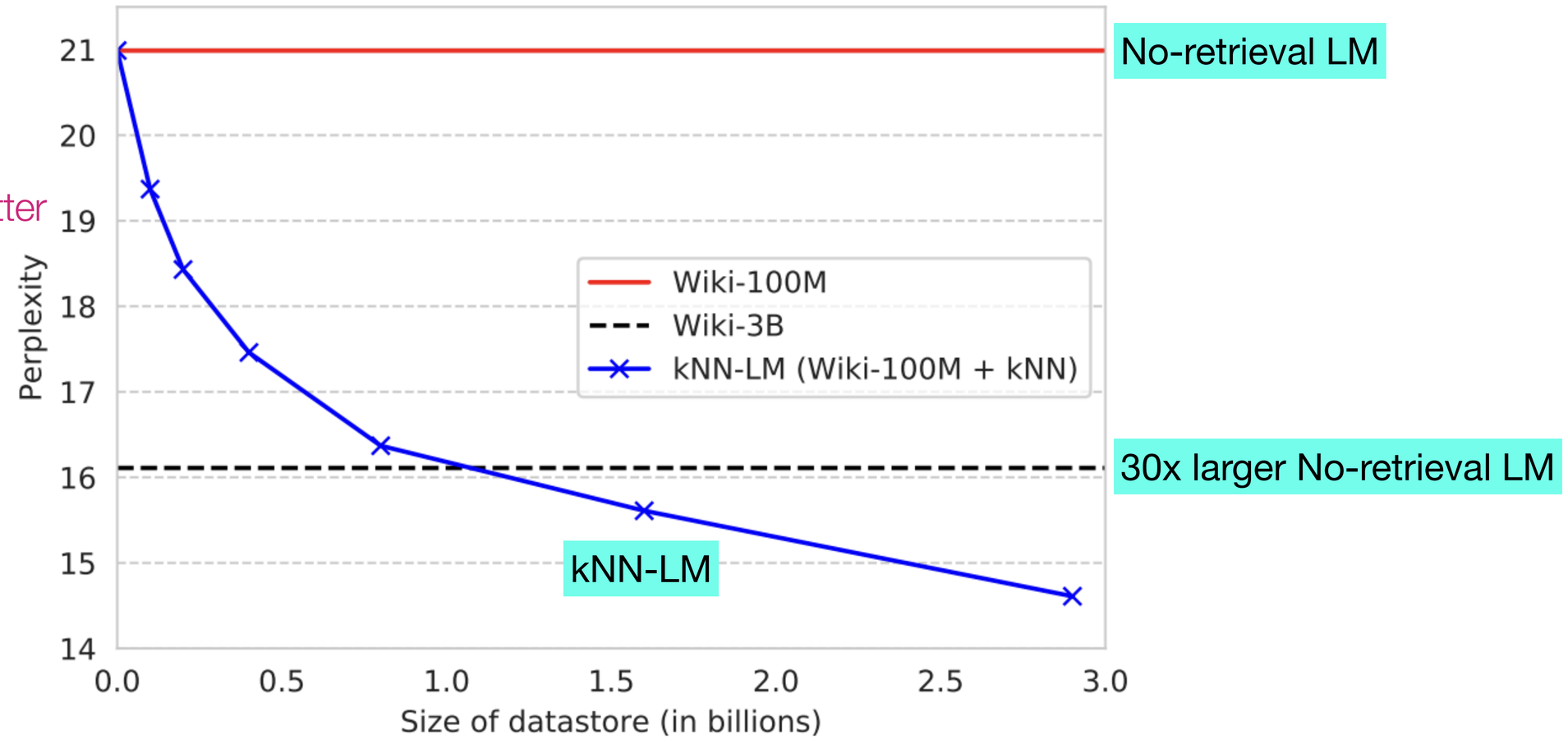
The lower the better



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

kNN-LM - results

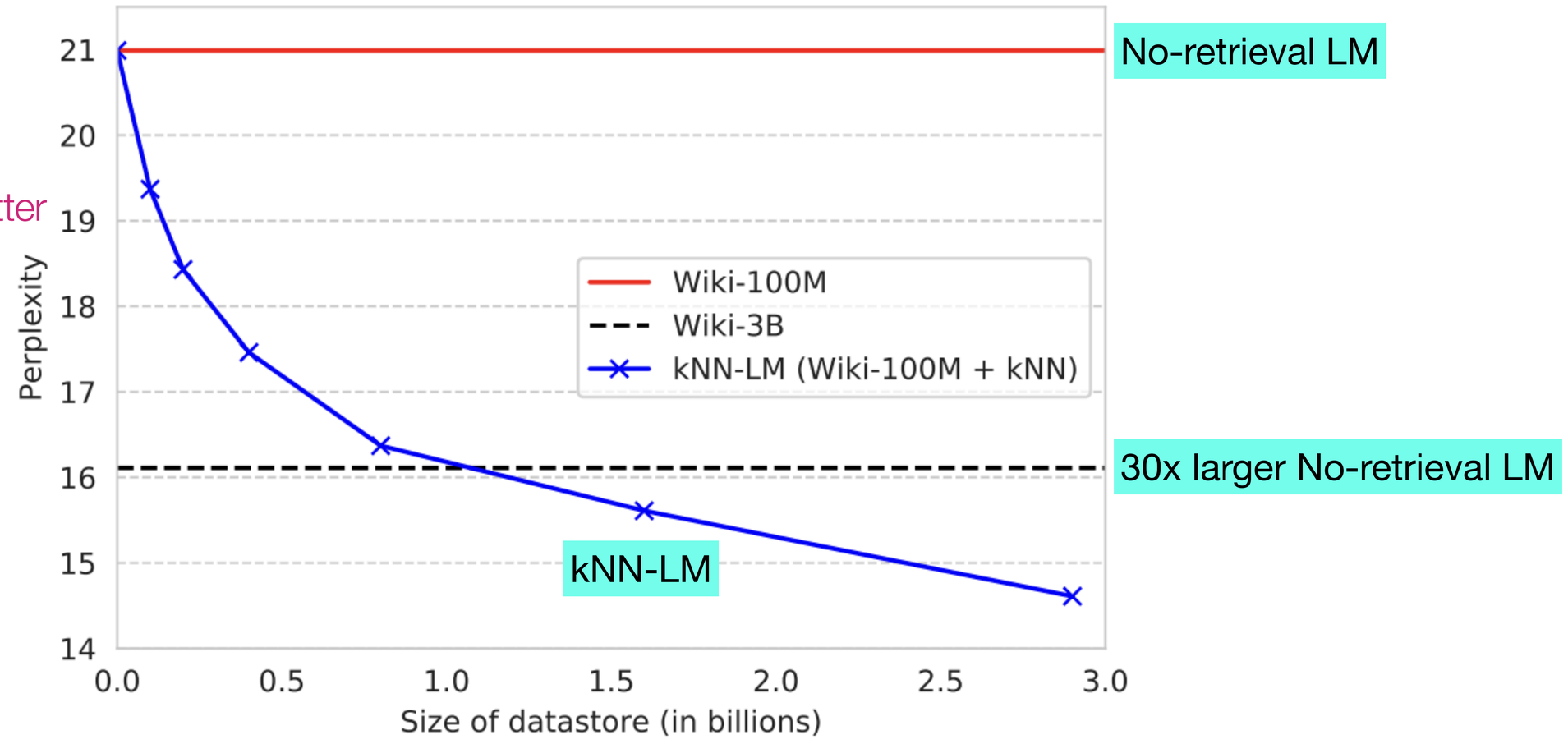
The lower the better



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

kNN-LM - results

The lower the better

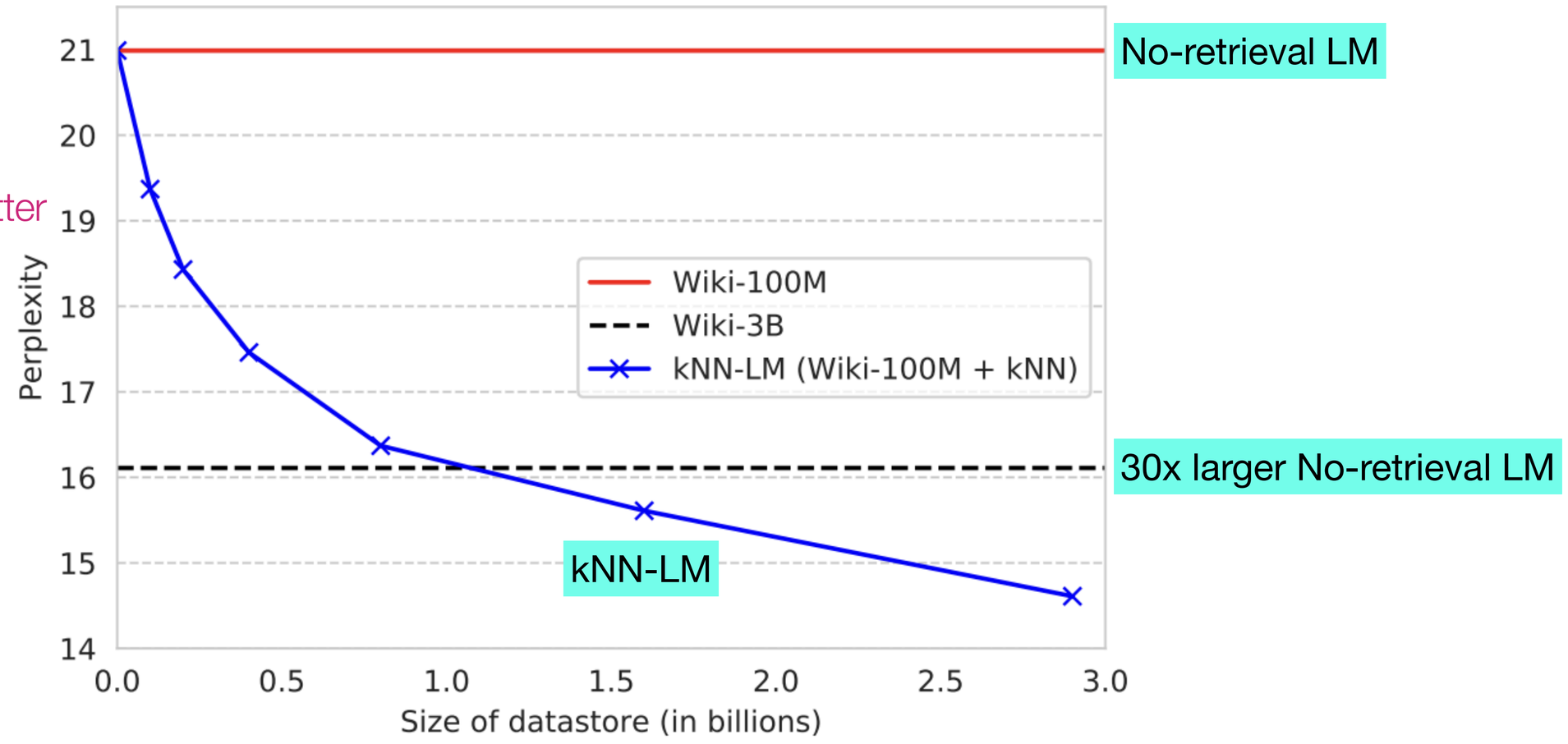


Outperforms no-retrieval LM

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)

Triggering Retrieval with Tokens

- Toolformer (Schick et al. 2023) generates tokens that trigger retrieval (or other tools)
- Training is done in an iterative manner - generate and identify successful retrievals

The New England Journal of Medicine is a registered trademark of **[QA("Who is the publisher of The New England Journal of Medicine?") → Massachusetts Medical Society]** the MMS.

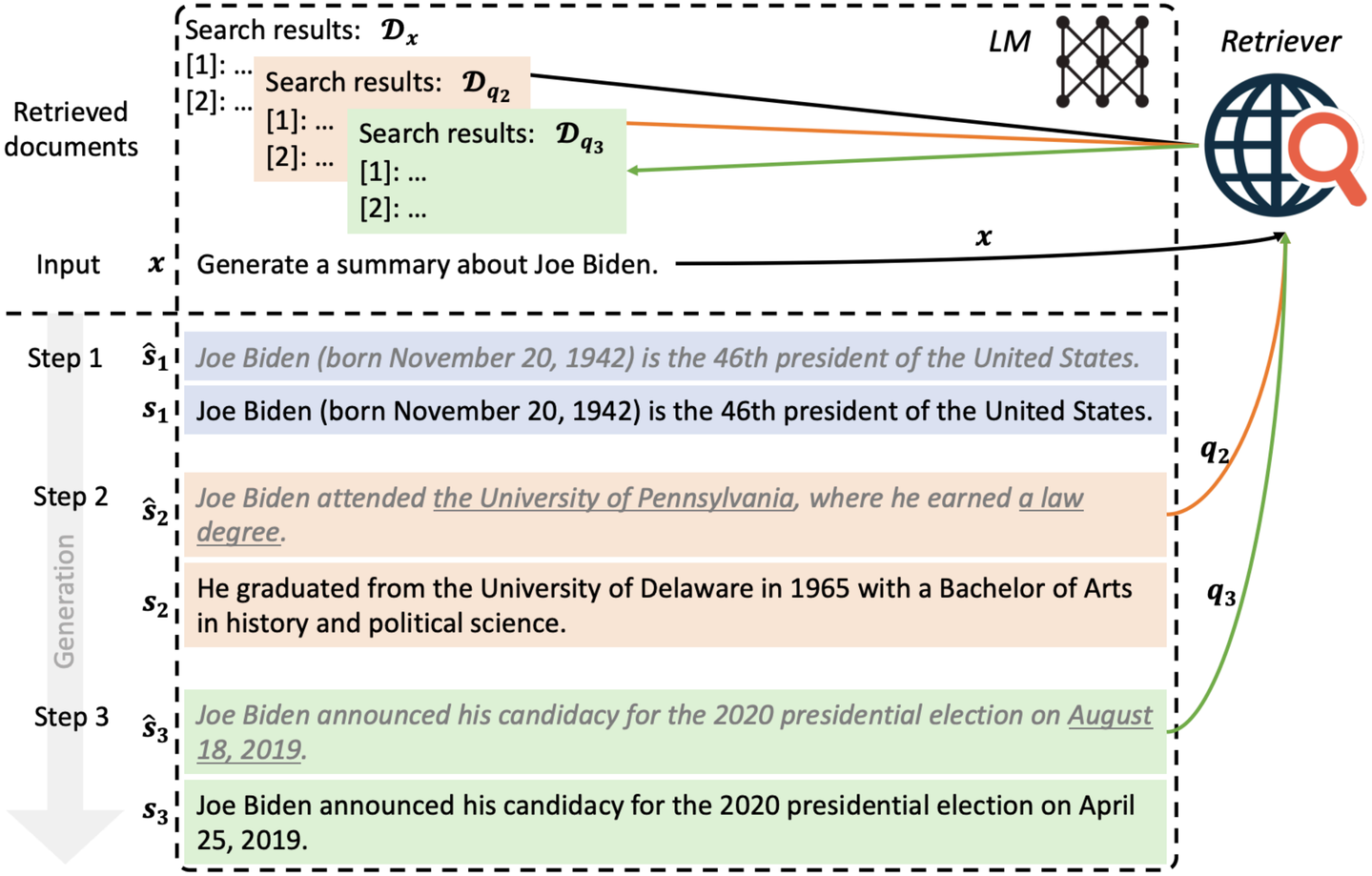
Out of 1400 participants, 400 (or **[Calculator(400 / 1400) → 0.29]** 29%) passed the test.

The name derives from "la tortuga", the Spanish word for **[MT("tortuga") → turtle]** turtle.

The Brown Act is California's law **[WikiSearch("Brown Act") → The Ralph M. Brown Act is an act of the California State Legislature that guarantees the public's right to attend and participate in meetings of local legislative bodies.]** that requires legislative bodies, like city councils, to hold their meetings open to the public.

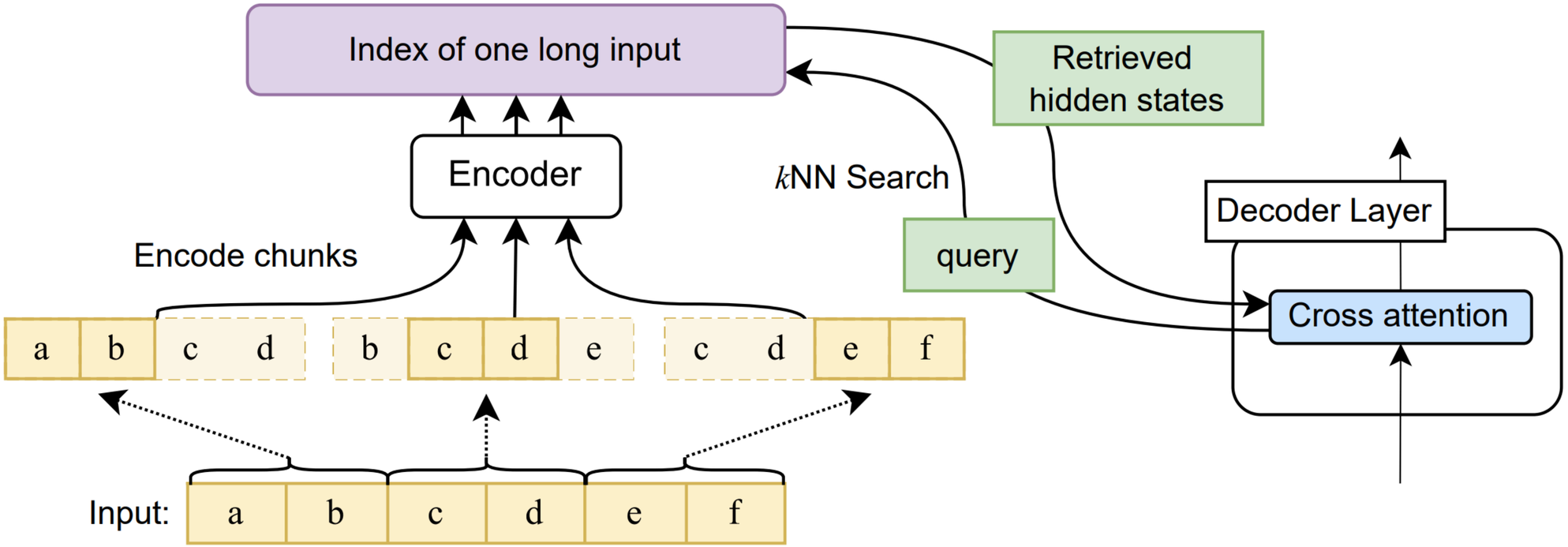
Triggering Retrieval with Uncertainty

FLARE (Jiang et al. 2023) tries to generate content, then does retrieval if LM certainty is low



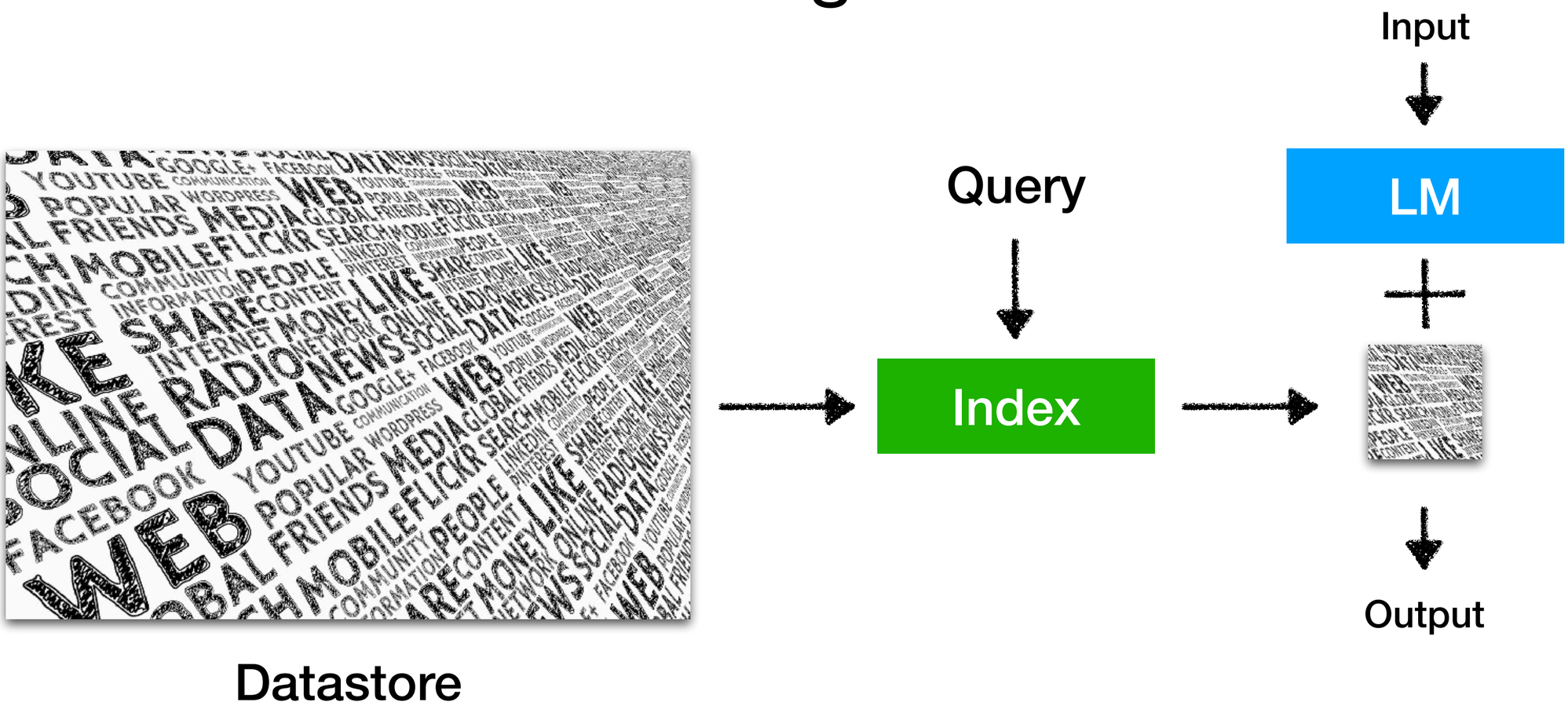
Token-Level Approximate Attention

- Unlimiformer (Bertsch et al. 2023) notes that attention is an inner-product and does top-k attention
- First, process input with a sliding window
- Then perform attention using a vector index

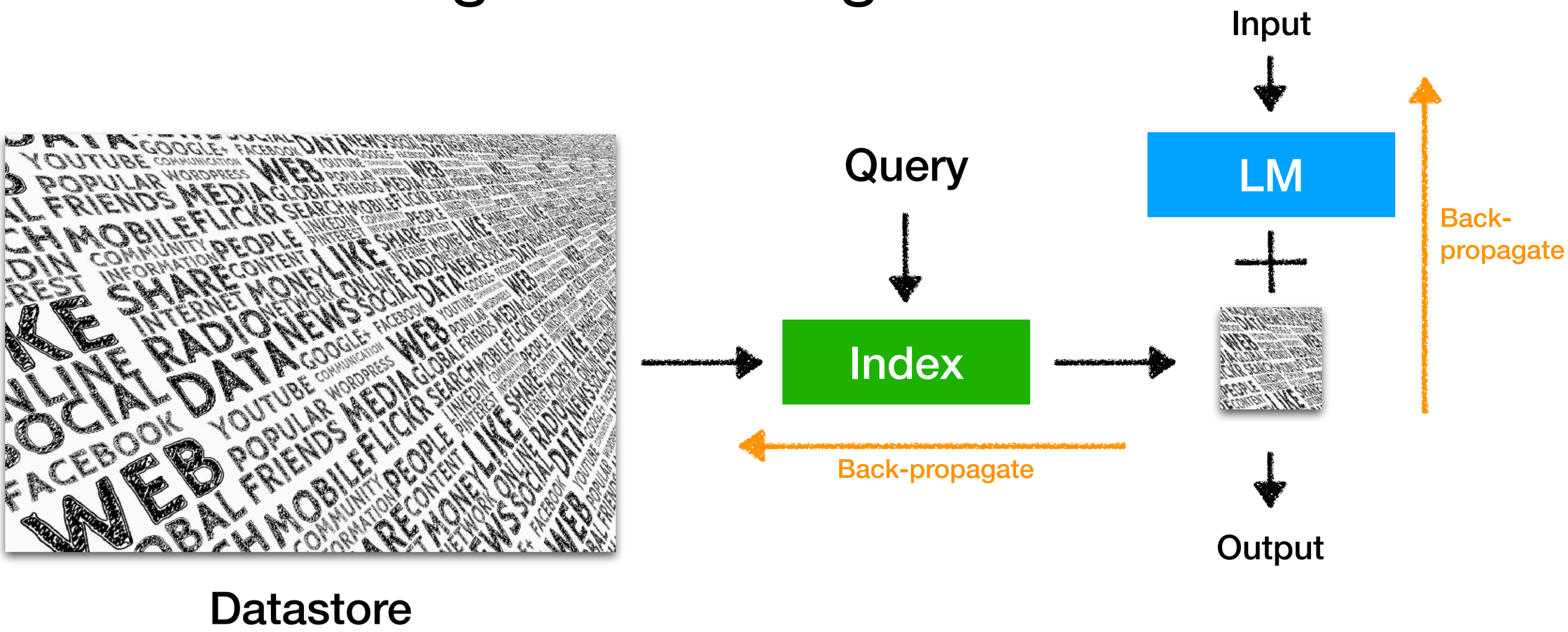


Training

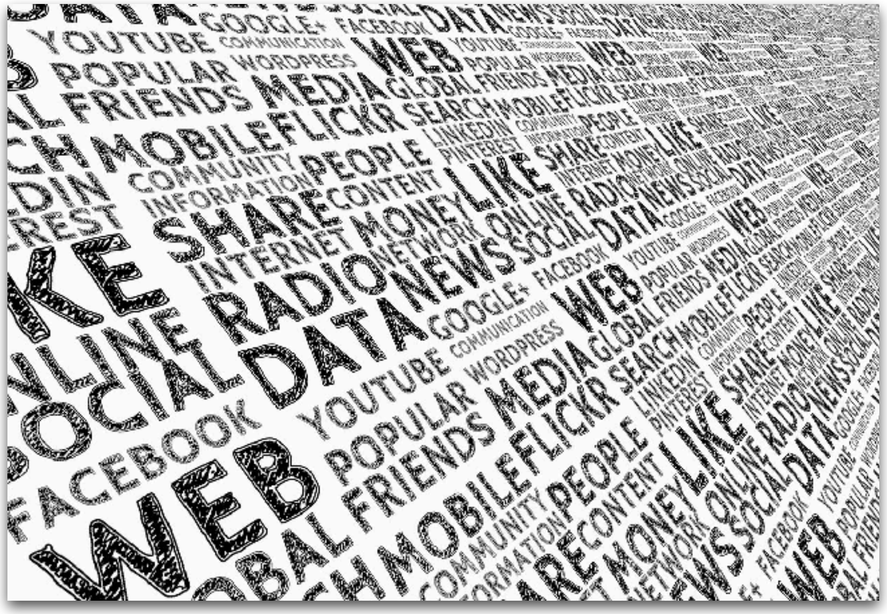
Retrieval-augmented LMs



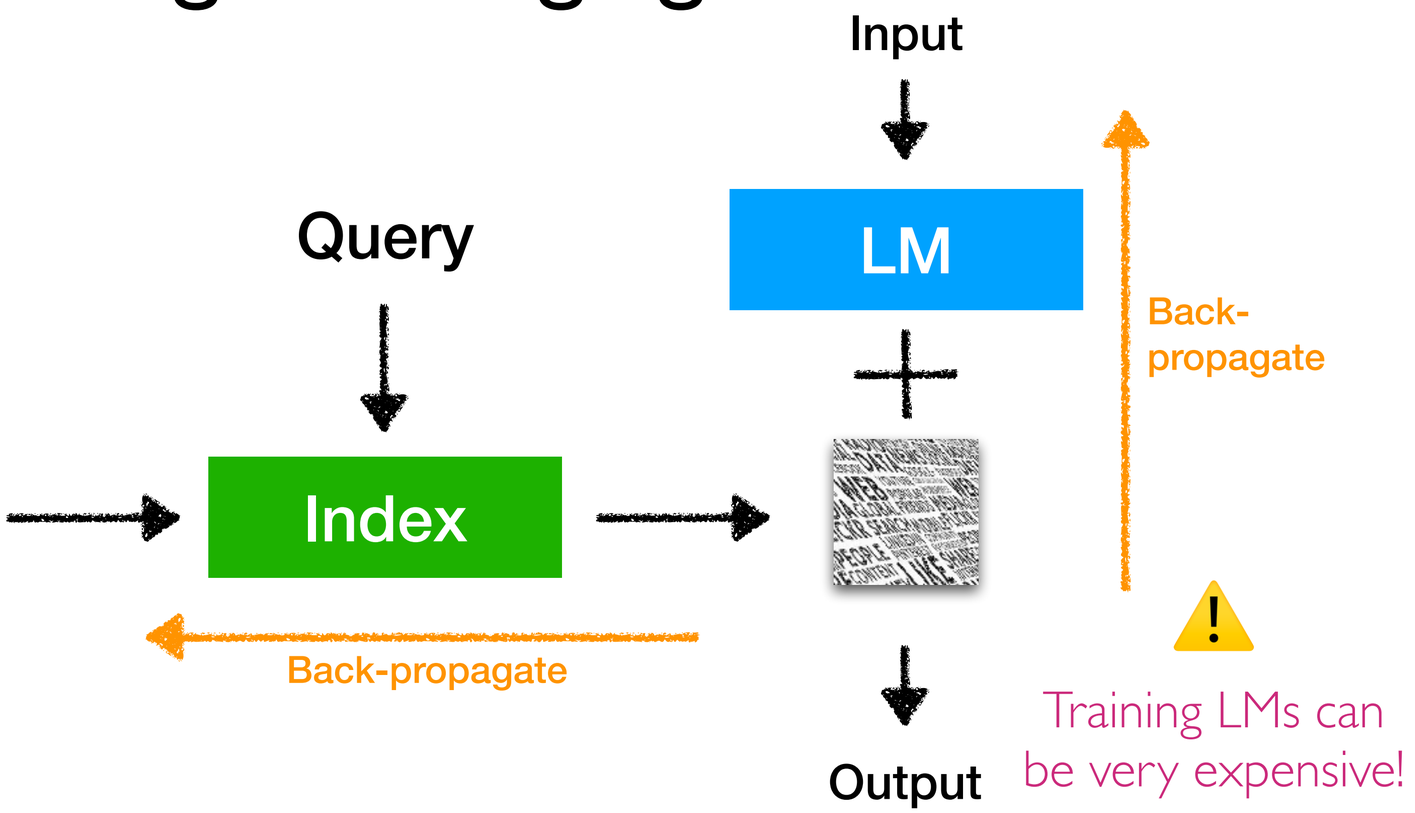
Training retrieval-augmented LMs



Why is training challenging?

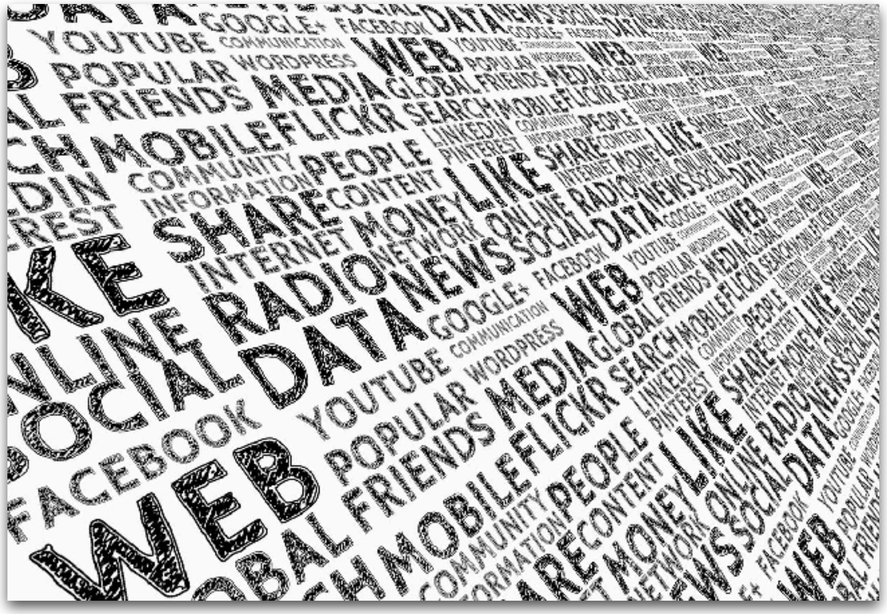


Datastore



Training LMs can be very expensive!

Why is training challenging?



Datastore

Query

Index

Back-propagate

Too large! Expensive to update index during training!

Input

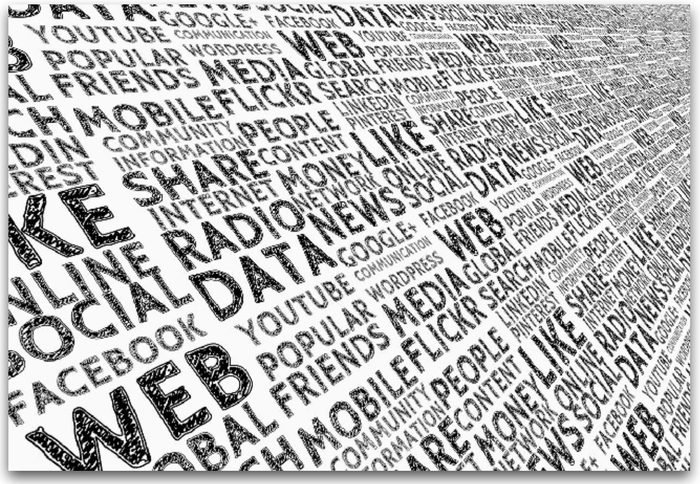
LM

Output

Back-propagate

Training LMs can be very expensive!

Challenges of updating retrieval models



Datastore

$\mathbf{x}_1$

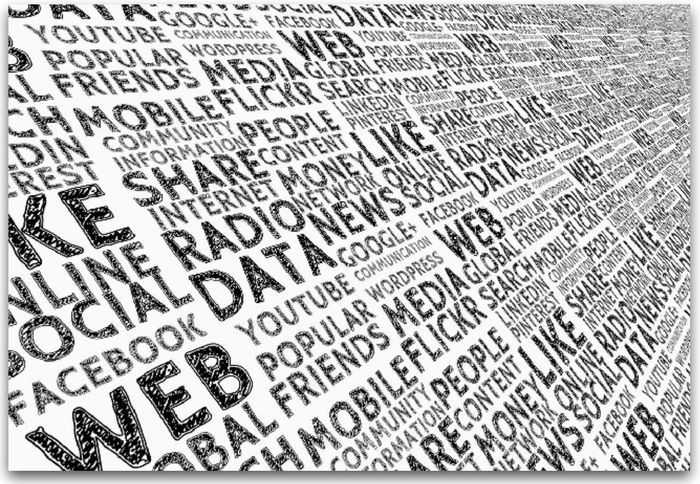
$\mathbf{x}_2$

$\mathbf{x}_3$

$\ldots$

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

Challenges of updating retrieval models



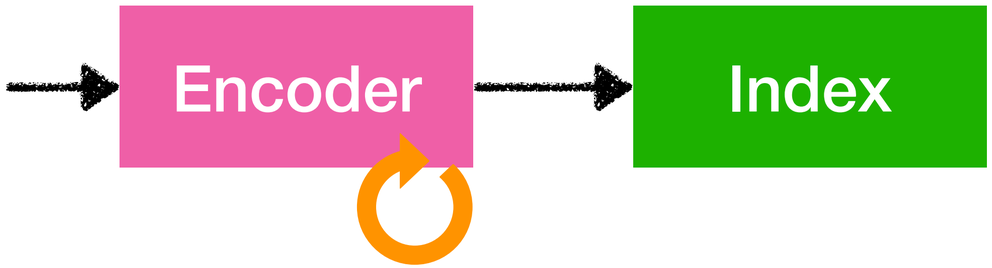
Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

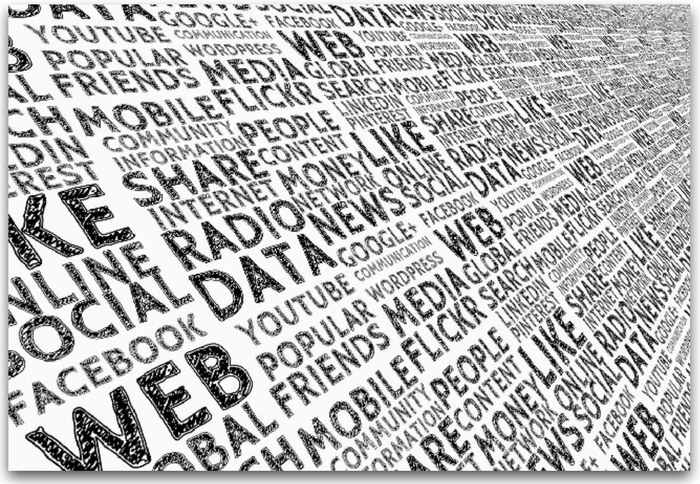
$\mathbf{x}_3$

$\ldots$



During training, we will update the encoder

Challenges of updating retrieval models



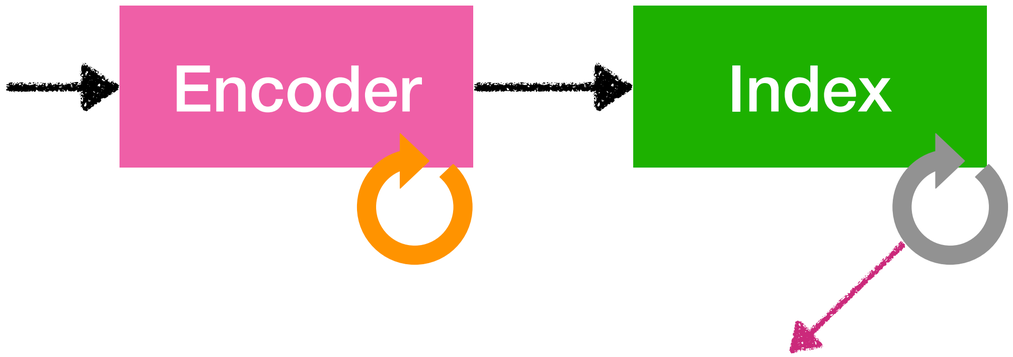
Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_3$

$\ldots$



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

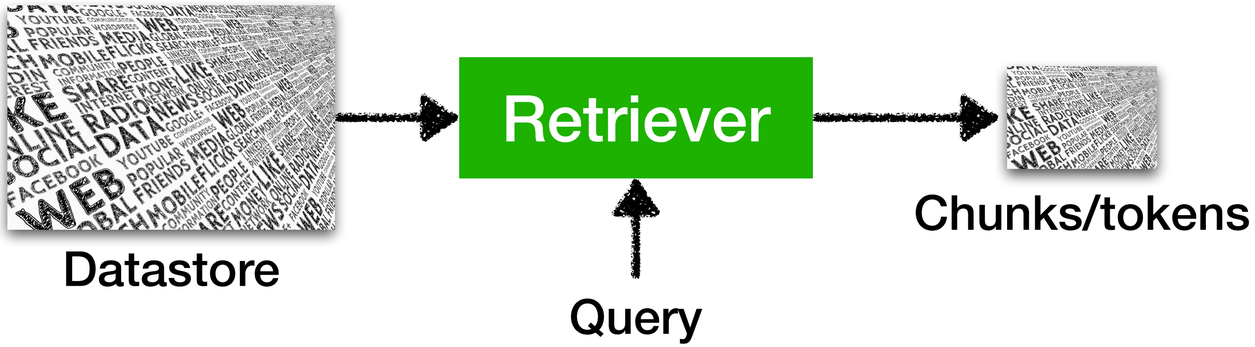
- **Independent training**
- Sequential training
- Joint training w/ asynchronous index update
- Joint training w/ in-batch approximation

Independent training

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

- Training language models

- Training retrieval models

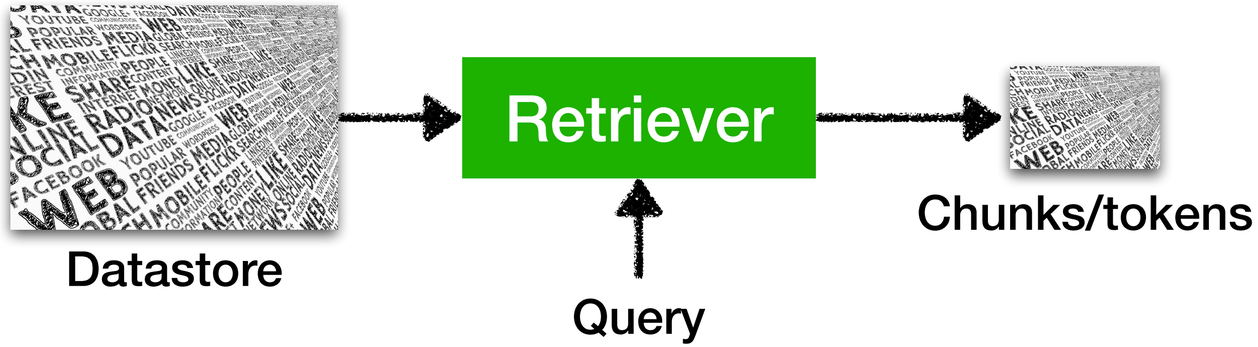


Independent training

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

- Training language models

- Training retrieval models



Sparse retrieval models: TF-IDF / BM25

In 1997, Apple merged with NeXT, and Steve Jobs became CEO of …

Jobs returned to Apple as CEO after the company's acquisition …

Text chunks

→

[0, 0, 0.4, 0, 0.8, 0.7, …]

[0, 1.2, 0.4, 0, 0.8, 0, …]

Sparse vectors

Lexical overlap

Often closed-form (cf. n-gram). No training!

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.

One-Hot Encoding

	time	fruit	flies	like	a	an	arrow	banana
1_{time}	1	0	0	0	0	0	0	0
1_{fruit}	0	1	0	0	0	0	0	0
1_{flies}	0	0	1	0	0	0	0	0
1_{like}	0	0	0	1	0	0	0	0
1_{a}	0	0	0	0	1	0	0	0
1_{an}	0	0	0	0	0	1	0	0
1_{arrow}	0	0	0	0	0	0	1	0
1_{banana}	0	0	0	0	0	0	0	1

Vector Space Model

- Information retrieval model developed by Salton and colleagues in 1968
- Documents and queries are both represented by vectors of term weighs
- Collection thus a matrix of term weights

$$D_i = (d_{i1}, d_{i2}, \ldots, d_{it}) \qquad Q = (q_1, q_2, \ldots, q_t)$$

$$\begin{array}{cccc}
 & Term_1 & Term_2 & \ldots & Term_t \\
Doc_1 & d_{11} & d_{12} & \ldots & d_{1t} \\
Doc_2 & d_{21} & d_{22} & \ldots & d_{2t} \\
\vdots & \vdots & & & \\
Doc_n & d_{n1} & d_{n2} & \ldots & d_{nt}
\end{array}$$

Vector Space Model

D_1 Tropical Freshwater Aquarium Fish.
D_2 Tropical Fish, Aquarium Care, Tank Setup.
D_3 Keeping Tropical Fish and Goldfish in Aquariums, and Fish Bowls.
D_4 The Tropical Tank Homepage - Tropical Fish and Aquariums.

Terms	**Documents**			
	D_1	**D_2**	**D_3**	**D_4**
aquarium	1	1	1	1
bowl	0	0	1	0
care	0	1	0	0
fish	1	1	2	1
freshwater	1	0	0	0
goldfish	0	0	1	0
homepage	0	0	0	1
keep	0	0	1	0
setup	0	1	0	0
tank	0	1	0	1
tropical	1	1	1	2

Vector Space Model

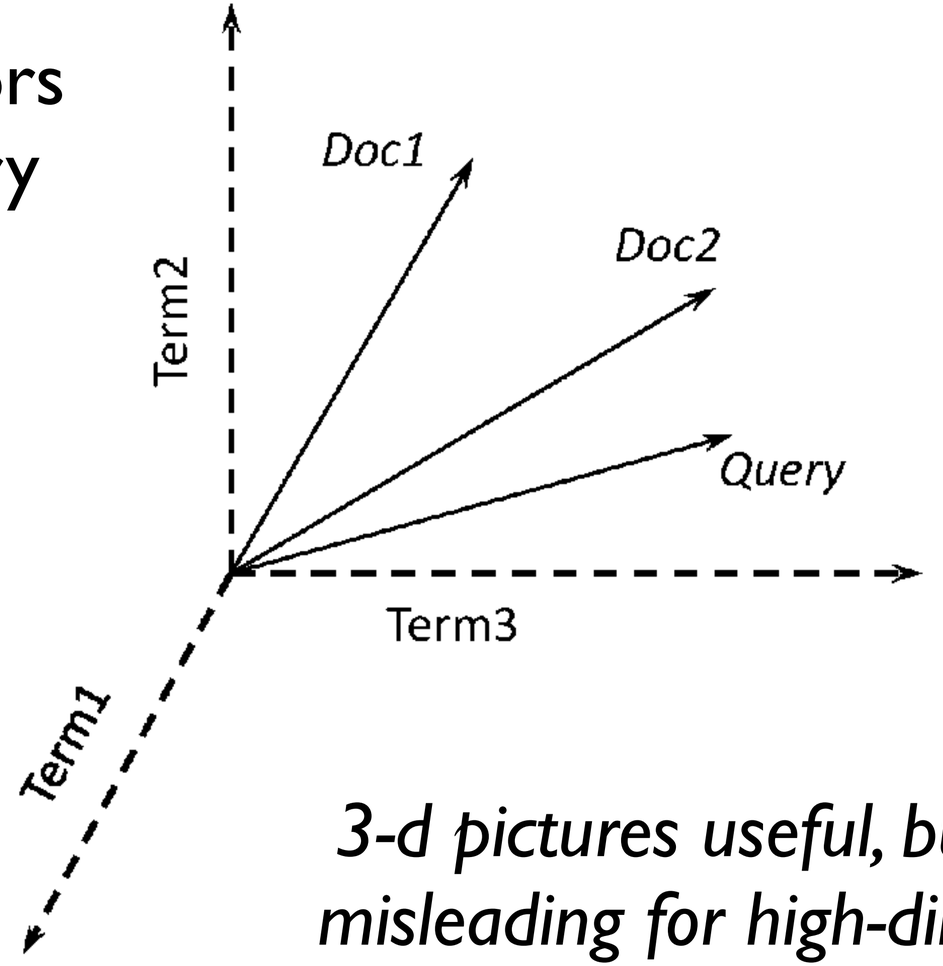
Query: tropical fish

Term	Query
aquarium	0
bowl	0
care	0
fish	**1**
freshwater	0
goldfish	0
homepage	0
keep	0
setup	0
tank	0
tropical	**1**

Usually much sparser than a document!

Vector Space Model

Retrieve vectors near the query



3-d pictures useful, but can be misleading for high-dimensional space

Vector Space Model

- Documents ranked by distance between points representing query and documents
  - *Similarity* measure more common than a distance or *dissimilarity* measure
  - e.g. Cosine correlation

$$Cosine(D_i, Q) = \frac{\sum_{j=1}^{t} d_{ij} \cdot q_j}{\sqrt{\sum_{j=1}^{t} {d_{ij}}^2 \cdot \sum_{j=1}^{t} {q_j}^2}}$$

  - Note speedup when query is sparse

Term Weights

- Often *tf.idf* weights (Spärck-Jones, 1973)
- Term frequency weight, normalized by all terms j in document i, measures importance in document: $tf_{ik} = \frac{f_{ik}}{\sum_{j=1}^{t} f_{ij}}$
- Inverse document frequency measures importance in collection: $idf_k = \log \frac{N}{n_k}$
- Heuristic combination (note add-1 smoothing to avoid log 0)
- $$d_{ik} = \frac{\log(f_{ik} + 1) \cdot \log(\frac{N}{n_k})}{\sqrt{\left[\sum_{j=1}^{t} \log(f_{ij} + 1) \cdot \log(\frac{N}{n_j})\right]^2}}$$

BM25

- Popular and effective ranking algorithm based on binary independence model (cf. naive Bayes)
  - adds document and query term weights

$$\sum_{i \in Q} \log \frac{(r_i+0.5)/(R-r_i+0.5)}{(n_i-r_i+0.5)/(N-n_i-R+r_i+0.5)} \cdot \frac{(k_1+1)f_i}{K+f_i} \cdot \frac{(k_2+1)qf_i}{k_2+qf_i}$$

  - k_1, k_2 and K are parameters whose values are set empirically; R, r for user feedback
  - $K = k_1((1-b) + b \cdot \frac{dl}{avdl})$ *dl* is doc length
  - Typical TREC value for k_1 is 1.2, k_2 varies from 0 to 1000, b = 0.75

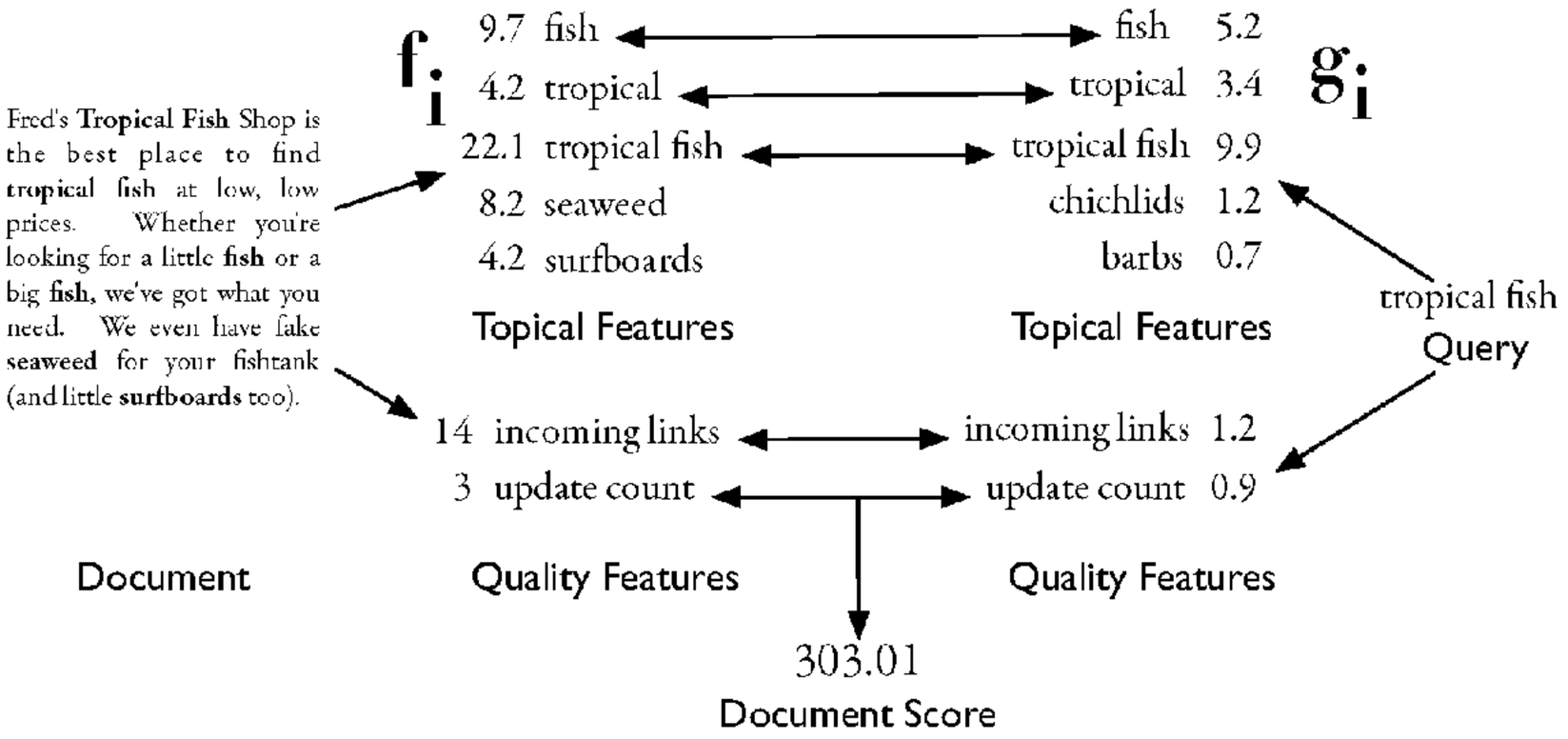
A Shakespeare Concordance

IRISHMAN—an Irishman with.... *Merry Wives*, ii. 2
altogether directed by an Irishman .. *Henry V*. iii. 2
IRISHMEN-against the Irishmen? .2 *Henry VI*. iii. 1
IRK—and yet it irks me..... *As you Like it*, ii. 1
it irks his heart, he cannot ..... 1 *Henry VI*. i. 4
it irks my very soul .....3 *Henry VI*. ii. 2
IRKSOME—was irksome to me .. *As you Like it*, iii. 5
is an irksome brawling scold .. *Taming of Shrew*, i. 2
irksome is this music to my heart!..2 *Henry VI*. ii. 1
IRON—to wear iron about you.... *Twelfth Night*, iii. 4
my young soldier, put up your iron.. — iv. 1
before barred up with ribs of iron! .. *Much Ado*, iv. 1
runs not this speech like iron through — v. 1
but yet you draw not iron..... *Mid. N.'s Dream*, ii. 2
the iron tongue of midnight hath .... — v. 1
iron may hold with her ..... *Taming of Shrew*, ii. 1
fetch me an iron crow..... *Comedy of Errors*, iii. 1
their iron indignation 'gainst your.. *King John*, ii. 1
with his iron tongue and brazen mouth — iii. 3
heat me these irons hot ..... — iv. 1
must you with hot irons burn (*rep.*).. — iv. 1
none, but in this iron age ..... — iv. 1
stubborn hard than hammered iron? — iv. 1

A Model of Ranking

$$R(Q, D) = \sum_i g_i(Q) f_i(D)$$

f_i is a document feature function
g_i is a query feature function



Example "Collection"

S_1 Tropical fish include fish found in tropical environments around the world, including both freshwater and salt water species.

S_2 Fishkeepers often use the term tropical fish to refer only those requiring fresh water, with saltwater tropical fish referred to as marine fish.

S_3 Tropical fish are popular aquarium fish, due to their often bright coloration.

S_4 In freshwater fish, this coloration typically derives from iridescence, while salt water fish are generally pigmented.

Four sentences from the Wikipedia entry for *tropical fish*

Simple Inverted Index

and	1
aquarium	3
are	3 4
around	1
as	2
both	1
bright	3
coloration	3 4
derives	4
due	3
environments	1
fish	1 2 3 4
fishkeepers	2
found	1
fresh	2
freshwater	1 4
from	4
generally	4
in	1 4
include	1
including	1
iridescence	4
marine	2
often	2 3
only	2
pigmented	4
popular	3
refer	2
referred	2
requiring	2
salt	1 4
saltwater	2
species	1
term	2
the	1 2
their	3
this	4
those	2
to	2 3
tropical	1 2 3
typically	4
use	2
water	1 2 4
while	4
with	2
world	1

Inverted Index with counts

- supports better ranking algorithms

and	1:1			
aquarium	3:1			
are	3:1	4:1		
around	1:1			
as	2:1			
both	1:1			
bright	3:1			
coloration	3:1	4:1		
derives	4:1			
due	3:1			
environments	1:1			
fish	1:2	2:3	3:2	4:2
fishkeepers	2:1			
found	1:1			
fresh	2:1			
freshwater	1:1	4:1		
from	4:1			
generally	4:1			
in	1:1	4:1		
include	1:1			
including	1:1			
iridescence	4:1			
marine	2:1			
often	2:1	3:1		

only	2:1		
pigmented	4:1		
popular	3:1		
refer	2:1		
referred	2:1		
requiring	2:1		
salt	1:1	4:1	
saltwater	2:1		
species	1:1		
term	2:1		
the	1:1	2:1	
their	3:1		
this	4:1		
those	2:1		
to	2:2	3:1	
tropical	1:2	2:2	3:1
typically	4:1		
use	2:1		
water	1:1	2:1	4:1
while	4:1		
with	2:1		
world	1:1		

Inverted Index with positions

- supports proximity matches

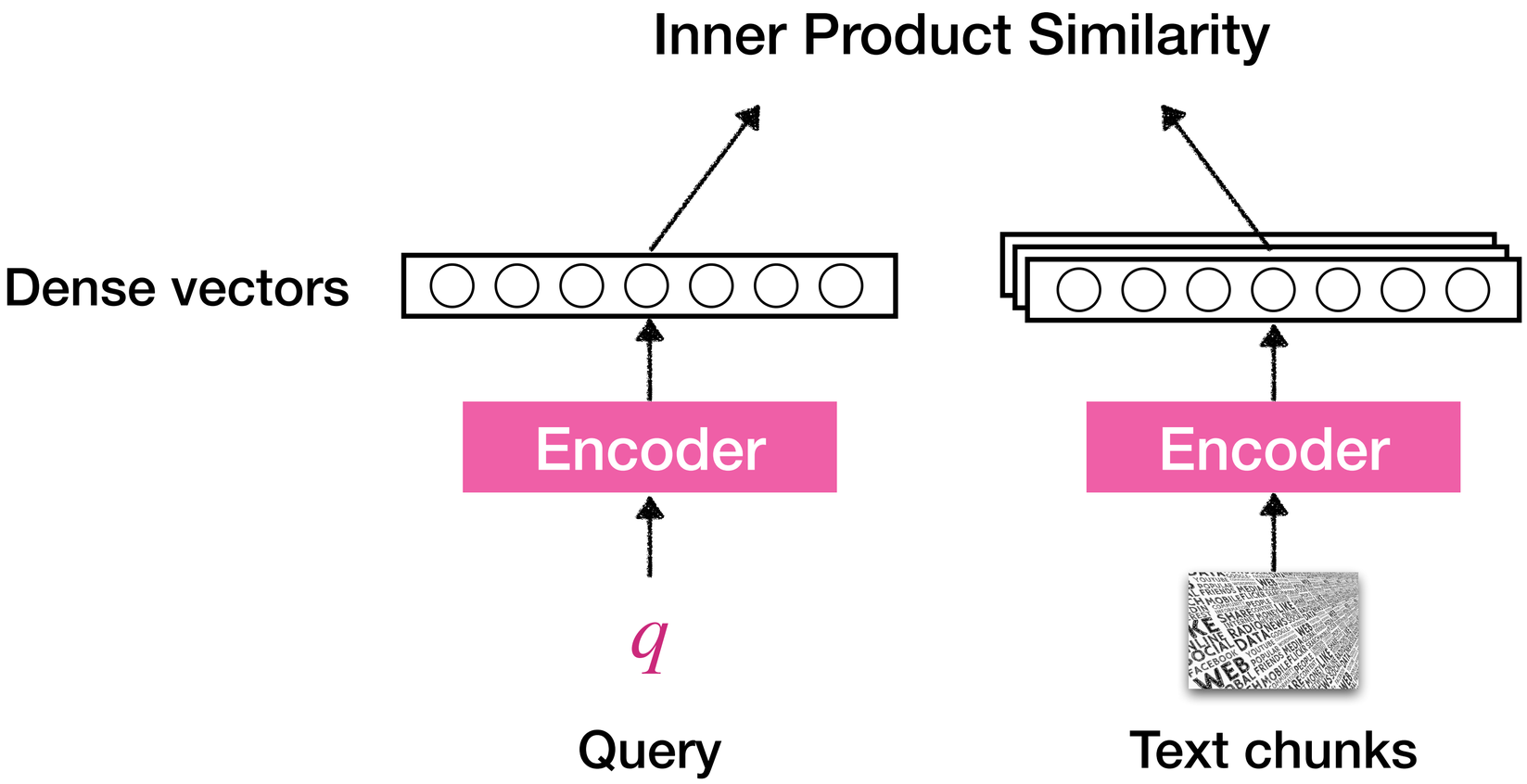
and	1,15				
aquarium	3,5				
are	3,3	4,14			
around	1,9				
as	2,21				
both	1,13				
bright	3,11				
coloration	3,12	4,5			
derives	4,7				
due	3,7				
environments	1,8				
fish	1,2	1,4	2,7	2,18	2,23
			3,2	3,6	4,3
			4,13		
fishkeepers	2,1				
found	1,5				
fresh	2,13				
freshwater	1,14	4,2			
from	4,8				
generally	4,15				
in	1,6	4,1			
include	1,3				
including	1,12				
iridescence	4,9				

marine	2,22				
often	2,2	3,10			
only	2,10				
pigmented	4,16				
popular	3,4				
refer	2,9				
referred	2,19				
requiring	2,12				
salt	1,16	4,11			
saltwater	2,16				
species	1,18				
term	2,5				
the	1,10	2,4			
their	3,9				
this	4,4				
those	2,11				
to	2,8	2,20	3,8		
tropical	1,1	1,7	2,6	2,17	3,1
typically	4,6				
use	2,3				
water	1,17	2,14	4,12		
while	4,10				
with	2,15				
world	1,11				

Inverted Index

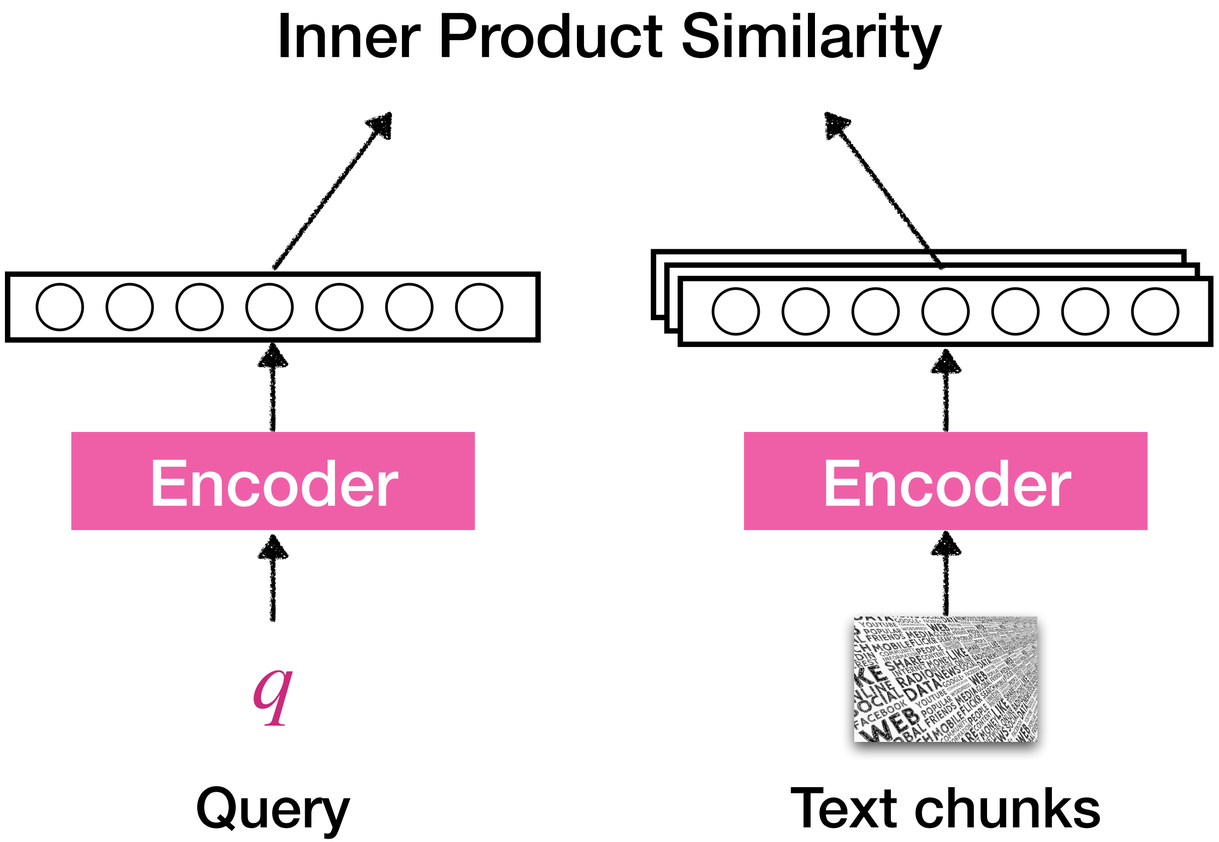
- Each index term (i.e., feature) is associated with an *inverted list*
- Each entry in the inverted list is a *posting*
- Each posting contains a document ID and possibly feature weights and positions
- How do we define a document?
- How to order the postings?
- When is this faster than a linear scan over all documents?
  - When features are **sparse**, i.e., the expected number of non-zero components in the dot product is small
- This and more in Information Retrieval course

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



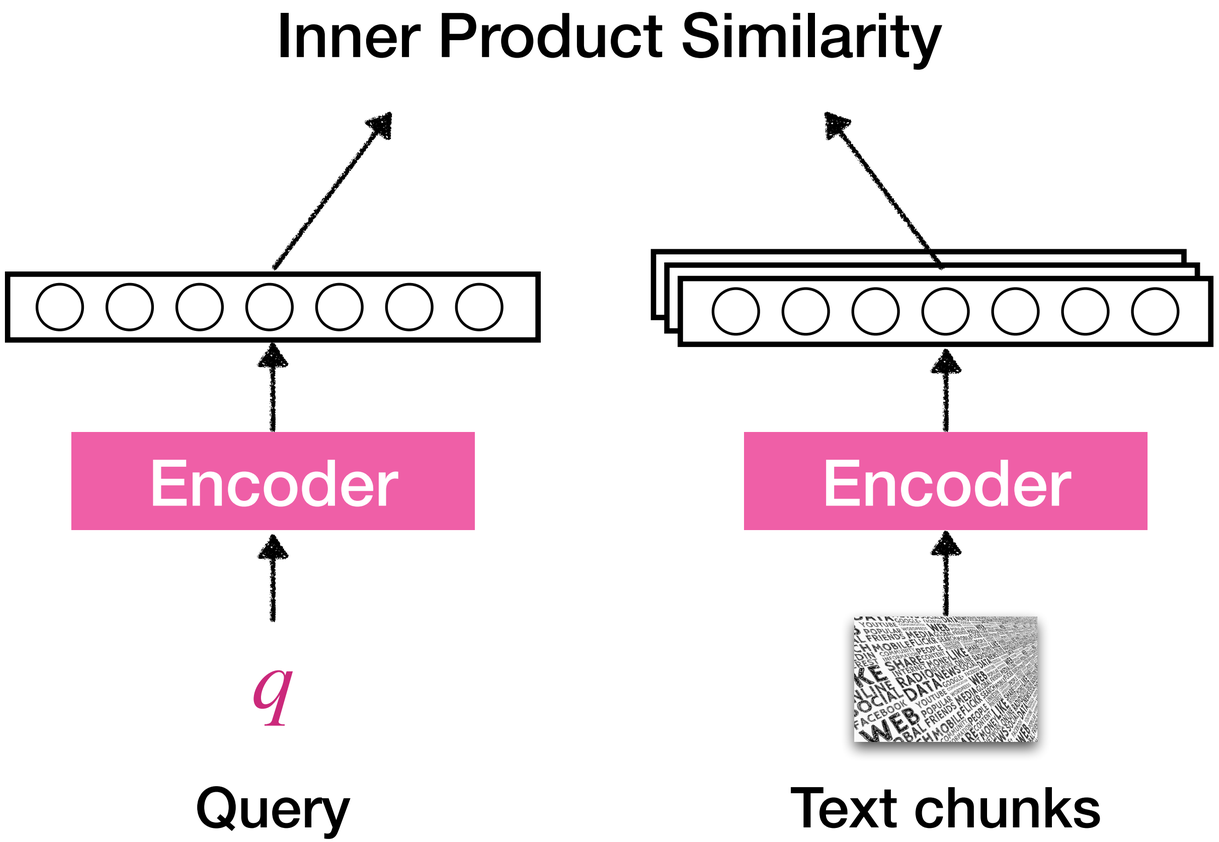
Karpukhin et al. Dense Passage Retrieval for Open-Domain Question Answering. EMNLP 2020.

Training dense retrieval models: DPR



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

Training dense retrieval models: DPR

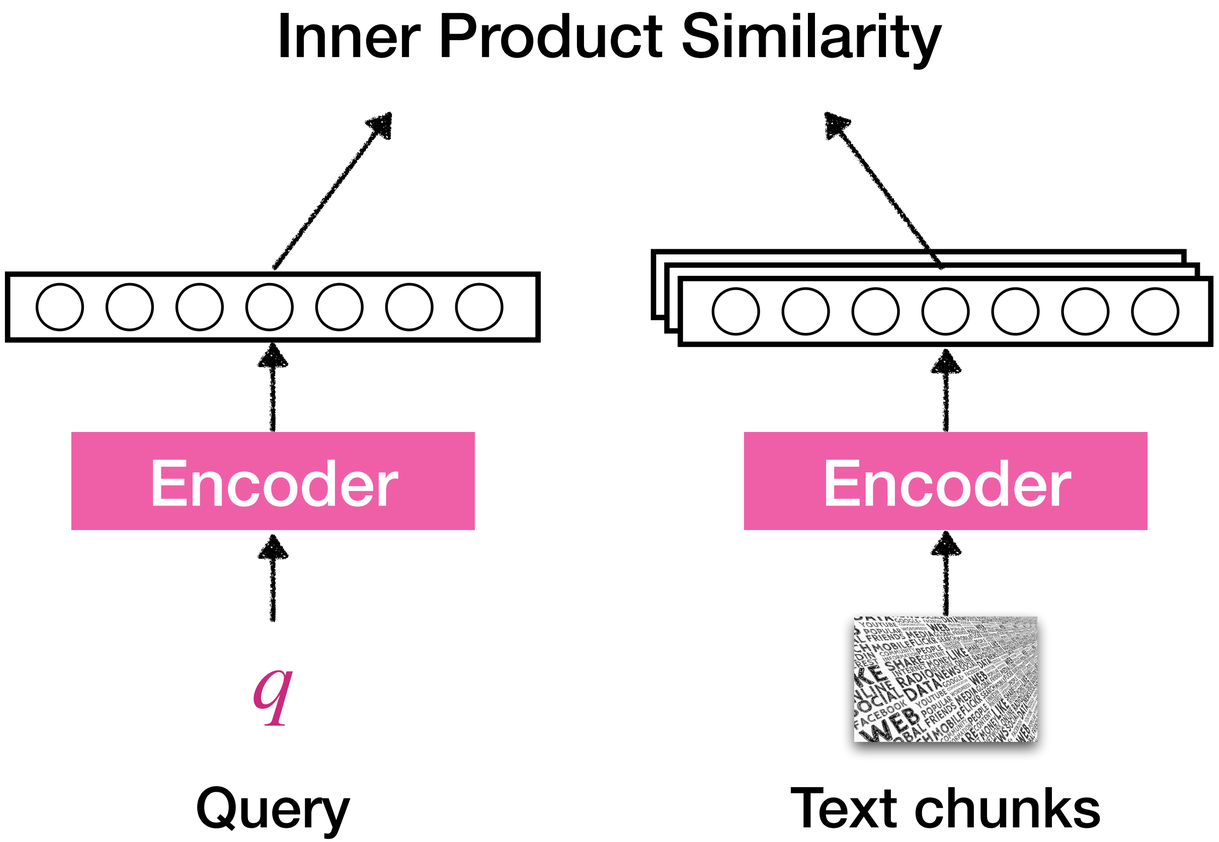


$$L(q, p^+, p_1^-, p_2^-, \ldots, p_n^-)$$

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

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

Training dense retrieval models: DPR



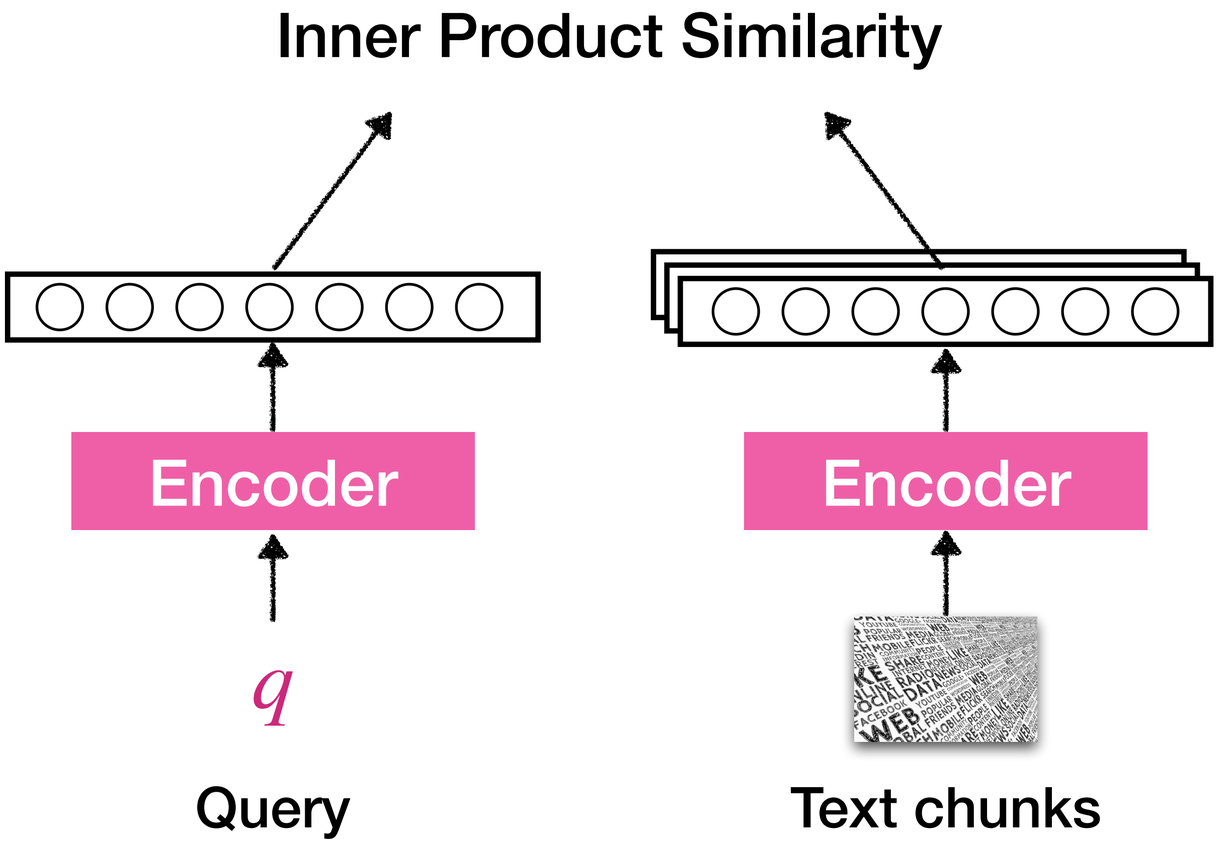
$$L(q, p^+, p_1^-, p_2^-, \ldots, p_n^-)$$

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

Contrastive learning

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

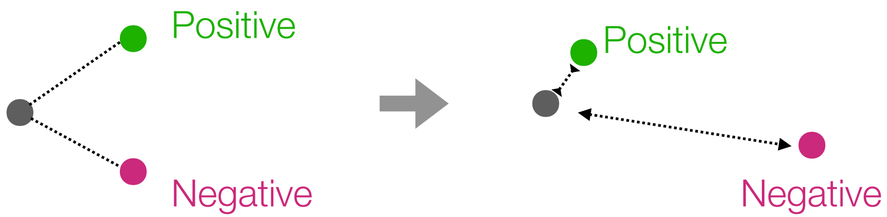
Training dense retrieval models: DPR



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

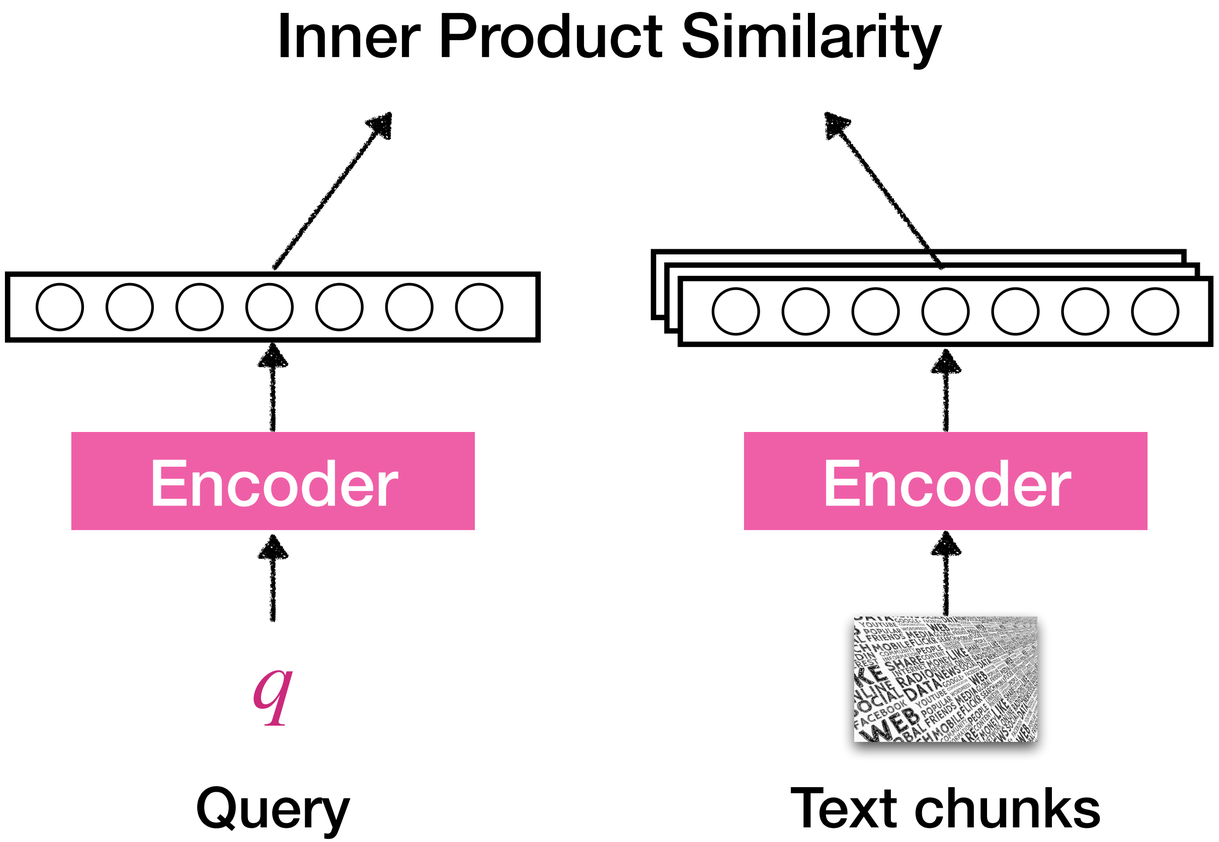
$$= -\log \frac{\exp(\mathrm{sim}(q, p^+))}{\exp(\mathrm{sim}(q, p^+)) + \sum_{j=1}^{n} \exp(\mathrm{sim}(q, p_j^-))}$$

Contrastive learning



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

Training dense retrieval models: DPR



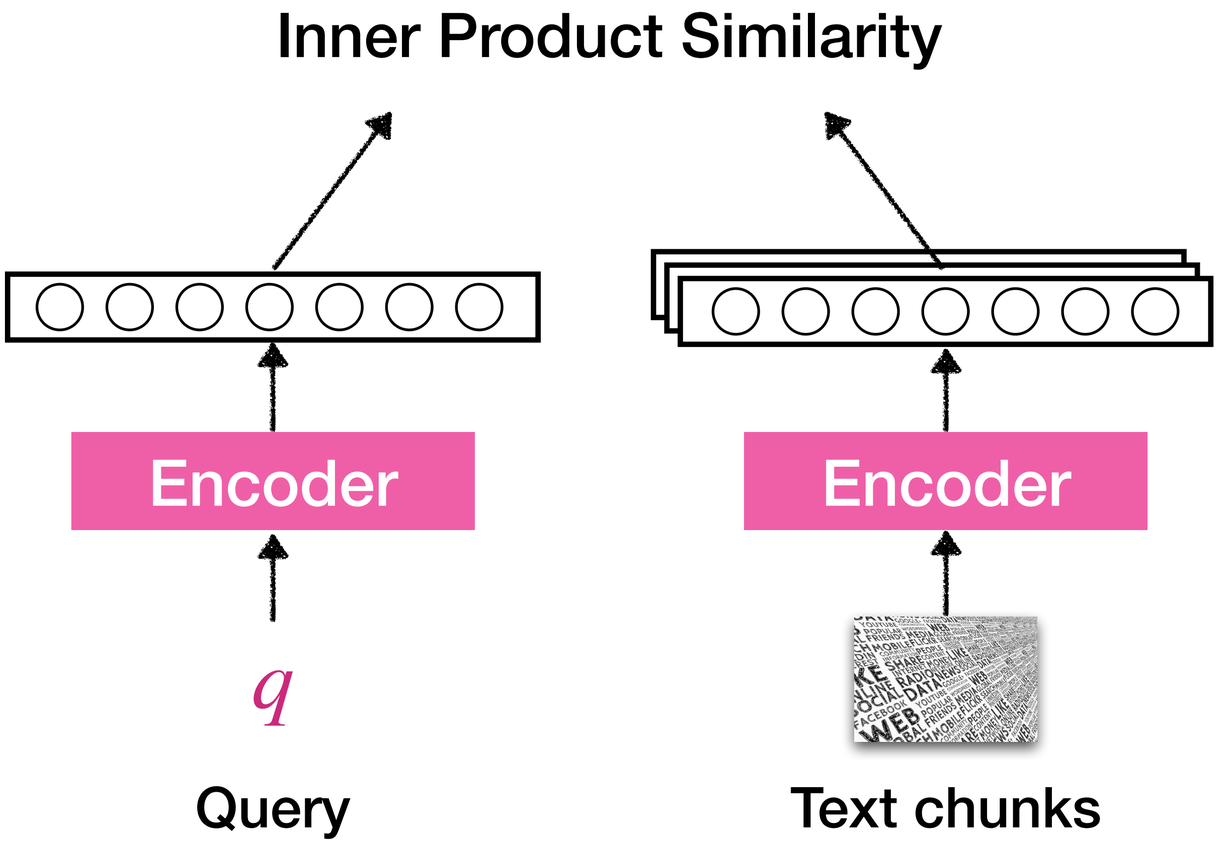
$$L(q, p^+, p_1^-, p_2^-, \ldots, p_n^-)$$

Positive passage

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

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!

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

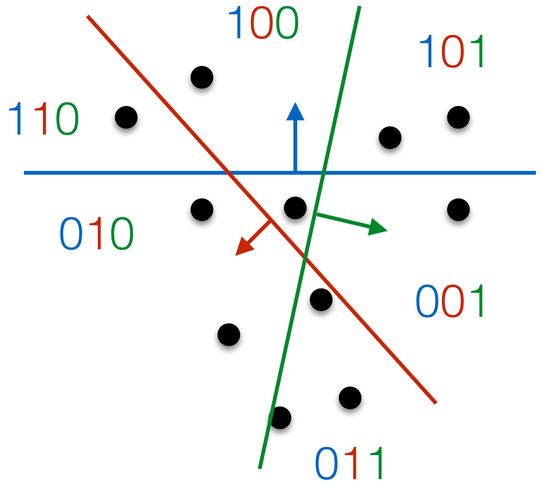
Positive passage

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

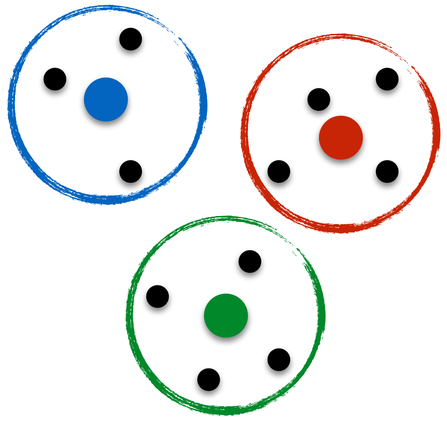
Approximate Nearest-Neighbor Search

- Methods to retrieve embeddings in sub-linear time

Locality sensitive hashing: make partitions in continuous space, use like inverted index



Graph-based search: create "hubs" and search from there



- Software: FAISS, ChromaDB

LSH for Cosine Similarity: Simhash

1. Process the document into a set of features with associated weights. We will assume the simple case where the features are words weighted by their frequency.

2. Generate a hash value with b bits (the desired size of the fingerprint) for each word. The hash value should be unique for each word.

3. In b-dimensional vector V, update the components of the vector by adding the weight for a word to every component for which the corresponding bit in the word's hash value is 1, and subtracting the weight if the value is 0.

4. After all words have been processed, generate a b-bit fingerprint by setting the ith bit to 1 if the ith component of V is positive, or 0 otherwise.

LSH for Cosine Similarity: Simhash

Tropical fish include fish found in tropical environments around the world, including both freshwater and salt water species.

(a) Original text

tropical 2 fish 2 include 1 found 1 environments 1 around 1 world 1
including 1 both 1 freshwater 1 salt 1 water 1 species 1

(b) Words with weights

tropical	01100001	fish	10101011	include	11100110
found	00011110	environments	00101101	around	10001011
world	00101010	including	11000000	both	10101110
freshwater	00111111	salt	10110101	water	00100101
species	11101110				

(c) 8 bit hash values

1 -5 9 -9 3 1 3 3

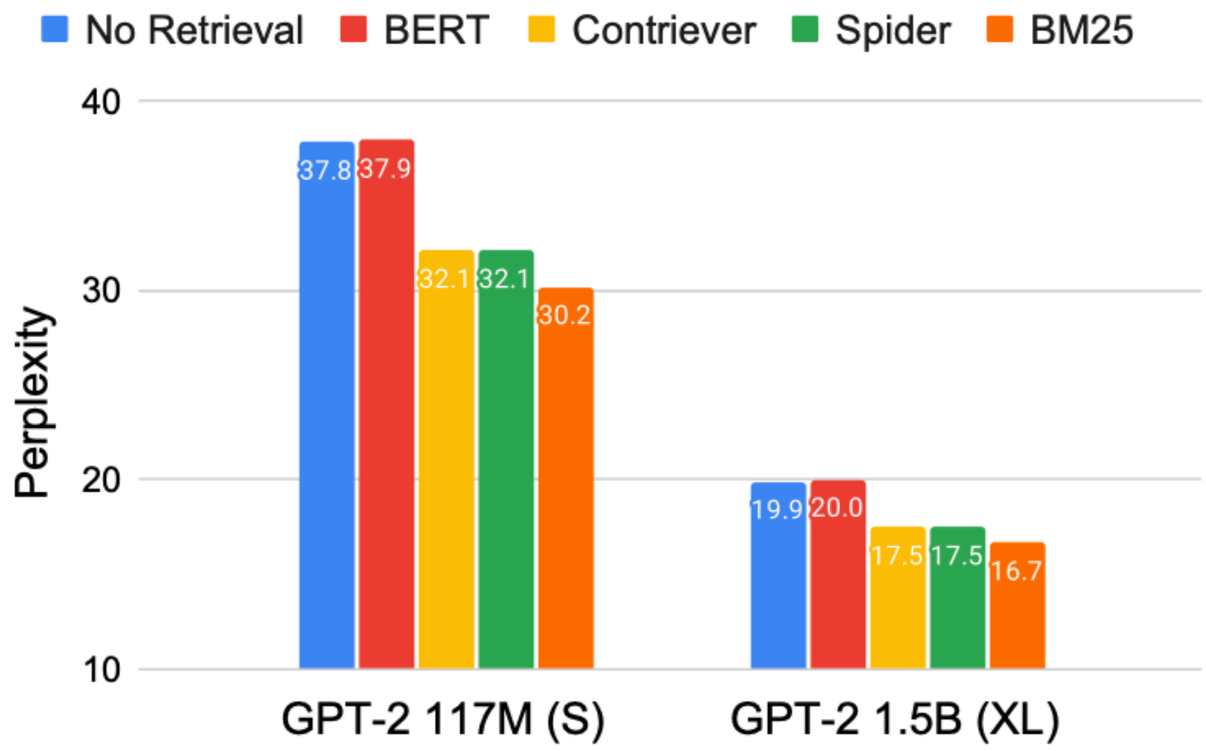
(d) Vector V formed by summing weights

1 0 1 0 1 1 1 1

(e) 8-bit fingerprint formed from V

One-sided (false negative) error. Check multiple hash functions to drive down error

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

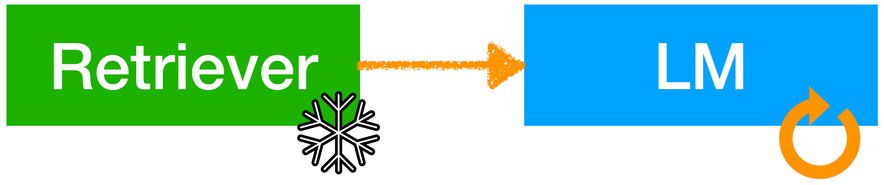
- Independent training
- **Sequential training**
- Joint training w/ asynchronous index update
- Joint training w/ in-batch approximation

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

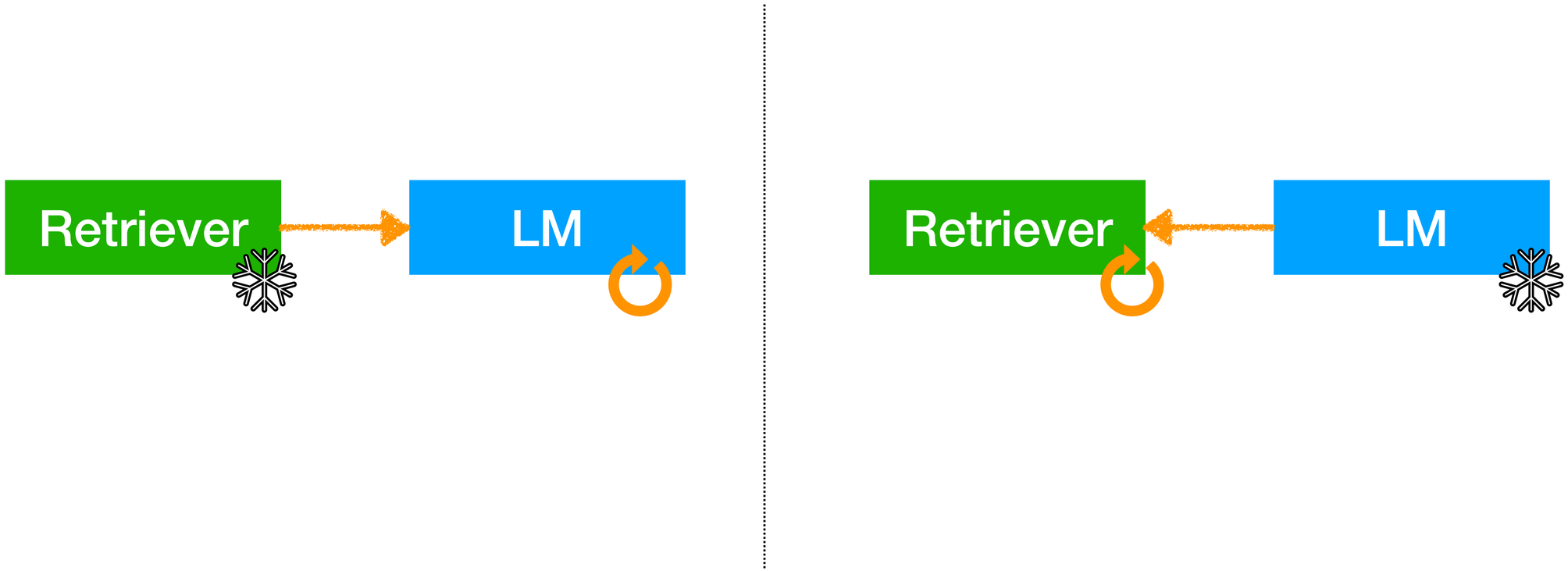
Sequential training

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



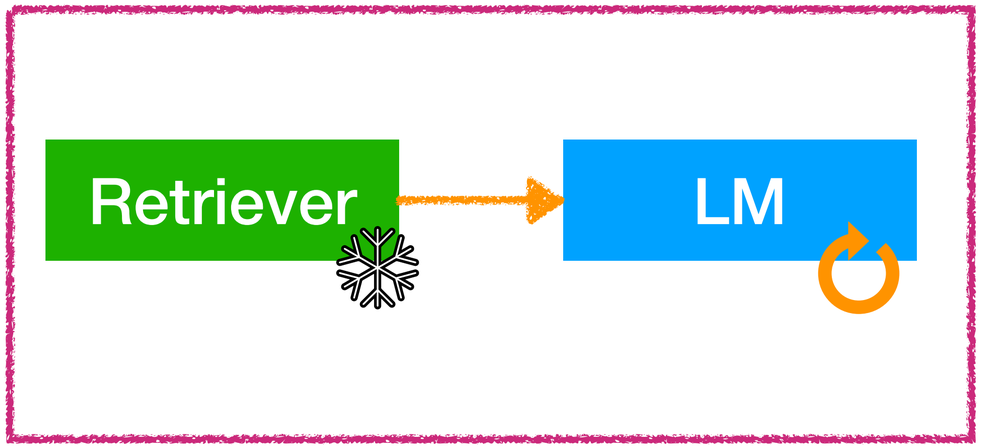
Sequential training

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

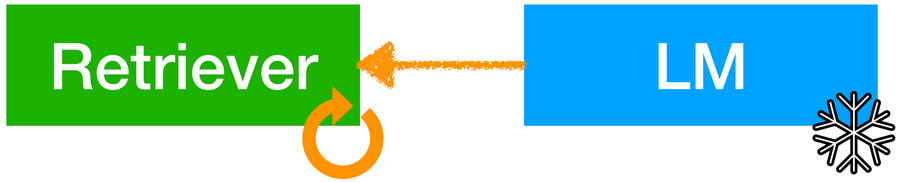


Sequential training

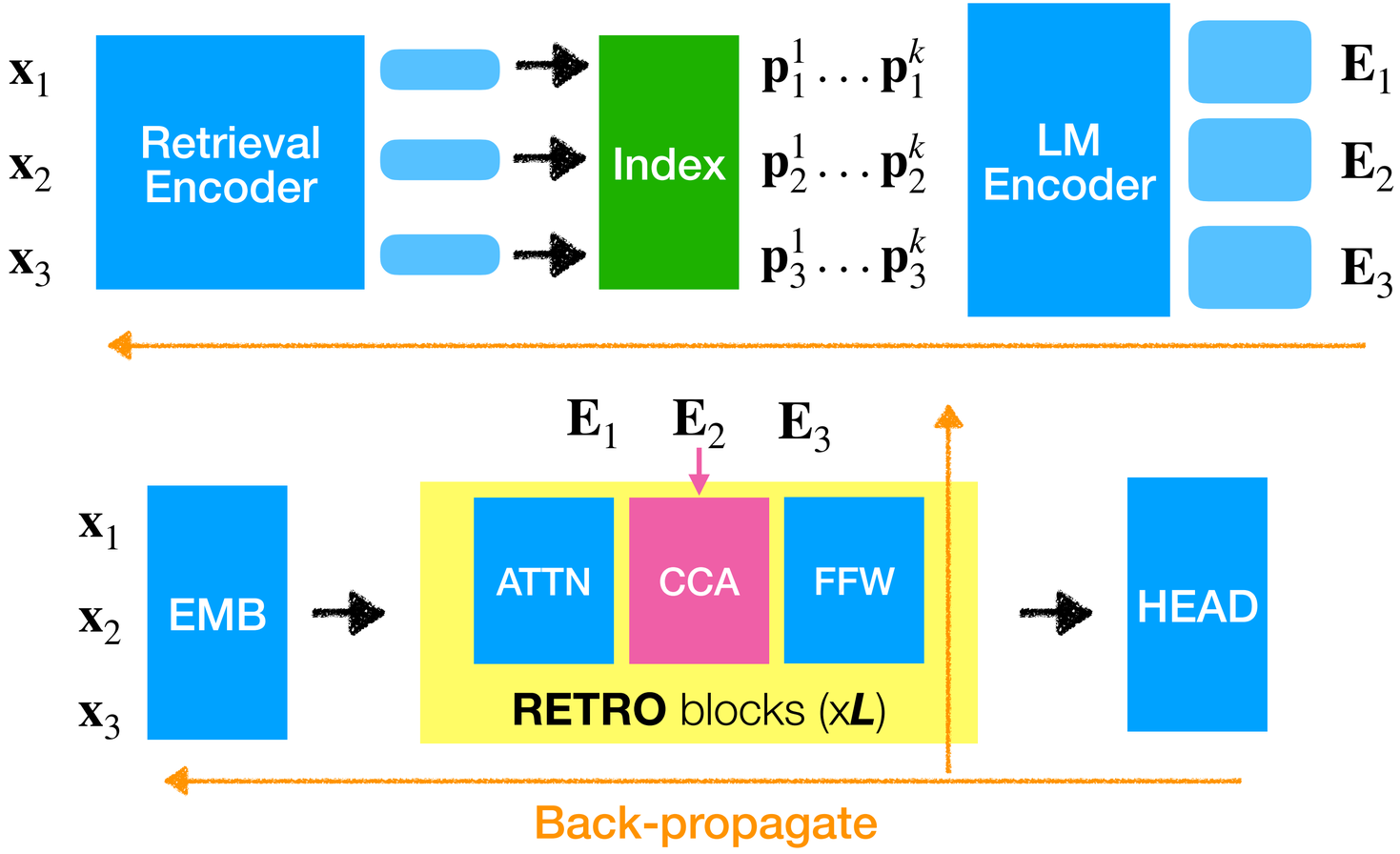
- One component is first trained independently and then fixed
- The other component is trained with an objective that depends on the first one



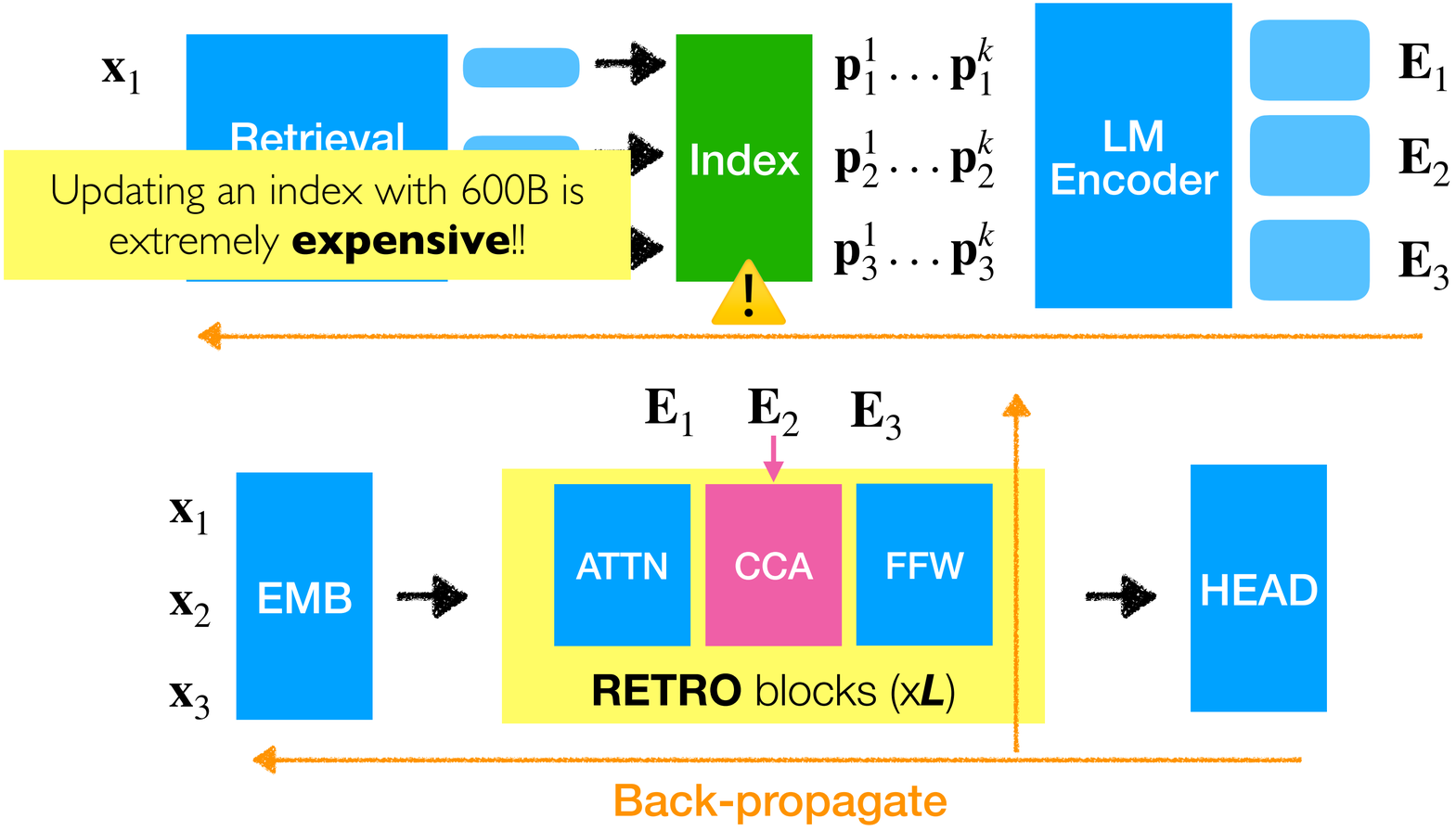
e.g., RETRO; WebGPT



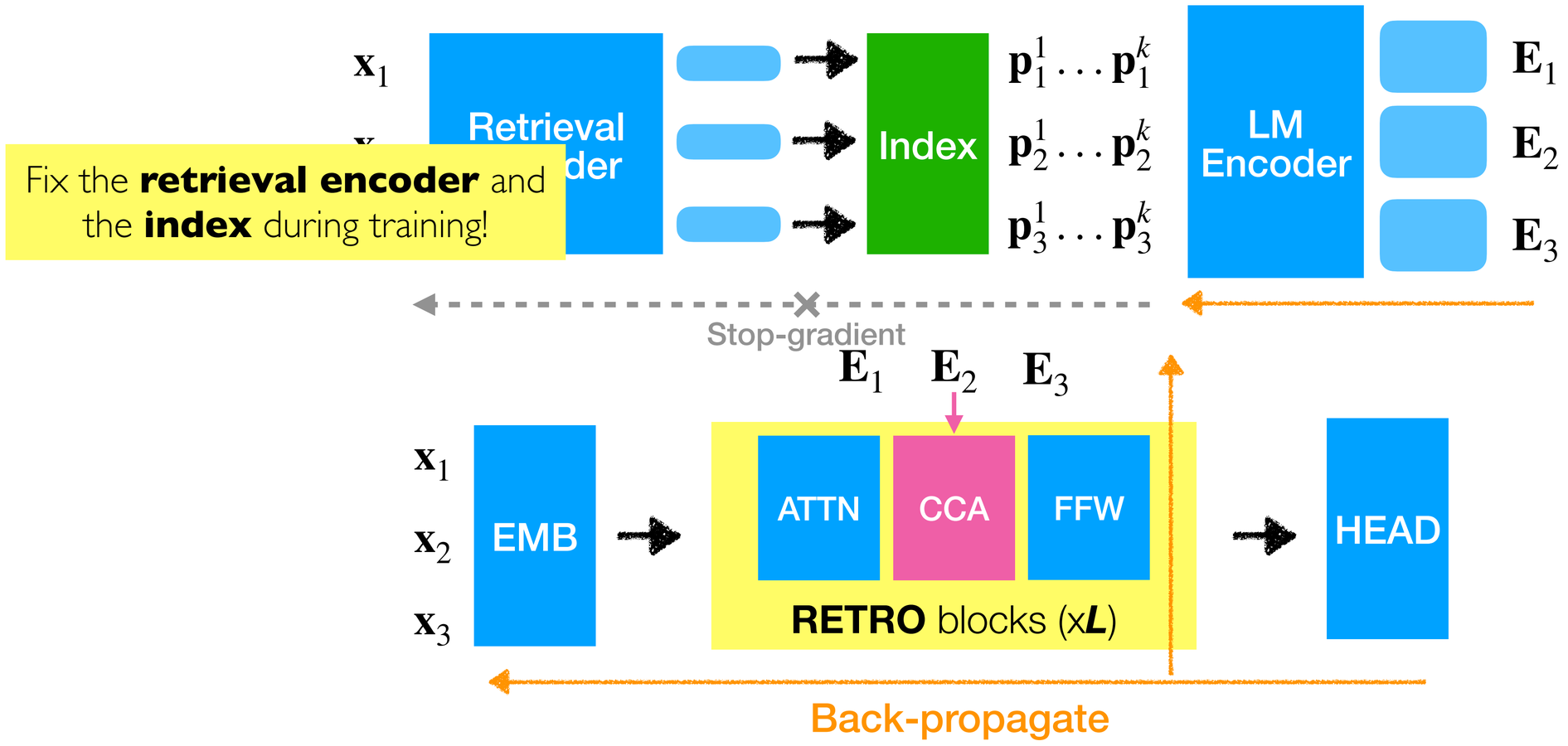
RETRO: Training



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

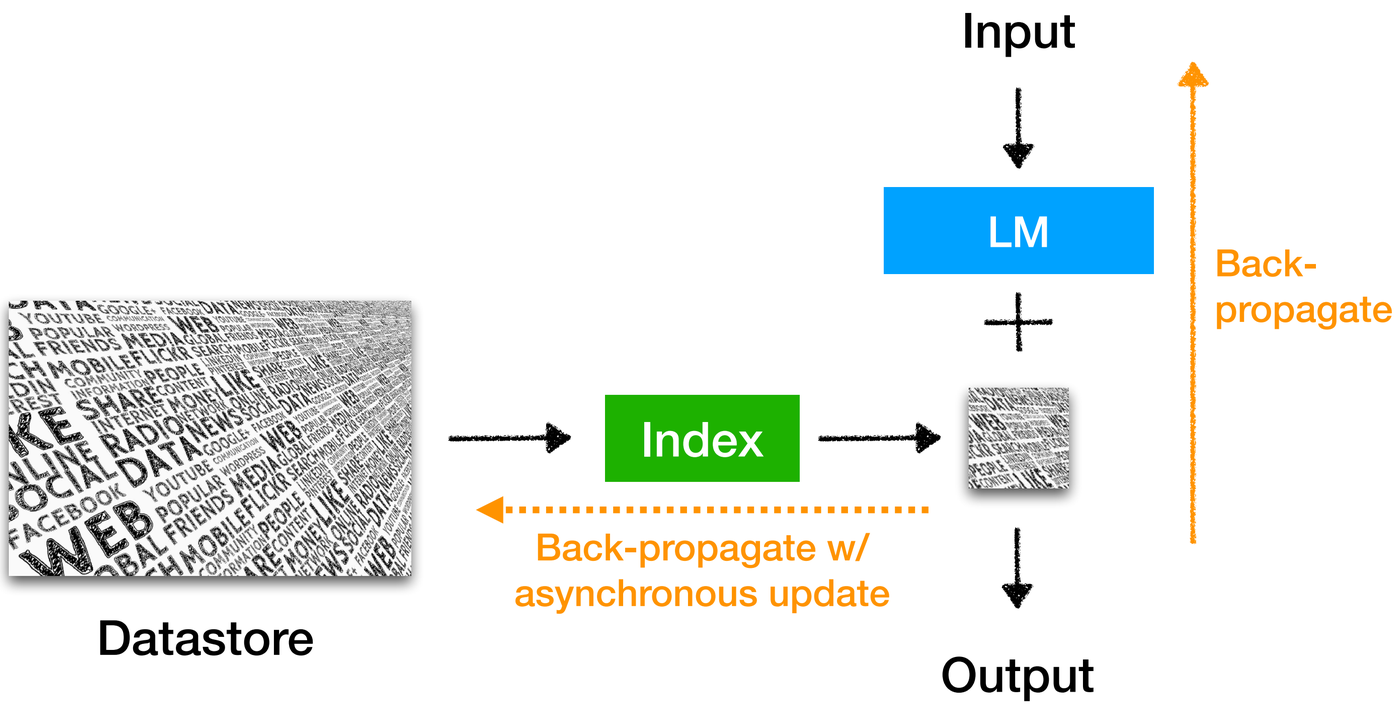
- Independent training
- Sequential training
- **Joint training w/ asynchronous index update**
- **Joint training w/ in-batch approximation**

Training methods for retrieval-augmented LMs

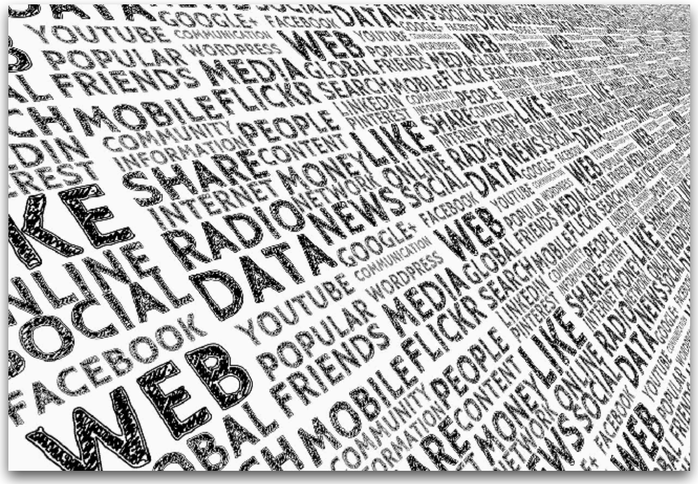
- Independent training
- Sequential training
- **Joint training w/ asynchronous index update**
- **Joint training w/ in-batch approximation**

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

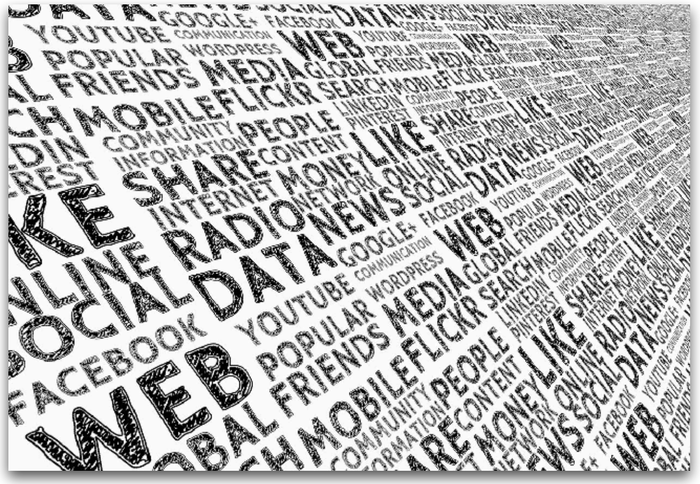
$\mathbf{x}_1$

$\mathbf{x}_2$

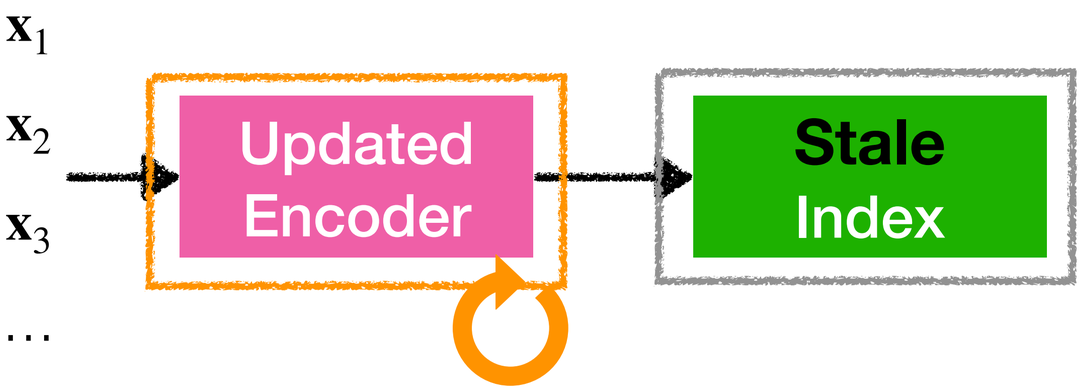
$\mathbf{x}_3$

$\ldots$

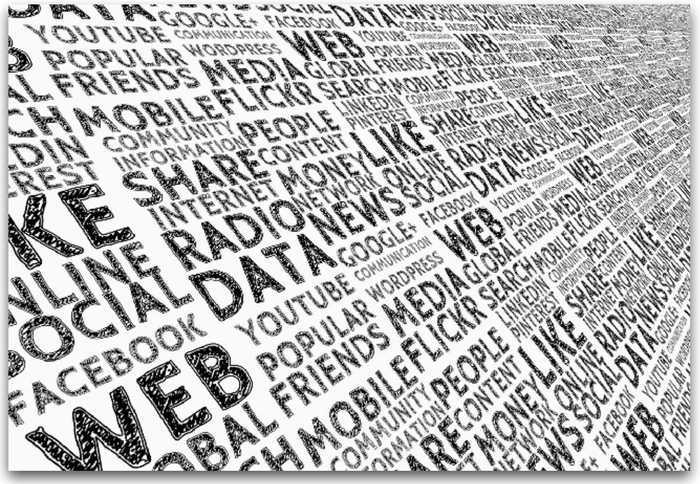
Asynchronous index update



Datastore



Asynchronous index update



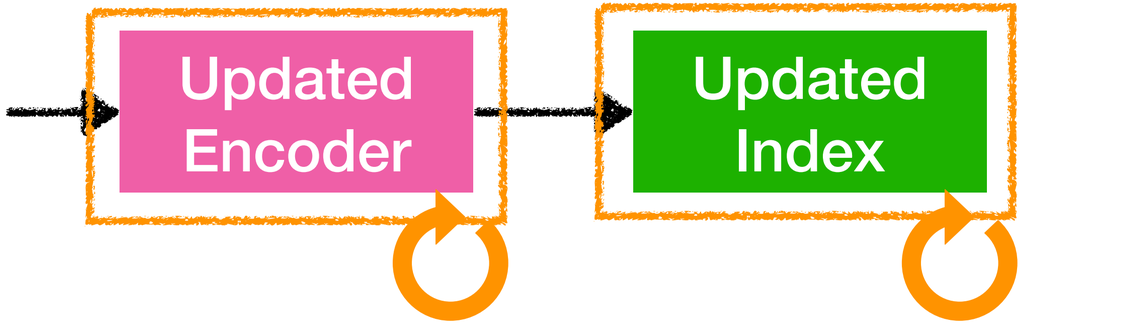
Datastore

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_3$

$\ldots$

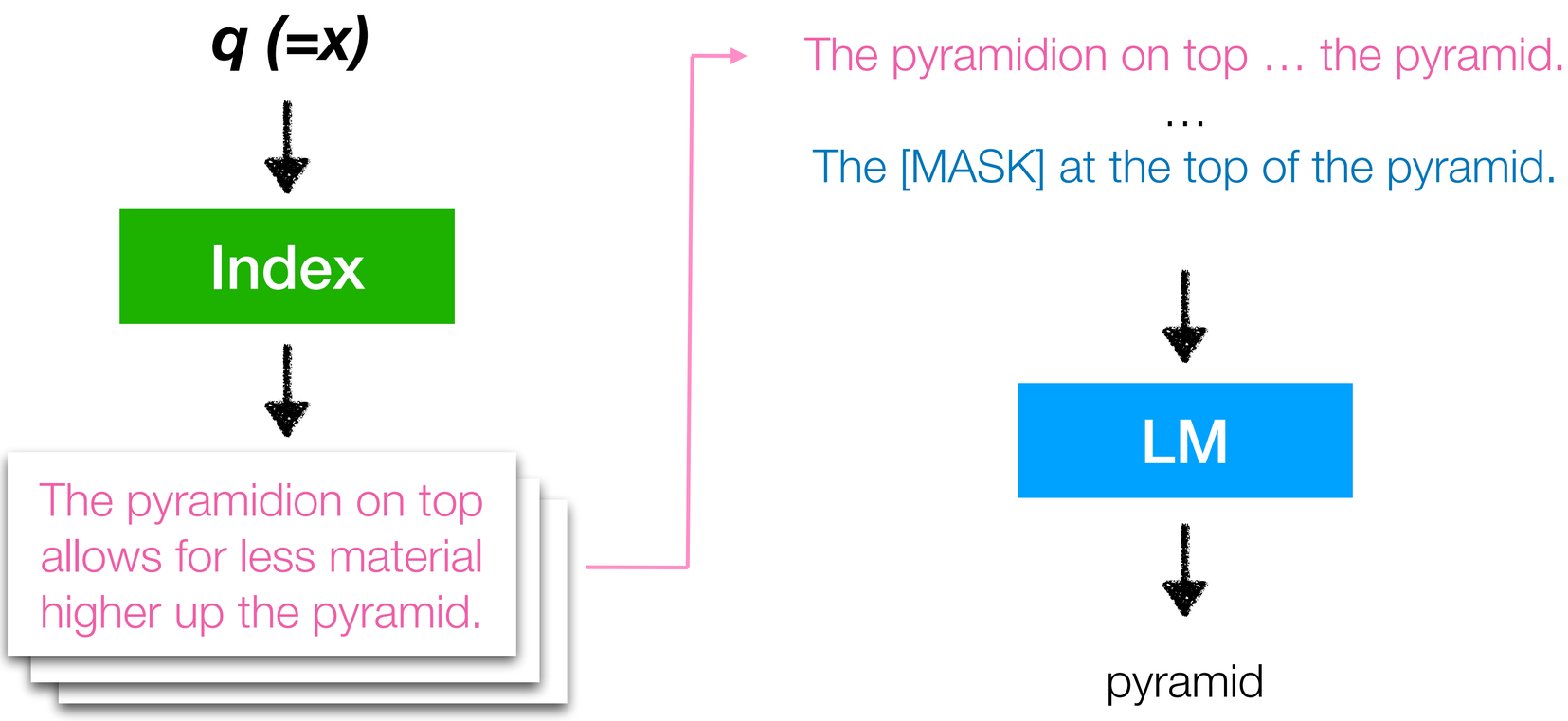


T updates

Refresh

REALM (Guu et al. 2020)

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

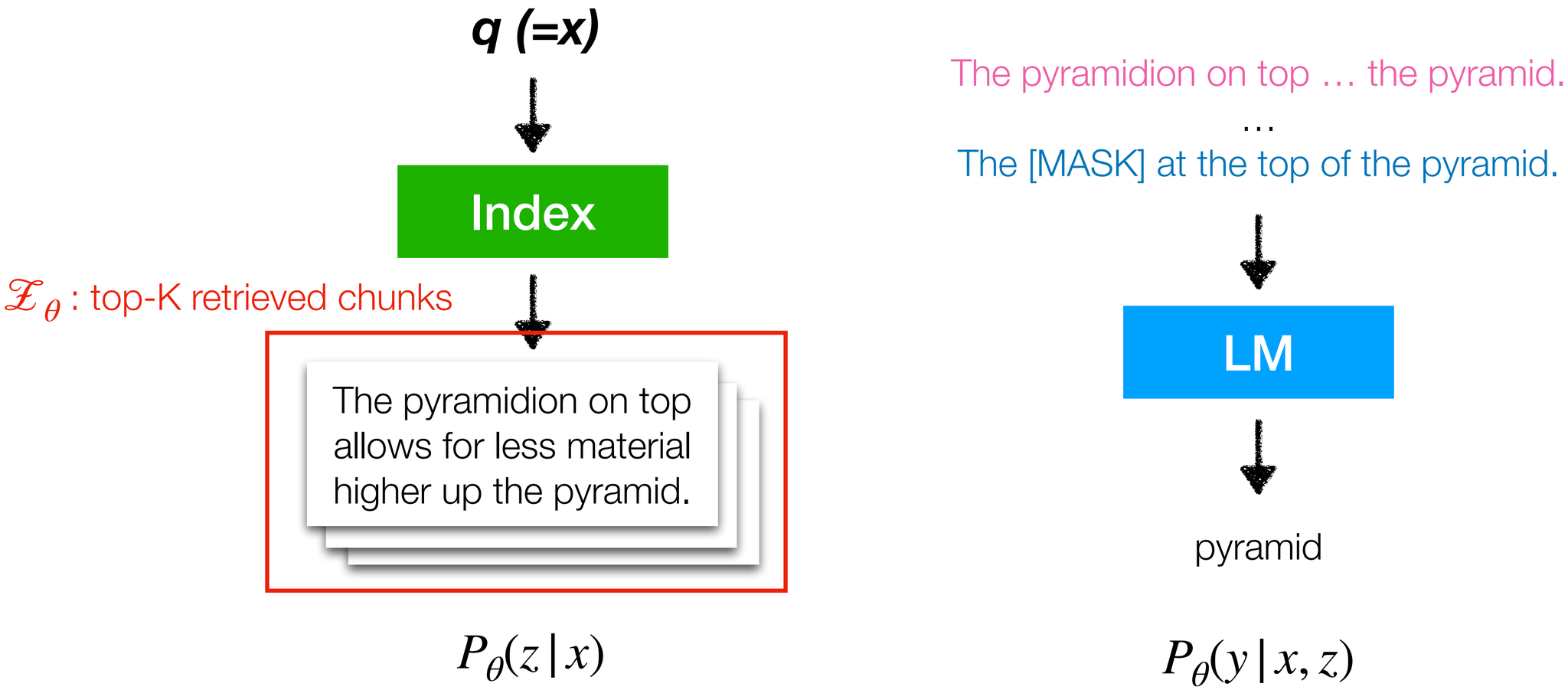


$P(z \mid x)$ $P(y \mid x, z)$

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

REALM: Training

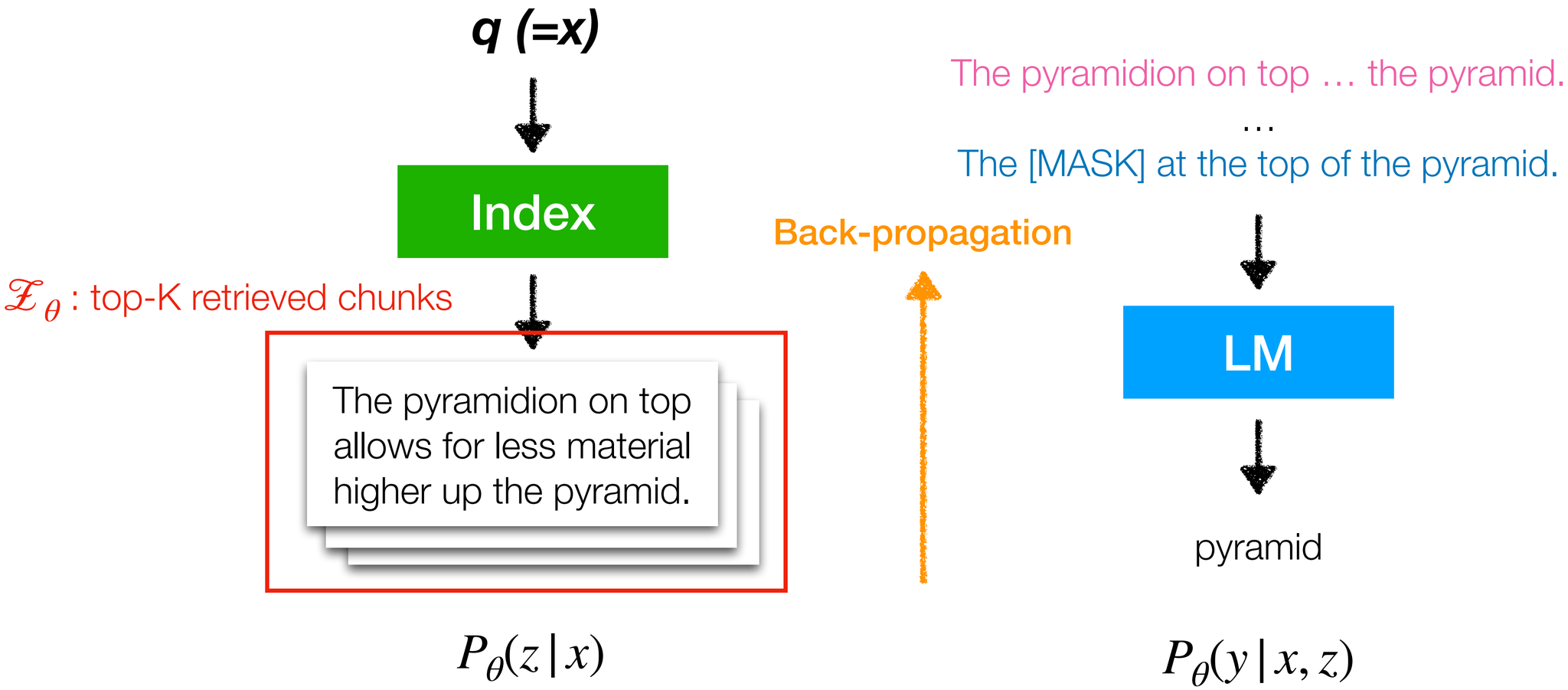
Objective: maximize $\sum_{z \in \mathcal{Z}_\theta} P_\theta(z \mid q) P_\theta(y \mid q, z)$



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

REALM: Training

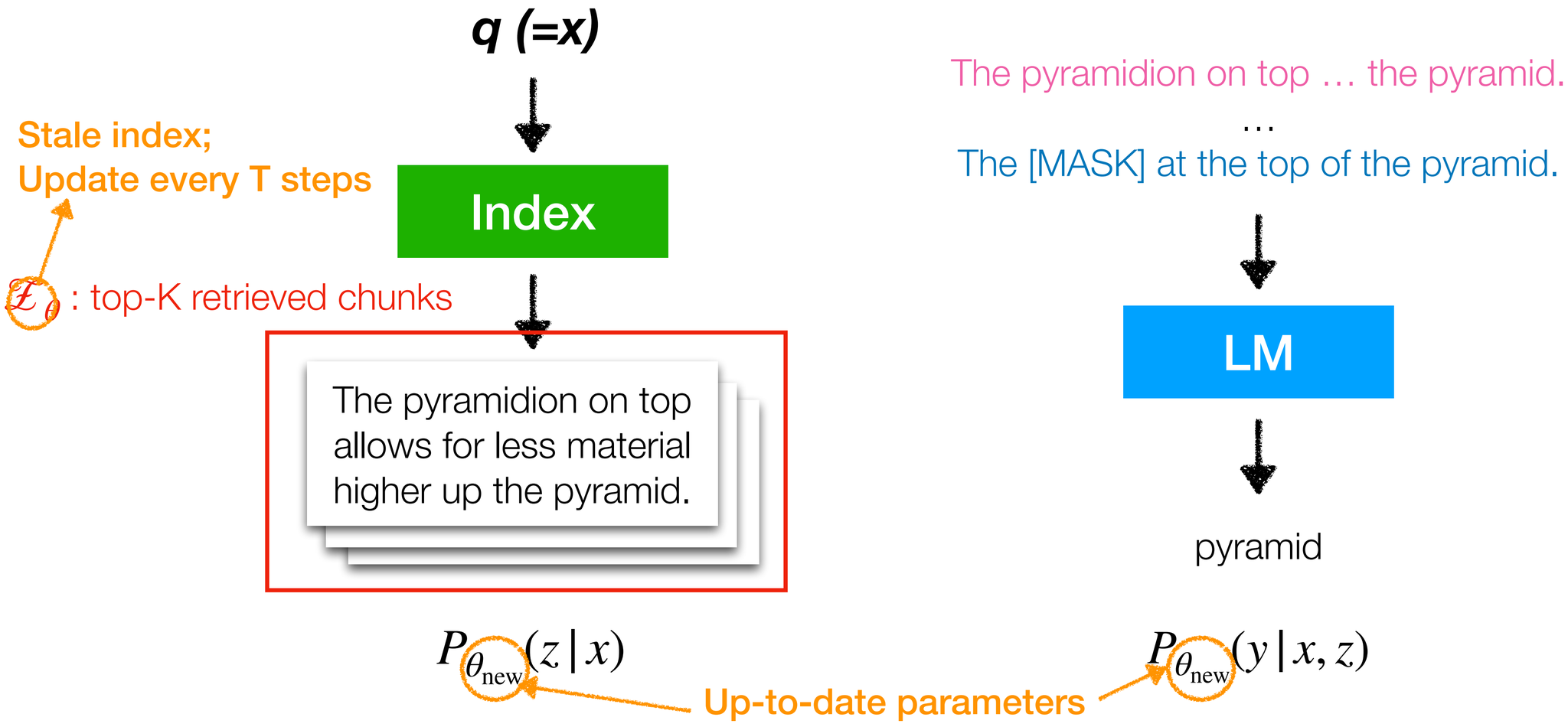
Objective: maximize $\sum_{z \in \mathcal{Z}_\theta} P_\theta(z \mid q) P_\theta(y \mid q, z)$



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

REALM: Training

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

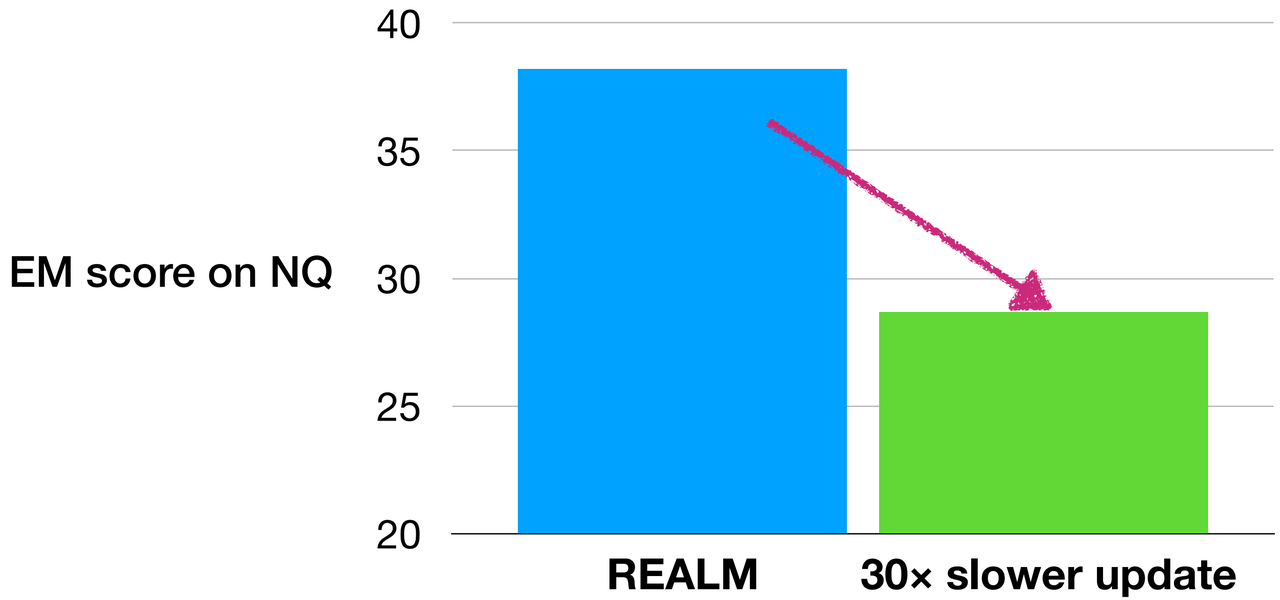


REALM: Index update rate

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

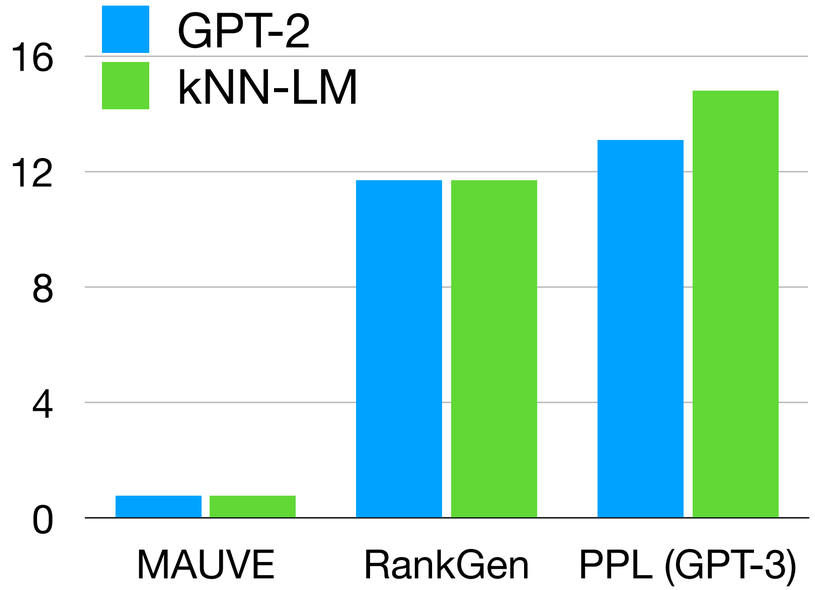
Open Questions

Challenge: retrieval-augmented LMs for applications

Open-ended text generation? Reasoning?

Challenge: retrieval-augmented LMs for applications

Open-ended text generation? Reasoning?

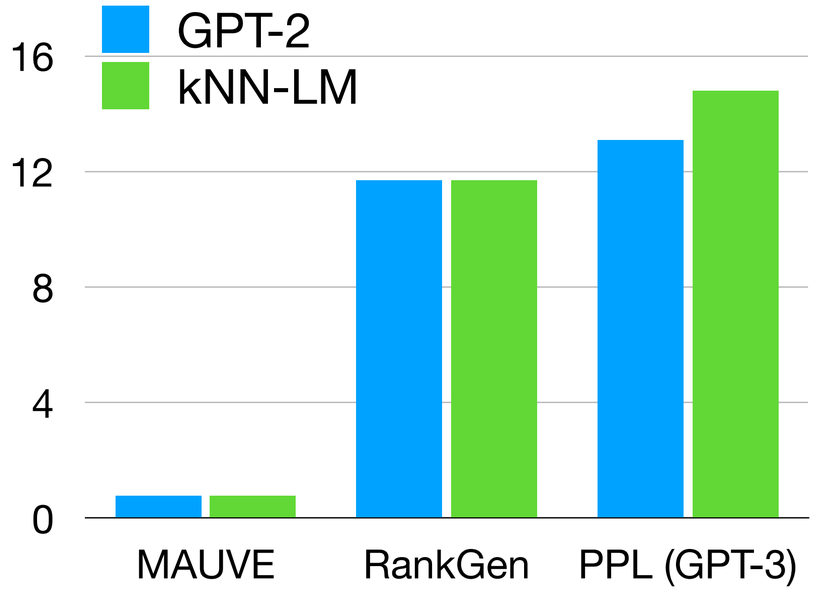


Wang et al. kNN-LM Does Not Improve Open-ended Text Generation. ACL 2023.

Challenge: retrieval-augmented LMs for applications

Open-ended text generation? Reasoning?

Doesn't improve open-ended generation



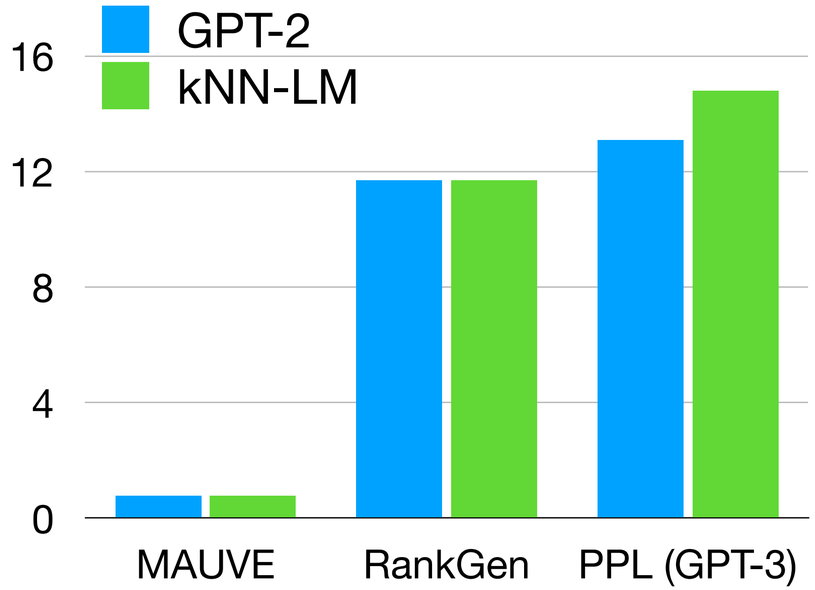
Wang et al. kNN-LM Does Not Improve Open-ended Text Generation. ACL 2023.

Challenge: retrieval-augmented LMs for applications

Open-ended text generation? Reasoning?

Doesn't improve open-ended generation

Failure of retrieval in reasoning task

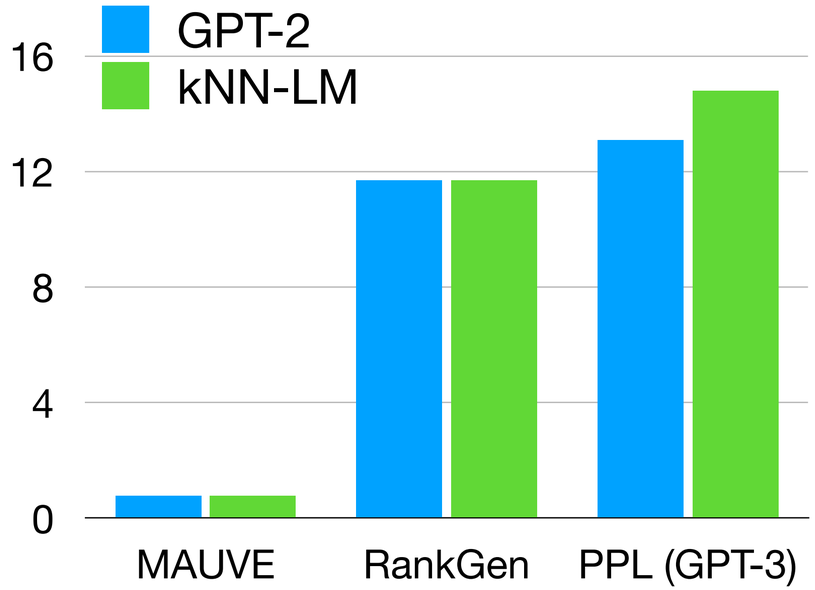


Wang et al. kNN-LM Does Not Improve Open-ended Text Generation. ACL 2023.

Challenge: retrieval-augmented LMs for applications

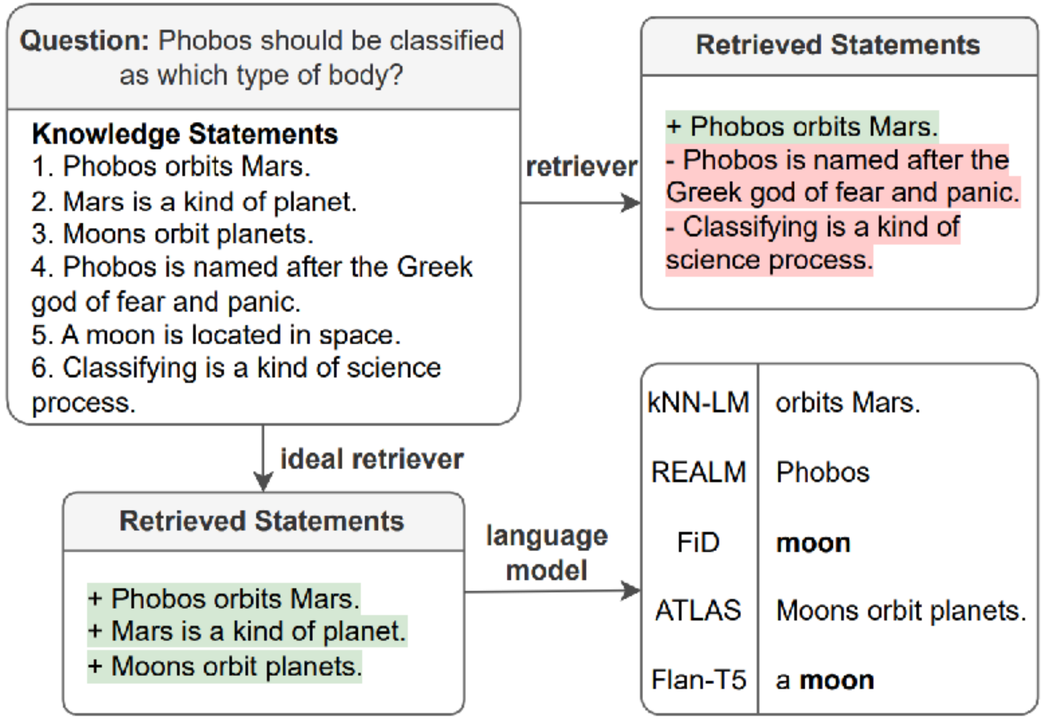
Open-ended text generation? Reasoning?

Doesn't improve open-ended generation



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

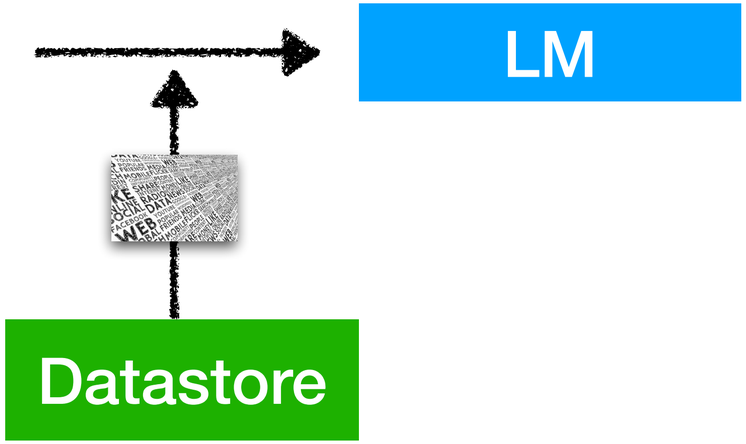
Failure of retrieval in reasoning task



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

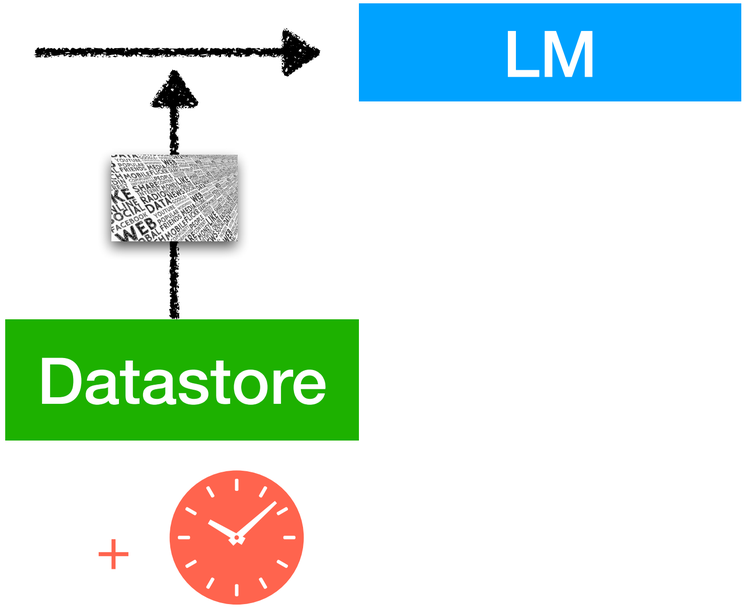
Challenge: efficiency retrieval-augmented LMs

Additional costs from retrieval augmentation



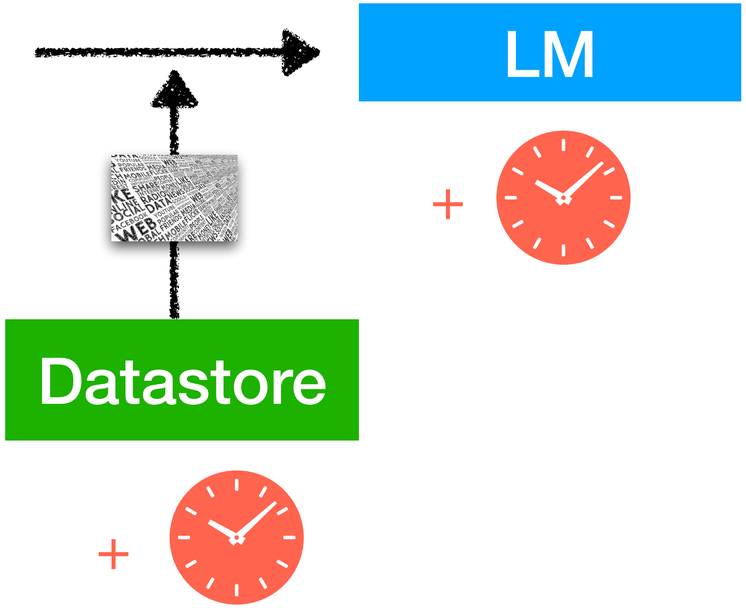
Challenge: efficiency retrieval-augmented LMs

Additional costs from retrieval augmentation



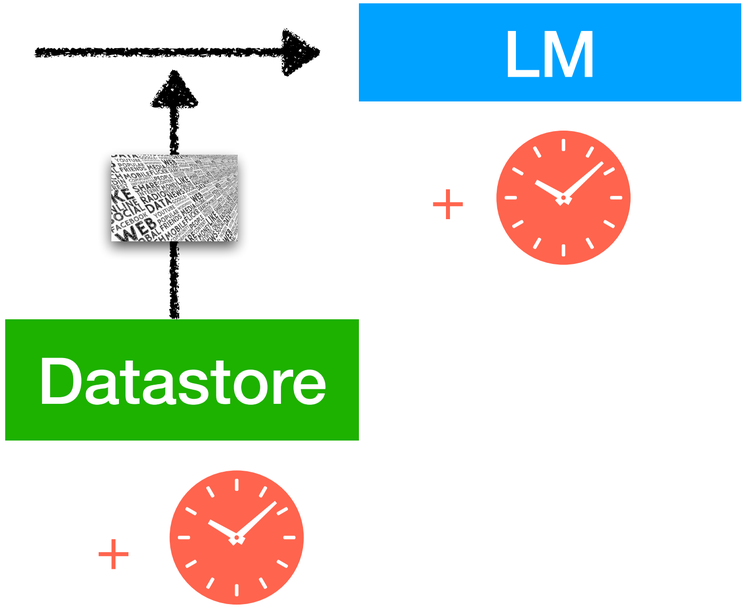
Challenge: efficiency retrieval-augmented LMs

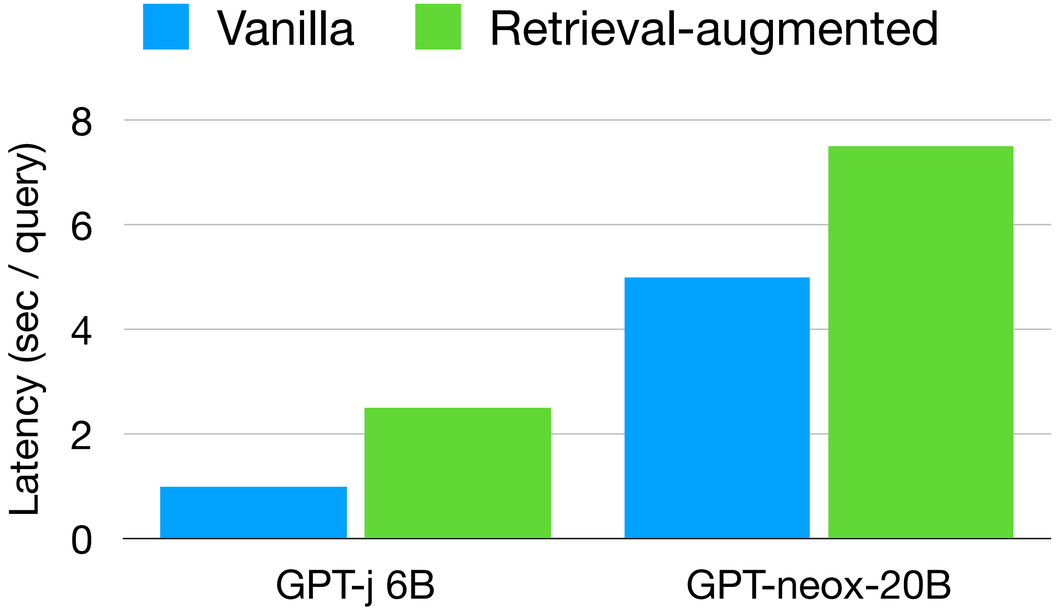
Additional costs from retrieval augmentation



Challenge: efficiency retrieval-augmented LMs

Additional costs from retrieval augmentation

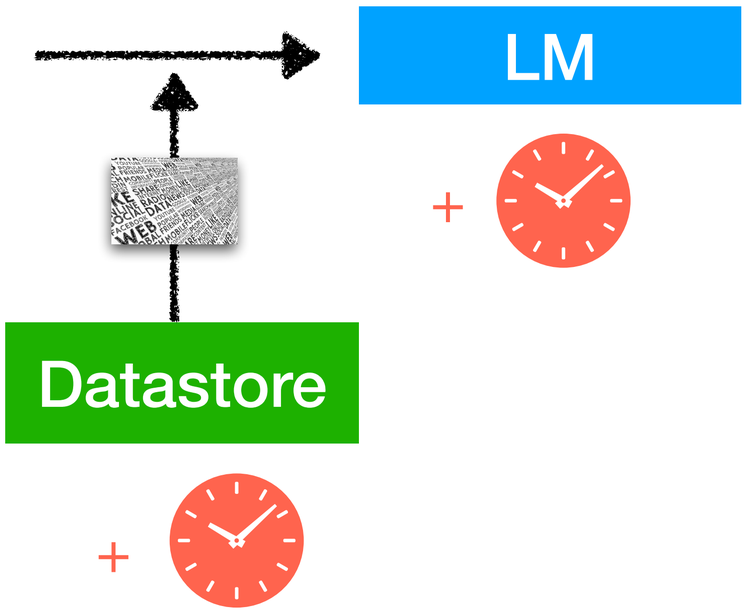




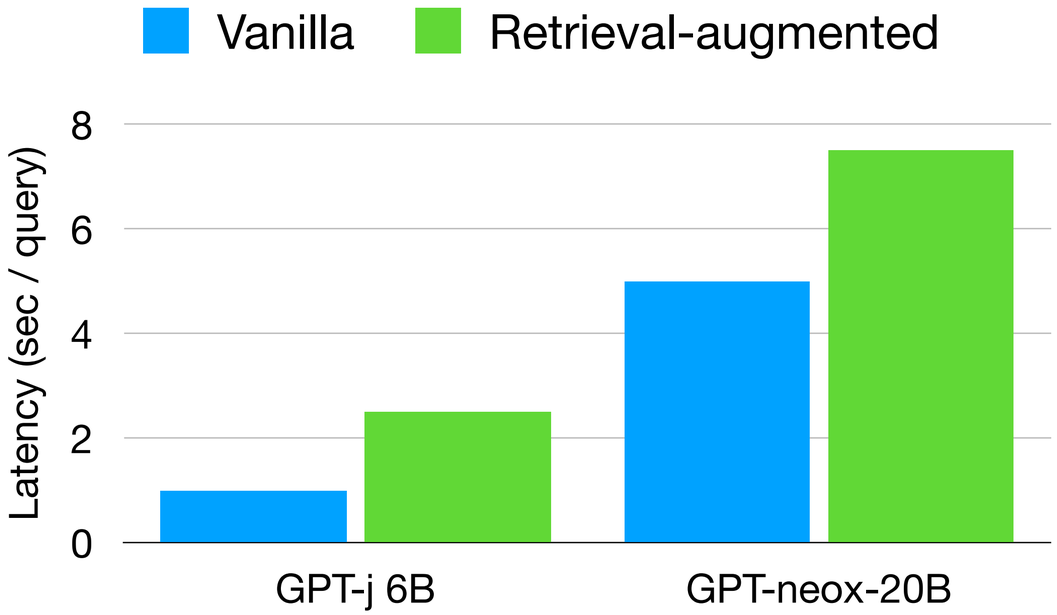
Mallen*, Asai* et al., When Not to Trust Language Models:
Investigating Effectiveness of Parametric and Non-Parametric Memories (Best Video; Oral) 2023.

Challenge: efficiency retrieval-augmented LMs

Additional costs from retrieval augmentation



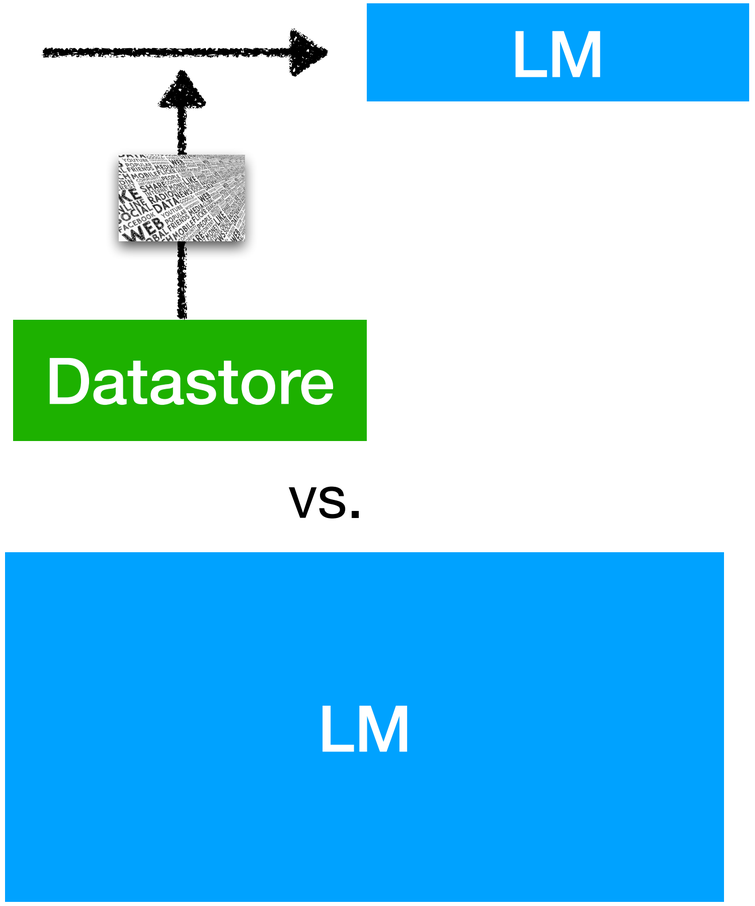
Retrieval-augmented LMs add inference costs



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

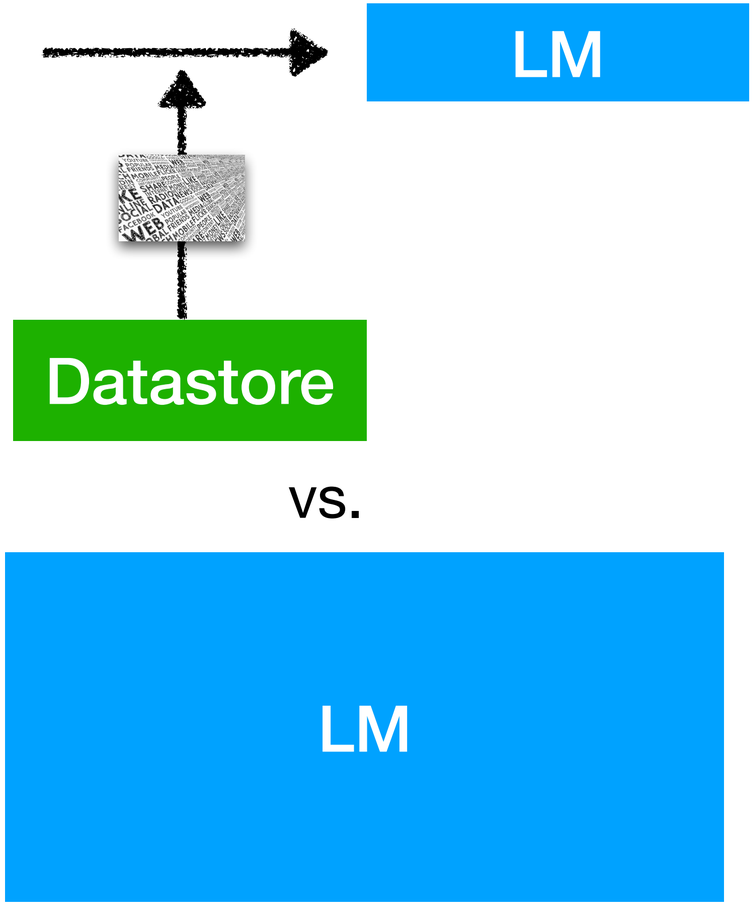
Challenge: scaling retrieval-augmented LMs

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



Challenge: scaling retrieval-augmented LMs

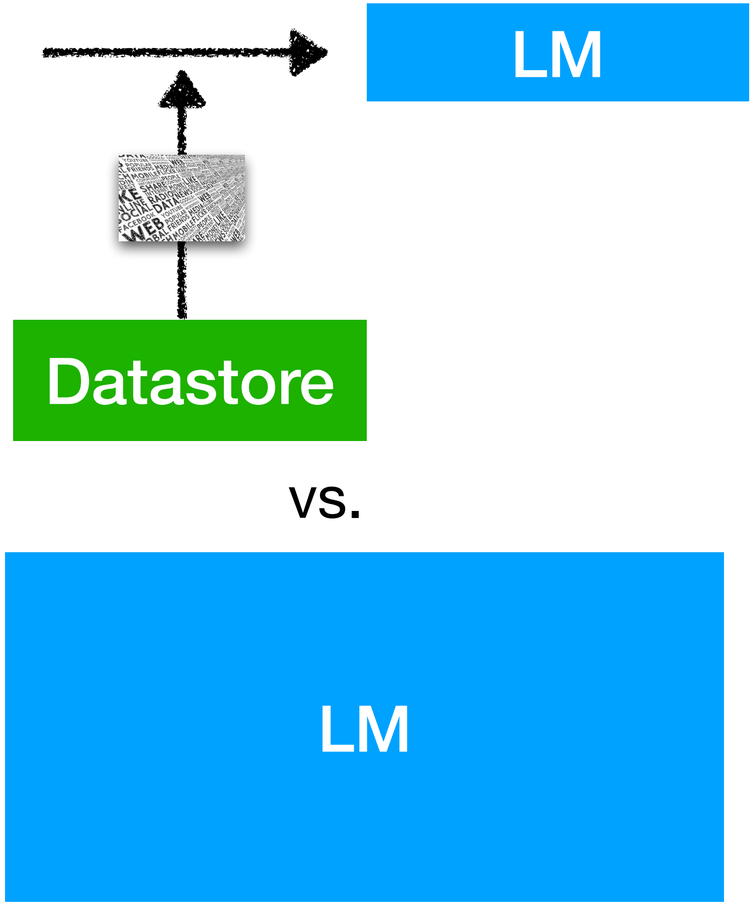
A small LM + a large datastore ≈ a large parametric LM?



	LM	Datastore
	# of parameters	# of tokens
kNN-LM (Khandelwal et al., 2020)	250M	$\leq$ 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)	$\leq$ 175B	~5B

Challenge: scaling retrieval-augmented LMs

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



	LM	Datastore
	# of parameters	# of tokens
kNN-LM (Khandelwal et al., 2020)	250M	$\leq$ 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)	$\leq$ 175B	~5B

Challenge: robustness and controllability

What are the latest discoveries from the James Webb Space Telescope?

The James Webb Space Telescope is designed to peer into the dusty clouds of gas where stars and planetary systems are born. Webb has captured the first direct image of an exoplanet, and the Pillars of Creation in the Eagle Nebula[1][2]. Additionally, the telescope will be used to study the next interstellar interloper[3].

(*Some generated statements may not be fully supported by citations, while others are fully supported.)

Cited Webpages

[1]: nasa.gov (✗ citation does not support its associated statement)
NASA's Webb Confirms Its First Exoplanet
... Researchers confirmed an exoplanet, a planet that orbits another star, using NASA's James Webb Space Telescope for the first time. ...

[2]: cnn.com (⚠ citation partially supports its associated statement)
Pillars of Creation: James Webb Space Telescope ...
... The Pillars of Creation, in the Eagle Nebula, is a star-forming region captured in a new image (right) by the James Webb Space Telescope that reveals more detail than a 2014 image (left) by Hubble ...

[3]: nasa.gov (✓ citation fully supports its associated statement)
Studying the Next Interstellar Interloper with Webb
...Scientists have had only limited ability to study these objects once discovered, but all of that is about to change with NASA's James Webb Space Telescope...The team will use Webb's spectroscopic capabilities in both the near-infrared and mid-infrared bands to study two different aspects of the interstellar object.

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

Challenge: robustness and controllability

What are the latest discoveries from the James Webb Space Telescope?

The James Webb Space Telescope is designed to peer into the dusty clouds of gas where stars and planetary systems are born. Webb has captured the first direct image of an exoplanet, and the Pillars of Creation in the Eagle Nebula[1][2]. Additionally, the telescope will be used to study the next interstellar interloper[3].

(*Some generated statements may not be fully supported by citations, while others are fully supported.)

Cited Webpages

[1]: nasa.gov (✗ citation does not support its associated statement)
NASA's Webb Confirms Its First Exoplanet
... Researchers confirmed an exoplanet, a planet that orbits another star, using NASA's James Webb Space Telescope for the first time. ...

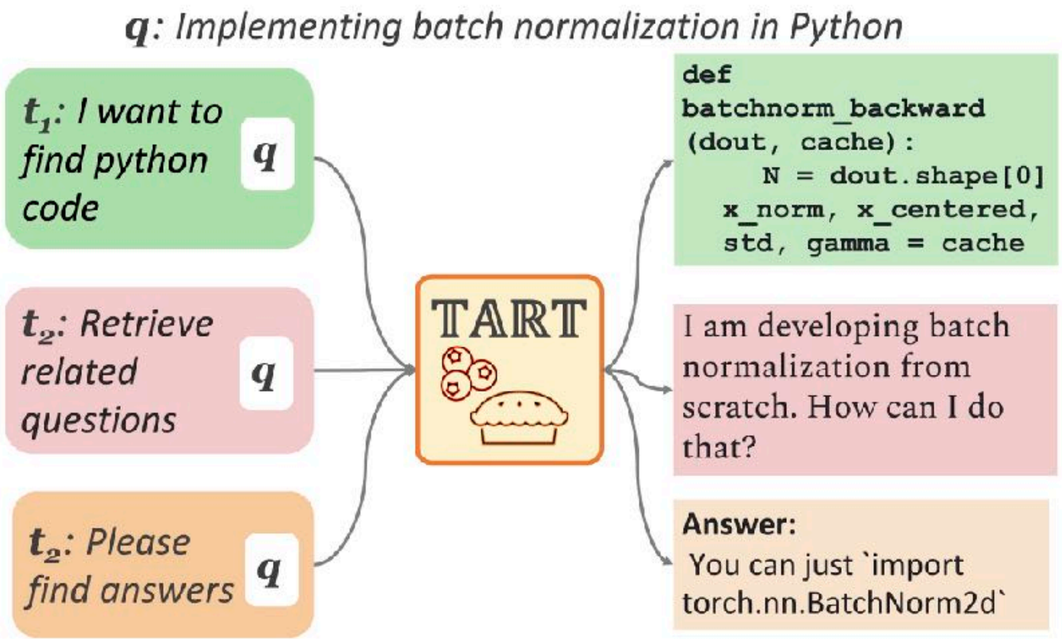
[2]: cnn.com (⚠ citation partially supports its associated statement)
Pillars of Creation: James Webb Space Telescope ...
... The Pillars of Creation, in the Eagle Nebula, is a star-forming region captured in a new image (right) by the James Webb Space Telescope that reveals more detail than a 2014 image (left) by Hubble ...

[3]: nasa.gov (✅ citation fully supports its associated statement)
Studying the Next Interstellar Interloper with Webb
...Scientists have had only limited ability to study these objects once discovered, but all of that is about to change with NASA's James Webb Space Telescope...The team will use Webb's spectroscopic capabilities in both the near-infrared and mid-infrared bands to study two different aspects of the interstellar object.

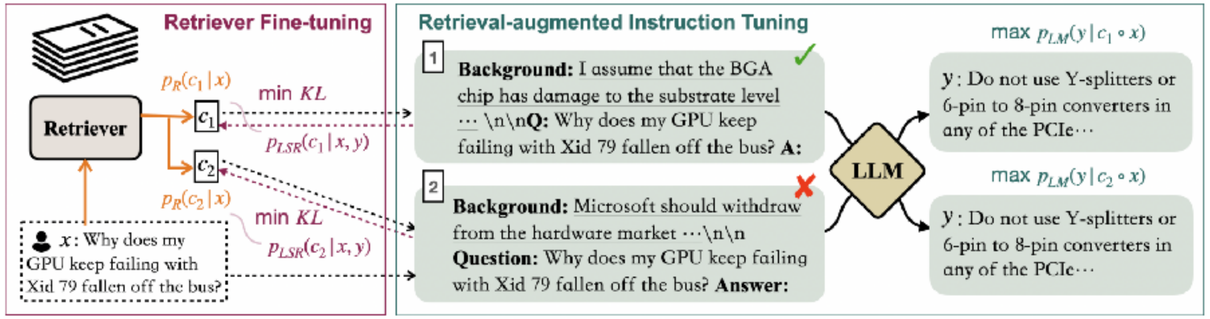
Retrieval-augmented LMs can still hallucinate

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

Beyond semantic and lexical-similarity based search



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

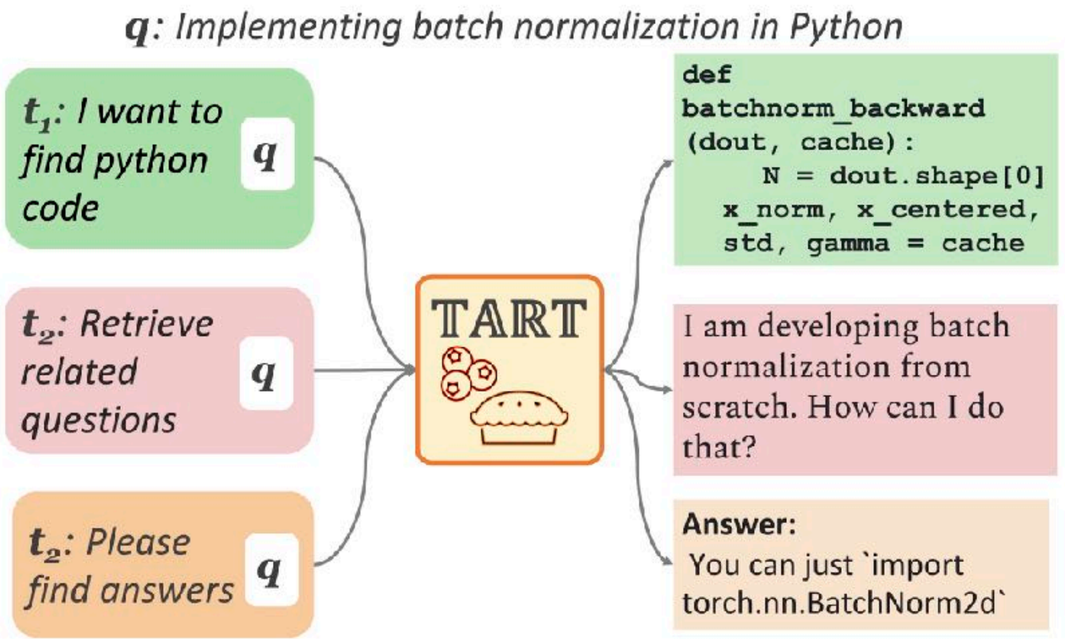


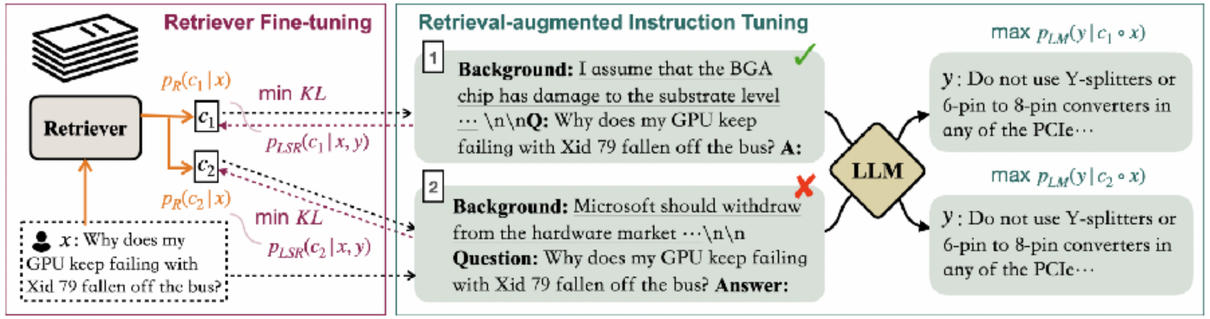
0-shot	BoolQ	PIQA	SIQA	HellaSwag	WinoGrande
LLAMA 65B	85.3	82.8	52.3	84.2	77.0
RA-DIT 65B w/o retrieval	**86.7**	83.7	57.9	85.1	79.8
RA-DIT 65B	85.6	**84.4**	**58.4**	**85.4**	**80.0**

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

Beyond semantic and lexical-similarity based search

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



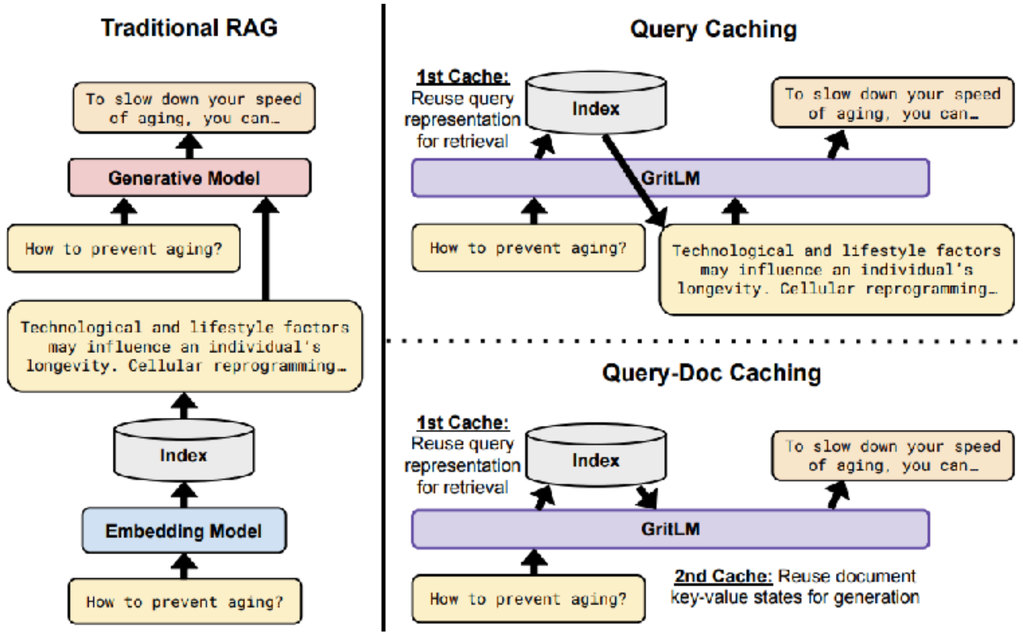


0-shot	BoolQ	PIQA	SIQA	HellaSwag	WinoGrande
LLAMA 65B	85.3	82.8	52.3	84.2	77.0
RA-DIT 65B w/o retrieval	**86.7**	83.7	57.9	85.1	79.8
RA-DIT 65B	85.6	**84.4**	**58.4**	**85.4**	**80.0**

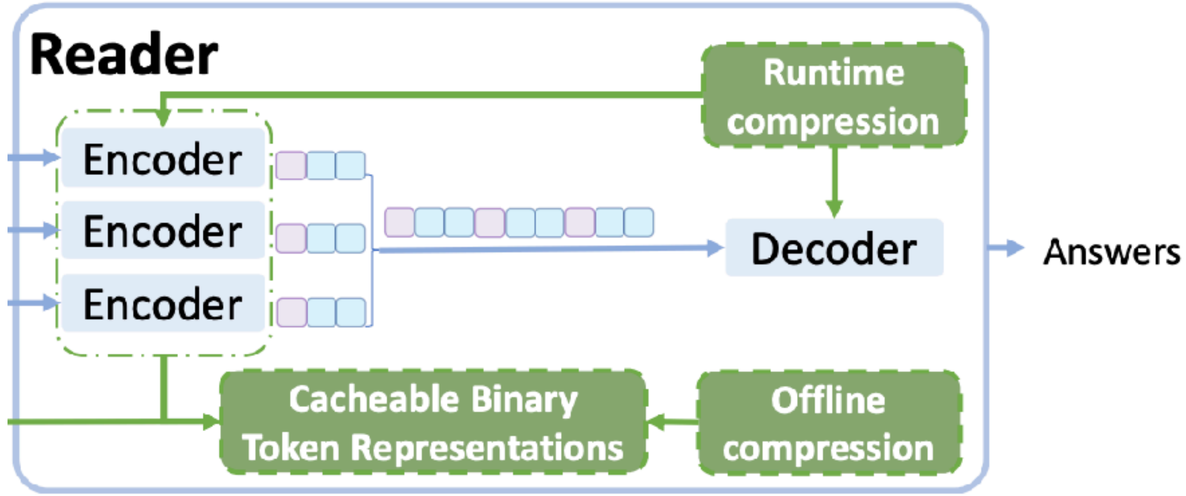
Asai et al., Task-aware Retrieval with Instruction. Findings of ACL 2023.

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

New architectures for performance and efficiency



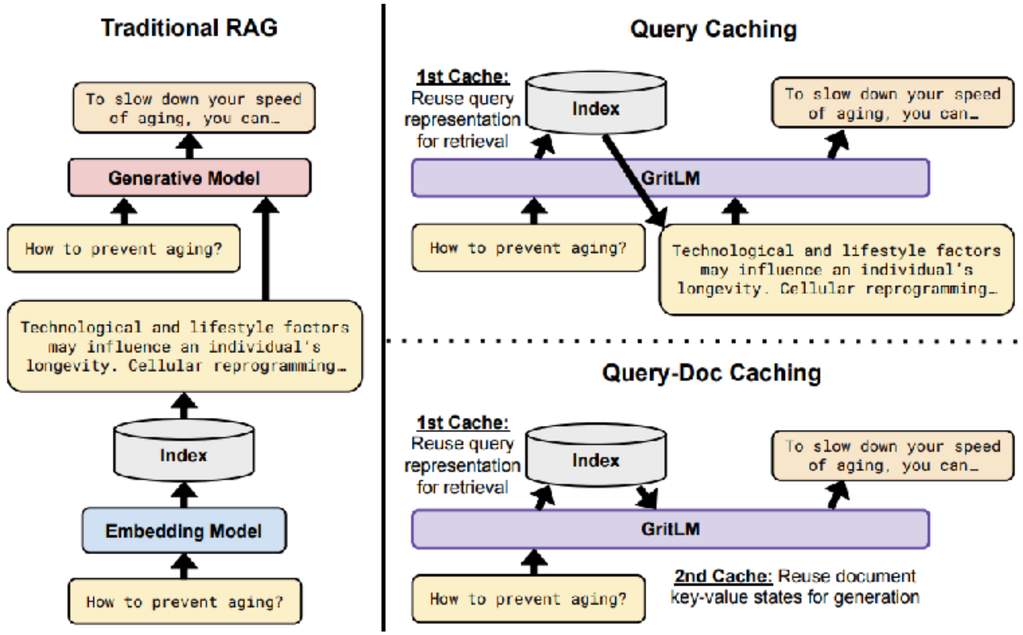
Muennighoff et al. Generative Representational Instruction Tuning. 2024.



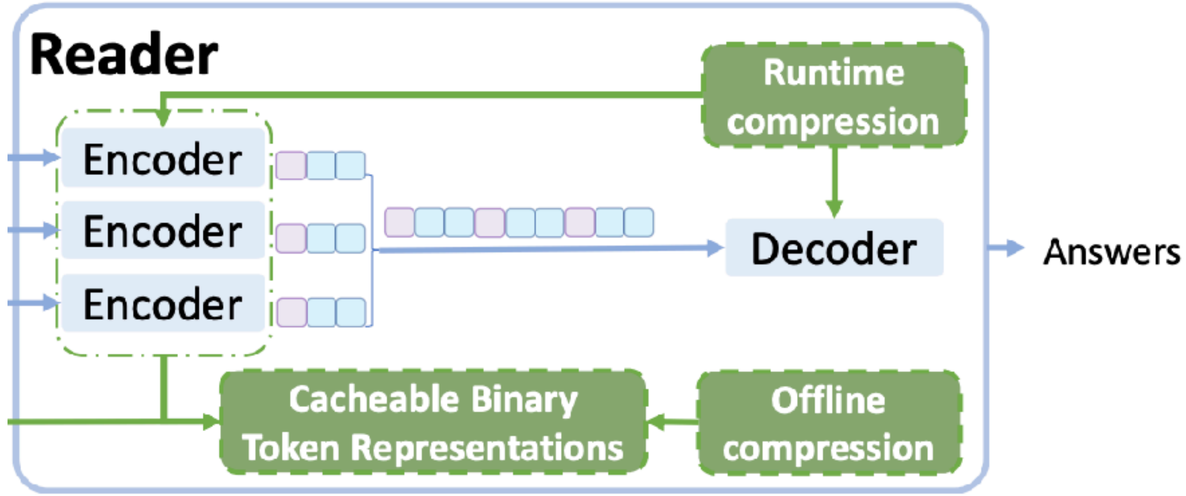
Cao et al. BTR: Binary Token Representations for Efficient Retrieval Augmented Language Models. ICLR 2024.

New architectures for performance and efficiency

Further explorations of unified architectures & caching



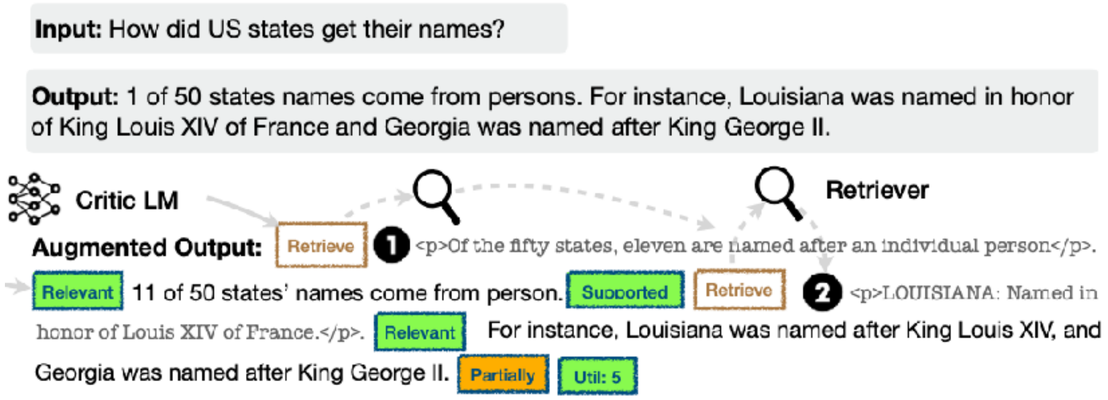
Muennighoff et al. Generative Representational Instruction Tuning. 2024.



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

Training LMs with Retrieval

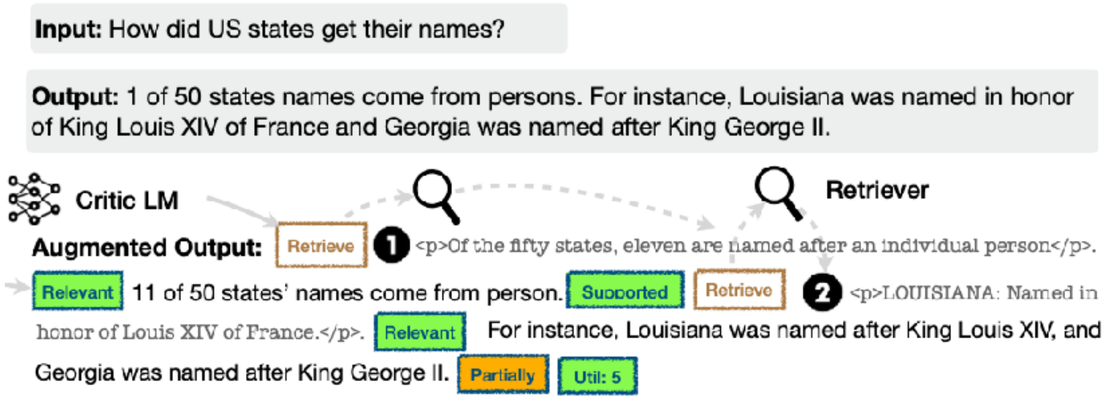
Training LMs with Retrieval



Instruction-tuning with retrieval

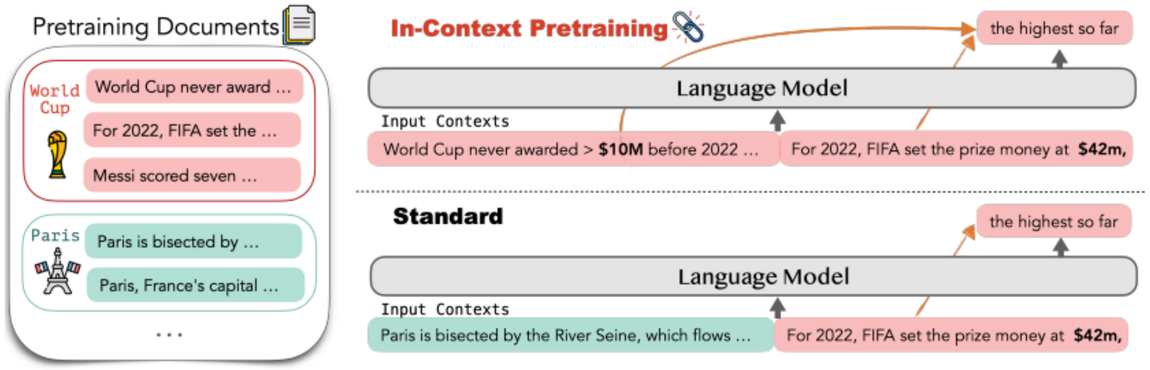
Asai et al. Self-RAG: Learning to Retrieve, Generate and Critique with Retrieval. ICLR 2024.

Training LMs with Retrieval



Instruction-tuning with retrieval

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

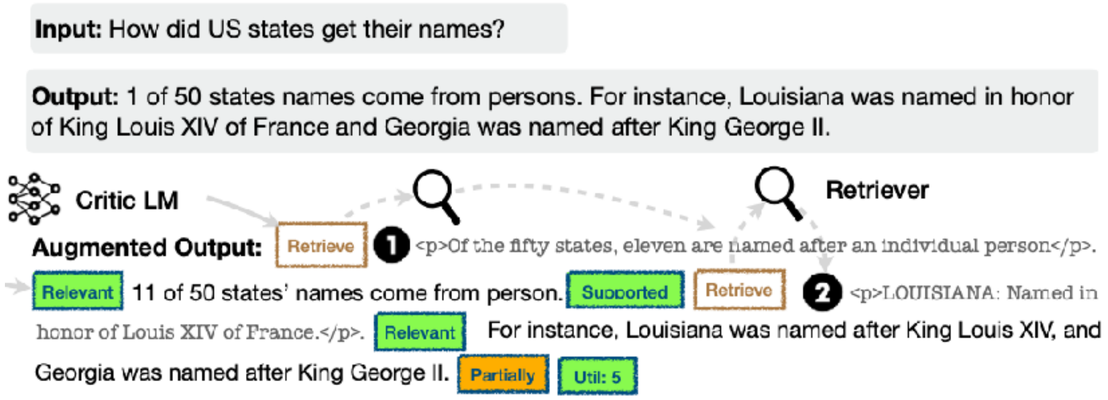


Retrieval-aware pre-training

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

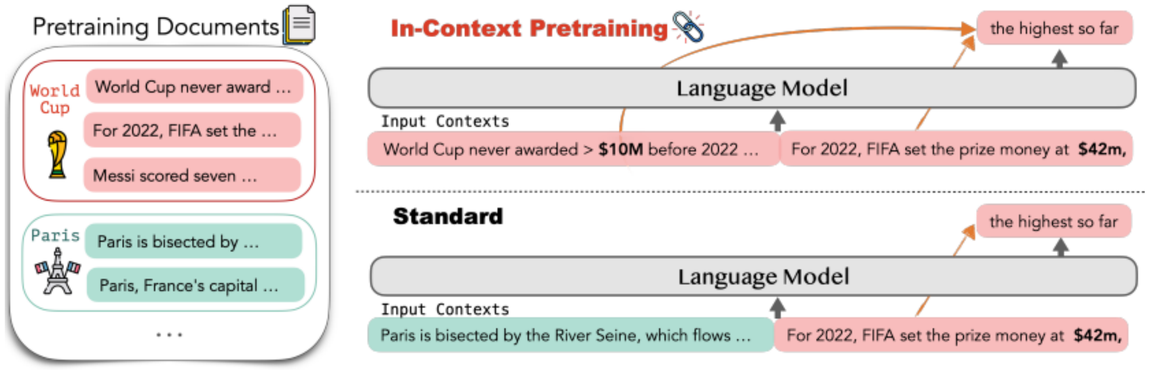
Training LMs with Retrieval

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.

Summing up Retrieval-Augmented LMs

- Retrieval-augmented LMs reduce many problems: hallucination, lack of interpretability, large parameter counts
- Different architectures with their own training and inference tradeoffs
- Jointly training retrieval and language models is important but hard
- Usually evaluated like other language models, but users often see retrieval results directly