Language Models as the New Search Engines

Tianyu Gao

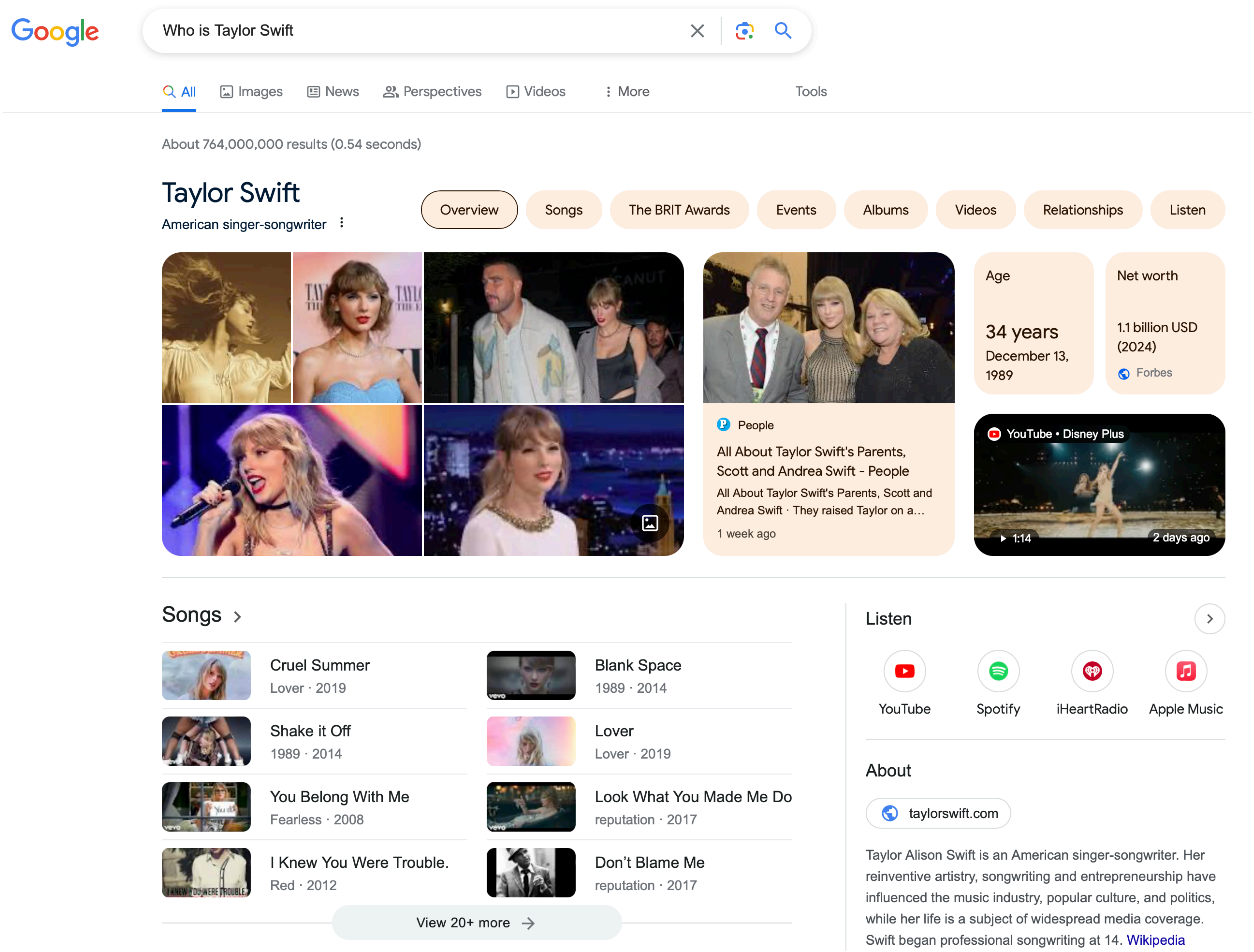
Princeton Language and Intelligence (PLI)

Princeton University

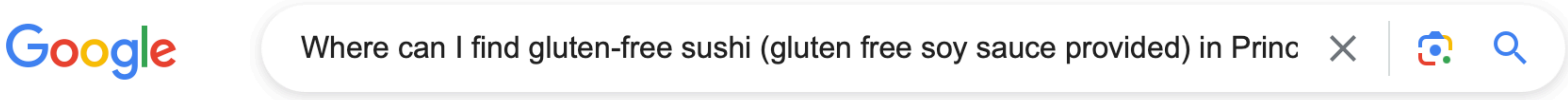
How do we conduct information-seeking activities?

The Internet era: search engines

The Internet era: search engines



The Internet era: search engines



The Internet era: search engines

Google

Where can I find gluten-free sushi (gluten free soy sauce provided) in Princ

Places

Rating · Price · Hours

Sponsored

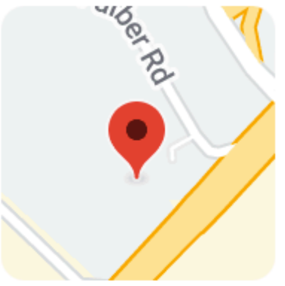
Whole Foods Market

4.4 ★★★★★ (1.4K) · $$$ · Grocery store

3495 U.S. 1 South

Open · Closes 9 PM

Delivery



MTea Sushi & Dessert

4.8 ★★★★★ (213) · Sushi

86 Nassau St

"... thought of little details like providing small containers of **soy sauce**."

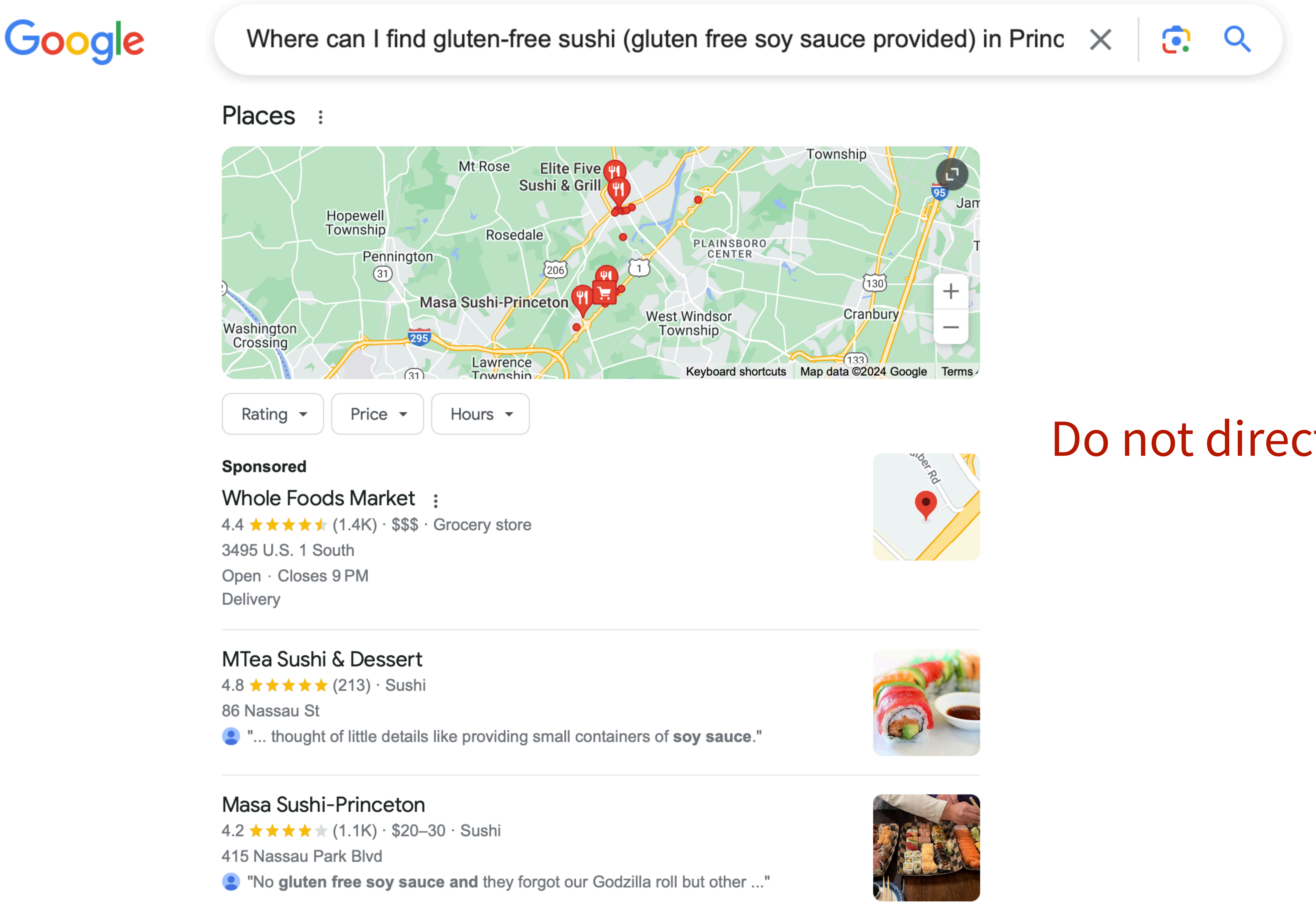
Masa Sushi-Princeton

4.2 ★★★★★ (1.1K) · $20–30 · Sushi

415 Nassau Park Blvd

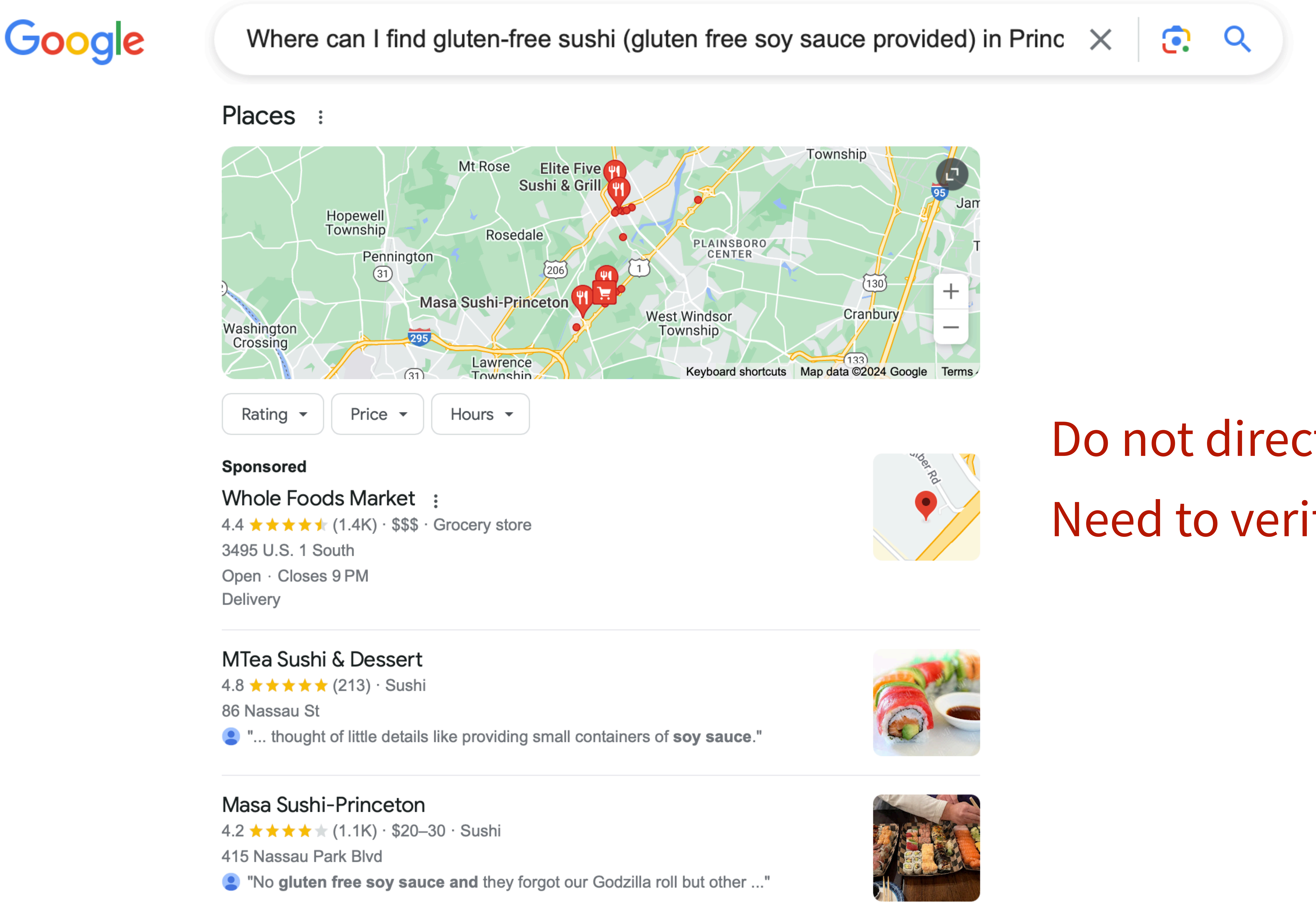
"No **gluten free soy sauce and** they forgot our Godzilla roll but other ..."

The Internet era: search engines



Do not directly provide answers

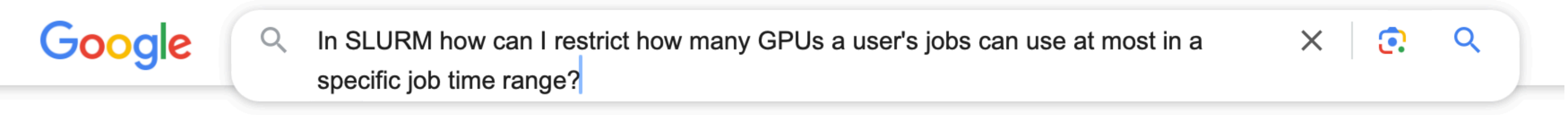
The Internet era: search engines



Do not directly provide answers

Need to verify results one by one

The Internet era: search engines



The Internet era: search engines

Google

In SLURM how can I restrict how many GPUs a user's jobs can use at most in a specific job time range?

Johns Hopkins University
https://jhpce.jhu.edu › knowledge-base › setting-a-time-li...

Setting a time limit for your SLURM job on JHPCE

Feb 13, 2024 — The default **time limit** for **job** execution on the shared and **gpu** partitions is set **to** 1 day, ensuring that shorter **tasks do** not face unnecessary ...

Missing: ~~restrict~~ ~~most~~

Unix & Linux Stack Exchange
https://unix.stackexchange.com › questions › running-...

Running multiple SLURM jobs on the same GPU

Nov 25, 2022 — It **will** run **jobs** in parallel if you have multiple **GPUs** that **can** run the **jobs**, otherwise it runs them in **series** as a **GPU** becomes available. I ...

1 answer · Top answer: You should use "Sharding" GRES (gres:shard) instead of gres:GPU, a…

Missing: ~~range?~~ | Show results with: range?

Slurm Documentation
https://slurm.schedmd.com › faq

Frequently Asked Questions - Slurm Workload Manager

Feb 12, 2024 — If **jobs do** not specify **time** limits, then all **jobs will** receive the same **time limit** (that associated with the partition), and the ability **to** ...

The Internet era: search engines

Google

In SLURM how can I restrict how many GPUs a user's jobs can use at most in a specific job time range?

Johns Hopkins University
https://jhpce.jhu.edu › knowledge-base › setting-a-time-li...

Setting a time limit for your SLURM job on JHPCE

Feb 13, 2024 — The default **time limit** for **job** execution on the shared and **gpu** partitions is set **to** 1 day, ensuring that shorter **tasks do** not face unnecessary ...

Missing: ~~restrict~~ ~~most~~

Unix & Linux Stack Exchange
https://unix.stackexchange.com › questions › running-...

Running multiple SLURM jobs on the same GPU

Nov 25, 2022 — It **will** run **jobs** in parallel if you have multiple **GPUs** that **can** run the **jobs**, otherwise it runs them in **series** as a **GPU** becomes available. I ...

1 answer · Top answer: You should use "Sharding" GRES (gres:shard) instead of gres:GPU, a...

Missing: ~~range?~~ | Show results with: range?

Slurm Documentation
https://slurm.schedmd.com › faq

Frequently Asked Questions - Slurm Workload Manager

Feb 12, 2024 — If **jobs do** not specify **time** limits, then all **jobs will** receive the same **time limit** (that associated with the partition), and the ability **to** ...

Too many irrelevant results

The Internet era: search engines

Google

In SLURM how can I restrict how many GPUs a user's jobs can use at most in a specific job time range?

Johns Hopkins University
https://jhpce.jhu.edu › knowledge-base › setting-a-time-li...

Setting a time limit for your SLURM job on JHPCE

Feb 13, 2024 — The default **time limit** for **job** execution on the shared and **gpu** partitions is set **to** 1 day, ensuring that shorter **tasks do** not face unnecessary ...
Missing: ~~restrict~~ ~~most~~

Unix & Linux Stack Exchange
https://unix.stackexchange.com › questions › running-...

Running multiple SLURM jobs on the same GPU

Nov 25, 2022 — It **will** run **jobs** in parallel if you have multiple **GPUs** that **can** run the **jobs**, otherwise it runs them in **series** as a **GPU** becomes available. I ...
1 answer · Top answer: You should use "Sharding" GRES (gres:shard) instead of gres:GPU, a...
Missing: ~~range?~~ | Show results with: range?

Slurm Documentation
https://slurm.schedmd.com › faq

Frequently Asked Questions - Slurm Workload Manager

Feb 12, 2024 — If **jobs do** not specify **time** limits, then all **jobs will** receive the same **time limit** (that associated with the partition), and the ability **to** ...

Too many irrelevant results

Word-matching does not return relevant webpages

Language models: the new "search engines"

You
In SLURM how can I restrict how many GPUs a user's jobs can use at most in a specific job time range?

Language models: the new “search engines”

You
In SLURM how can I restrict how many GPUs a user's jobs can use at most in a specific job time range?

ChatGPT
To restrict the number of GPUs a user's jobs can use at most within a specific job time range in a SLURM environment, you will need to utilize SLURM's Quality of Service (QoS) and account management features. This approach allows you to define limits on resource usage that apply to users, accounts, or even specific jobs based on various criteria, including job time ranges. Here's a step-by-step guide to setting up such restrictions:

1. Define or Update a Quality of Service (QoS)

First, you need to define a Quality of Service (QoS) that specifies the GPU usage limits. If a suitable QoS already exists, you may update it instead.

To create a new QoS:

bash Copy code

```bash
sudo sacctmgr add qos name=gpu_limit maxtresperuser=gpu=X
```

Language models: the new “search engines”

You
In SLURM how can I restrict how many GPUs a user's jobs can use at most in a specific job time range?

ChatGPT
To restrict the number of GPUs a user's jobs can use at most within a specific job time range in a SLURM environment, you will need to utilize SLURM's Quality of Service (QoS) and account management features. This approach allows you to define limits on resource usage that apply to users, accounts, or even specific jobs based on various criteria, including job time ranges. Here's a step-by-step guide to setting up such restrictions:

1. Define or Update a Quality of Service (QoS)

First, you need to define a Quality of Service (QoS) that specifies the GPU usage limits. If a suitable QoS already exists, you may update it instead.

To create a new QoS:

bash Copy code

```bash
sudo sacctmgr add qos name=gpu_limit maxtresperuser=gpu=X
```

User-friendly presentations

Language models: the new “search engines”

You
In SLURM how can I restrict how many GPUs a user's jobs can use at most in a specific job time range?

ChatGPT
To restrict the number of GPUs a user's jobs can use at most within a specific job time range in a SLURM environment, you will need to utilize SLURM's Quality of Service (QoS) and account management features. This approach allows you to define limits on resource usage that apply to users, accounts, or even specific jobs based on various criteria, including job time ranges. Here's a step-by-step guide to setting up such restrictions:

1. Define or Update a Quality of Service (QoS)

First, you need to define a Quality of Service (QoS) that specifies the GPU usage limits. If a suitable QoS already exists, you may update it instead.

To create a new QoS:

bash Copy code

```bash
sudo sacctmgr add qos name=gpu_limit maxtresperuser=gpu=X
```

User-friendly presentations

Relevant answers

Language models: the new "search engines"

You
In SLURM how can I restrict how many GPUs a user's jobs can use at most in a specific job time range?

ChatGPT
To restrict the number of GPUs a user's jobs can use at most within a specific job time range in a SLURM environment, you will need to utilize SLURM's Quality of Service (QoS) and account management features. This approach allows you to define limits on resource usage that apply to users, accounts, or even specific jobs based on various criteria, including job time ranges. Here's a step-by-step guide to setting up such restrictions:

1. Define or Update a Quality of Service (QoS)

First, you need to define a Quality of Service (QoS) that specifies the GPU usage limits. If a suitable QoS already exists, you may update it instead.

To create a new QoS:

bash Copy code

```bash
sudo sacctmgr add qos name=gpu_limit maxtresperuser=gpu=X
```

User-friendly presentations

Relevant answers

Boost productivity

Language models: the new “search engines”

You

Where can I find gluten-free sushi (gluten free soy sauce provided) in Princeton, NJ?

Language models: the new "search engines"

You

Where can I find gluten-free sushi (gluten free soy sauce provided) in Princeton, NJ?

ChatGPT

One option to find gluten-free sushi in Princeton, NJ, is to try Ajihei Sushi, located at 11 Chambers St, Princeton, NJ 08542. They offer a variety of sushi options and often accommodate dietary restrictions. It's recommended to call them directly to inquire about their gluten-free options and if they provide gluten-free soy sauce.

Another option is Sakura Japanese Sushi & Steak House, located at 4437 NJ-27, Princeton, NJ 08540. They typically offer a wide range of sushi options and may be able to accommodate gluten-free requests. Again, it's best to call ahead and ask about their gluten-free options and soy sauce availability.

Language models: the new "search engines"

You

Where can I find gluten-free sushi (gluten free soy sauce provided) in Princeton, NJ?

ChatGPT

One option to find gluten-free sushi in Princeton, NJ, is to try Ajihei Sushi, located at 11 Chambers St, Princeton, NJ 08542. They offer a variety of sushi options and often accommodate dietary restrictions. It's recommended to call them directly to inquire about their gluten-free options and if they provide gluten-free soy sauce. Does not provide gluten-free soy sauce

Another option is Sakura Japanese Sushi & Steak House, located at 4437 NJ-27, Princeton, NJ 08540. They typically offer a wide range of sushi options and may be able to accommodate gluten-free requests. Again, it's best to call ahead and ask about their gluten-free options and soy sauce availability.

Language models: the new "search engines"

You

Where can I find gluten-free sushi (gluten free soy sauce provided) in Princeton, NJ?

ChatGPT

One option to find gluten-free sushi in Princeton, NJ, is to try Ajihei Sushi, located at 11 Chambers St, Princeton, NJ 08542. They offer a variety of sushi options and often accommodate dietary restrictions. It's recommended to call them directly to inquire about their gluten-free options and if they provide gluten-free soy sauce. Does not provide gluten-free soy sauce

Another option is Sakura Japanese Sushi & Steak House, located at 4437 NJ-27, Princeton, NJ 08540. They typically offer a wide range of sushi options and may be able to accommodate gluten-free requests. Again, it's best to call ahead and ask about their gluten-free options and soy sauce availability. Does not exist?!

Language models: the new “search engines”

You
Where can I find gluten-free sushi (gluten free soy sauce provided) in Princeton, NJ?

ChatGPT
One option to find gluten-free sushi in Princeton, NJ, is to try Ajihei Sushi, located at 11 Chambers St, Princeton, NJ 08542. They offer a variety of sushi options and often accommodate dietary restrictions. It's recommended to call them directly to inquire about their gluten-free options and if they provide gluten-free soy sauce. Does not provide gluten-free soy sauce

Another option is Sakura Japanese Sushi & Steak House, located at 4437 NJ-27, Princeton, NJ 08540. They typically offer a wide range of sushi options and may be able to accommodate gluten-free requests. Again, it's best to call ahead and ask about their gluten-free options and soy sauce availability. Does not exist?!

“hallucination”

Language models: the new "search engines"

You

Where can I find gluten-free sushi (gluten free soy sauce provided) in Princeton, NJ?

ChatGPT

One option to find gluten-free sushi in Princeton, NJ, is to try Ajihei Sushi, located at 11 Chambers St, Princeton, NJ 08542. They offer a variety of sushi options and often accommodate dietary restrictions. It's recommended to call them directly to inquire about their gluten-free options and if they provide gluten-free soy sauce. Does not provide gluten-free soy sauce

Another option is Sakura Japanese Sushi & Steak House, located at 4437 NJ-27, Princeton, NJ 08540. They typically offer a wide range of sushi options and may be able to accommodate gluten-free requests. Again, it's best to call ahead and ask about their gluten-free options and soy sauce availability. Does not exist?!

"hallucination"

No references for users to verify

Language models: the new “search engines”

Language models: the new “search engines”

How can we make LMs better information-seeking tools?

Language models: the new “search engines”

How can we make LMs better information-seeking tools?

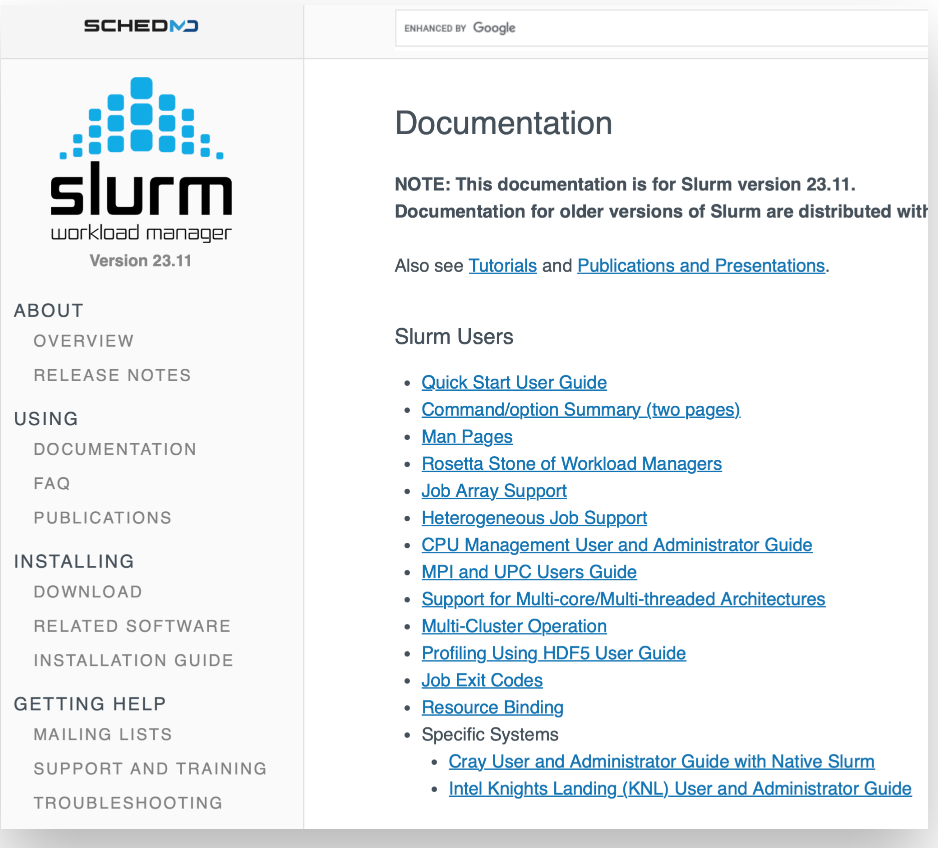
1. The ability to find and utilize reference materials

Language models: the new “search engines”

How can we make LMs better information-seeking tools?

1. The ability to find and utilize reference materials

Document-augmented


SCHEDMD
slurm
workload manager
Version 23.11
ABOUT
OVERVIEW
RELEASE NOTES
USING
DOCUMENTATION
FAQ
PUBLICATIONS
INSTALLING
DOWNLOAD
RELATED SOFTWARE
INSTALLATION GUIDE
GETTING HELP
MAILING LISTS
SUPPORT AND TRAINING
TROUBLESHOOTING

Documentation

NOTE: This documentation is for Slurm version 23.11.
Documentation for older versions of Slurm are distributed wit

Also see Tutorials and Publications and Presentations.

Slurm Users

- Quick Start User Guide
- Command/option Summary (two pages)
- Man Pages
- Rosetta Stone of Workload Managers
- Job Array Support
- Heterogeneous Job Support
- CPU Management User and Administrator Guide
- MPI and UPC Users Guide
- Support for Multi-core/Multi-threaded Architectures
- Multi-Cluster Operation
- Profiling Using HDF5 User Guide
- Job Exit Codes
- Resource Binding
- Specific Systems
  - Cray User and Administrator Guide with Native Slurm
  - Intel Knights Landing (KNL) User and Administrator Guide

Q: In SLURM how can i restrict …

↓

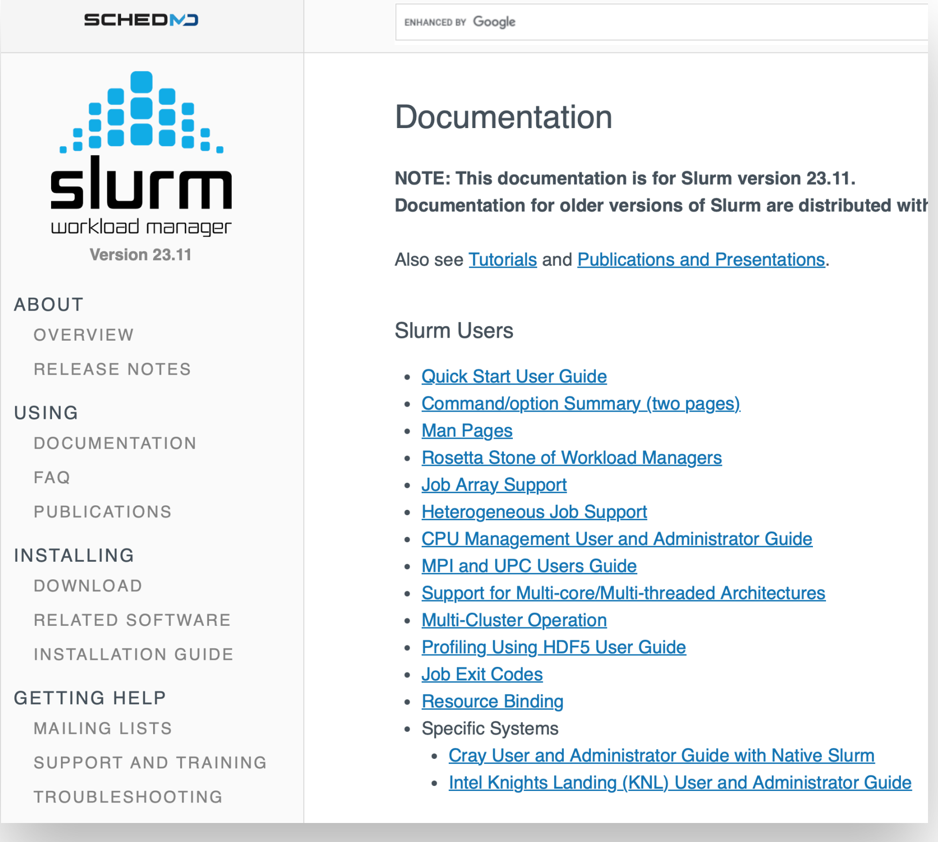
LM

Language models: the new "search engines"

How can we make LMs better information-seeking tools?

1. The ability to find and utilize reference materials

Document-augmented

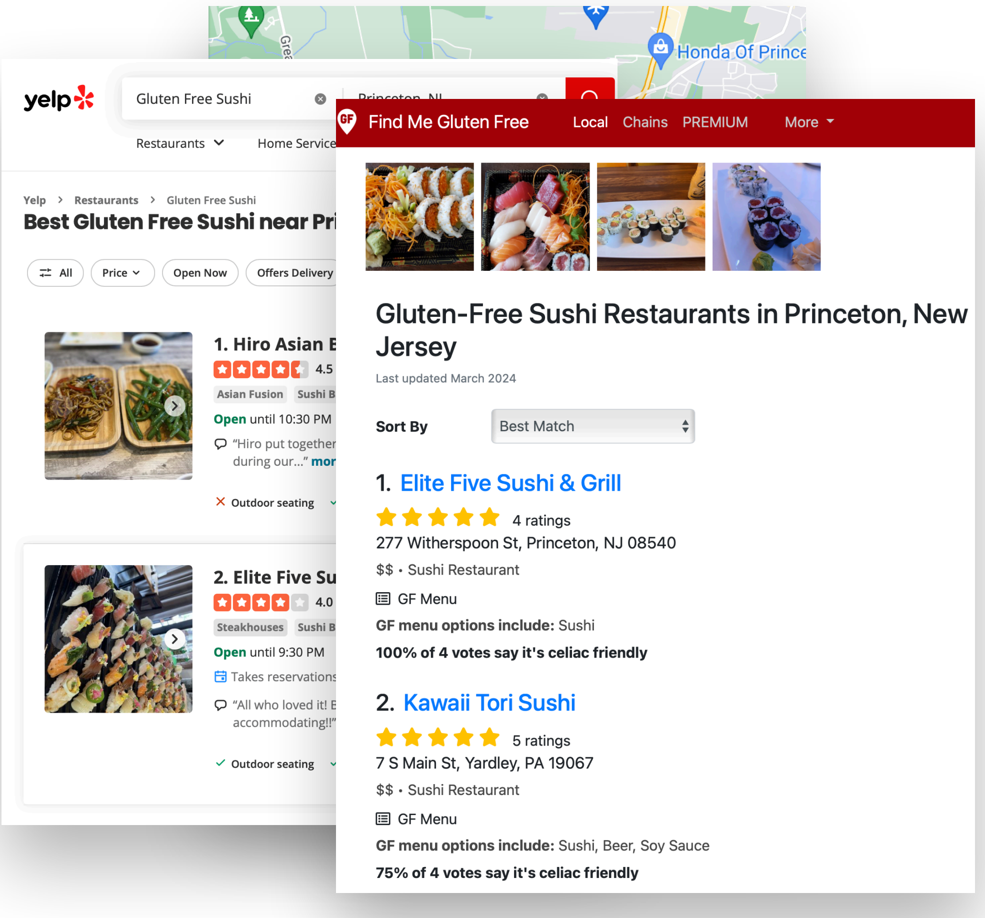


Q: In SLURM how can i restrict …

↓

LM

Retrieval-augmented



Q: Where can I find gluten-free sushi …

↓

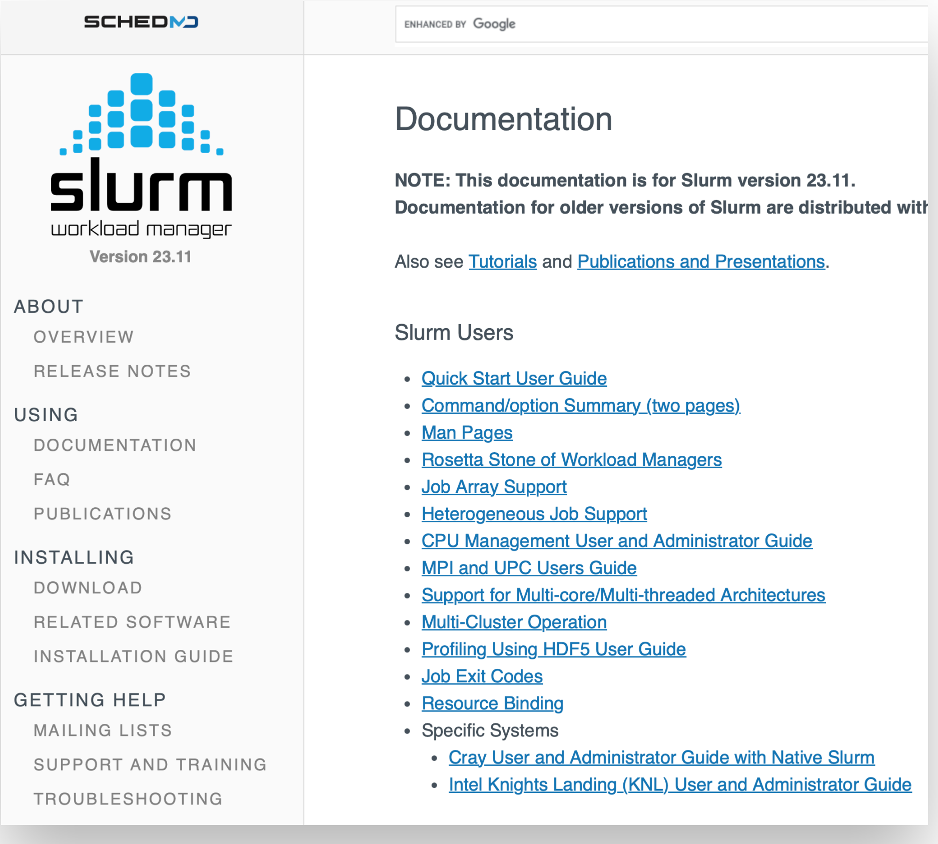
LM

Language models: the new "search engines"

How can we make LMs better information-seeking tools?

1. The ability to find and utilize reference materials

Document-augmented

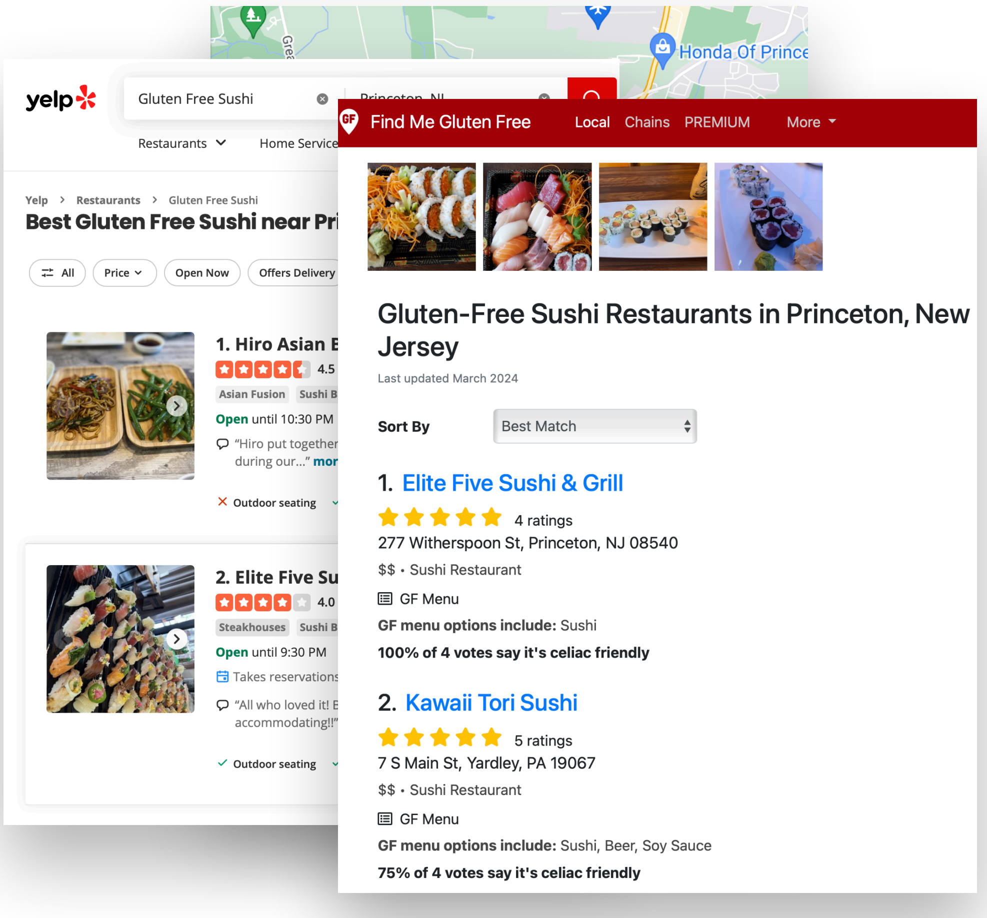


Q: In SLURM how can i restrict …

↓

LM

Retrieval-augmented



Q: Where can I find gluten-free sushi …

↓

LM

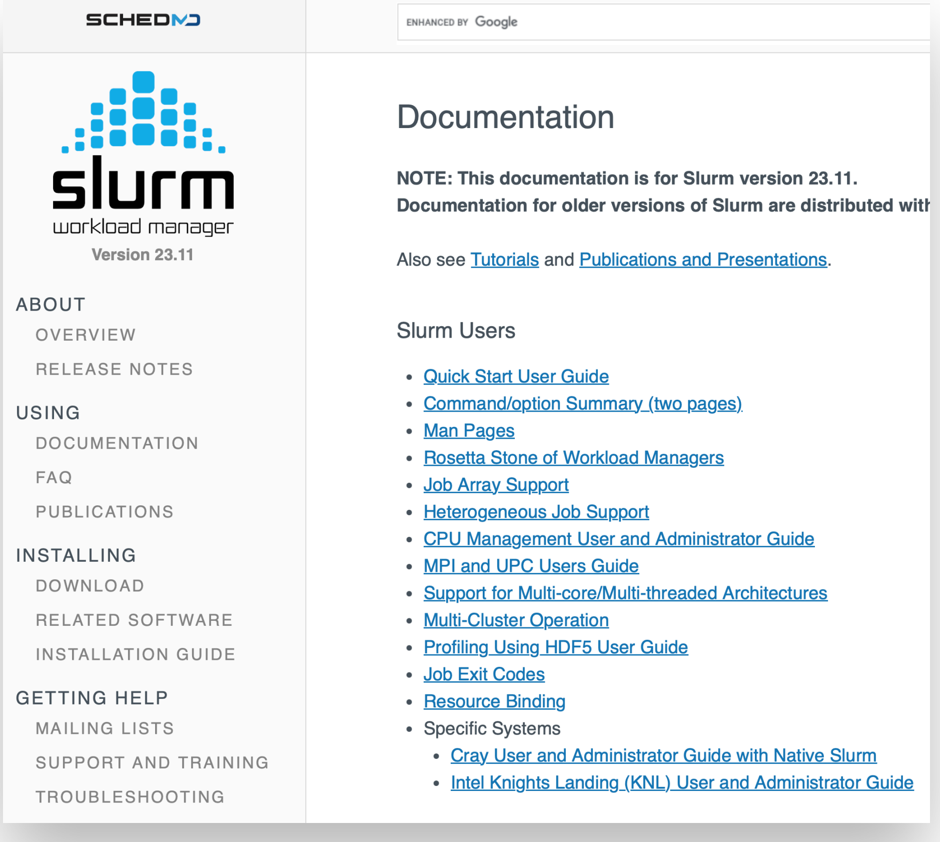
2. The ability to provide "citations"

Language models: the new “search engines”

How can we make LMs better information-seeking tools?

1. The ability to find and utilize reference materials

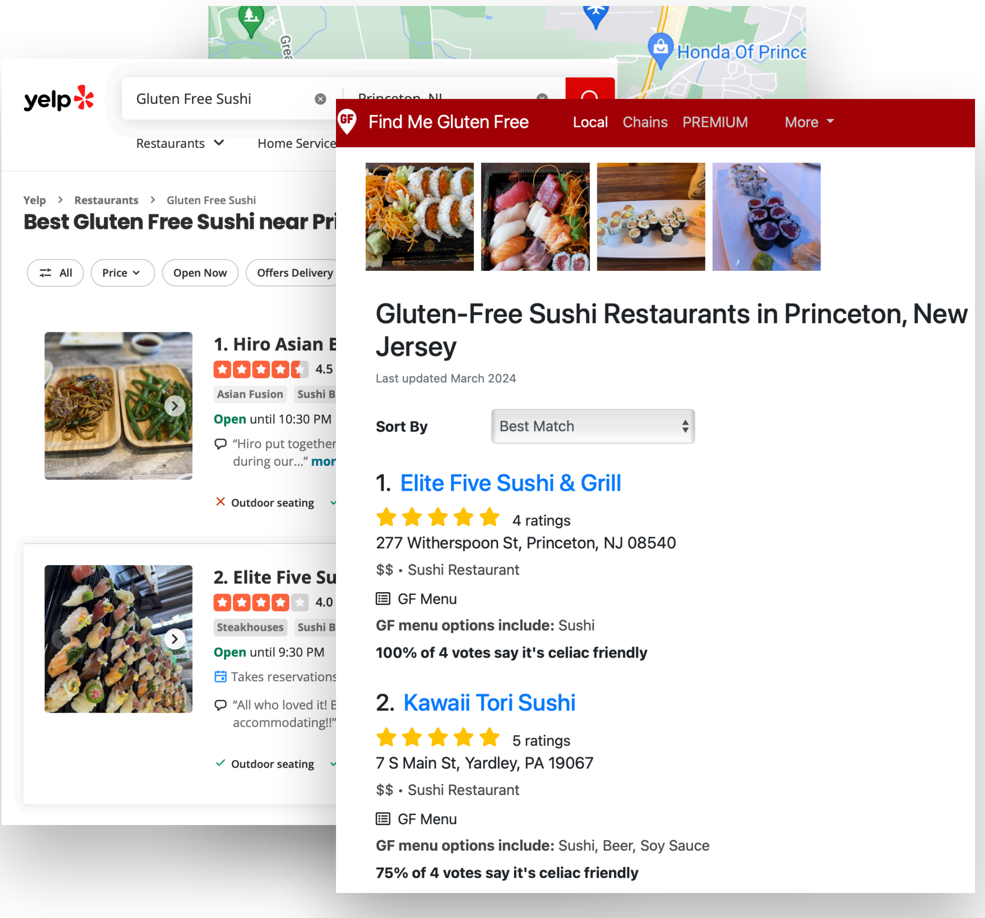
Document-augmented



Q: In SLURM how can i restrict …

LM

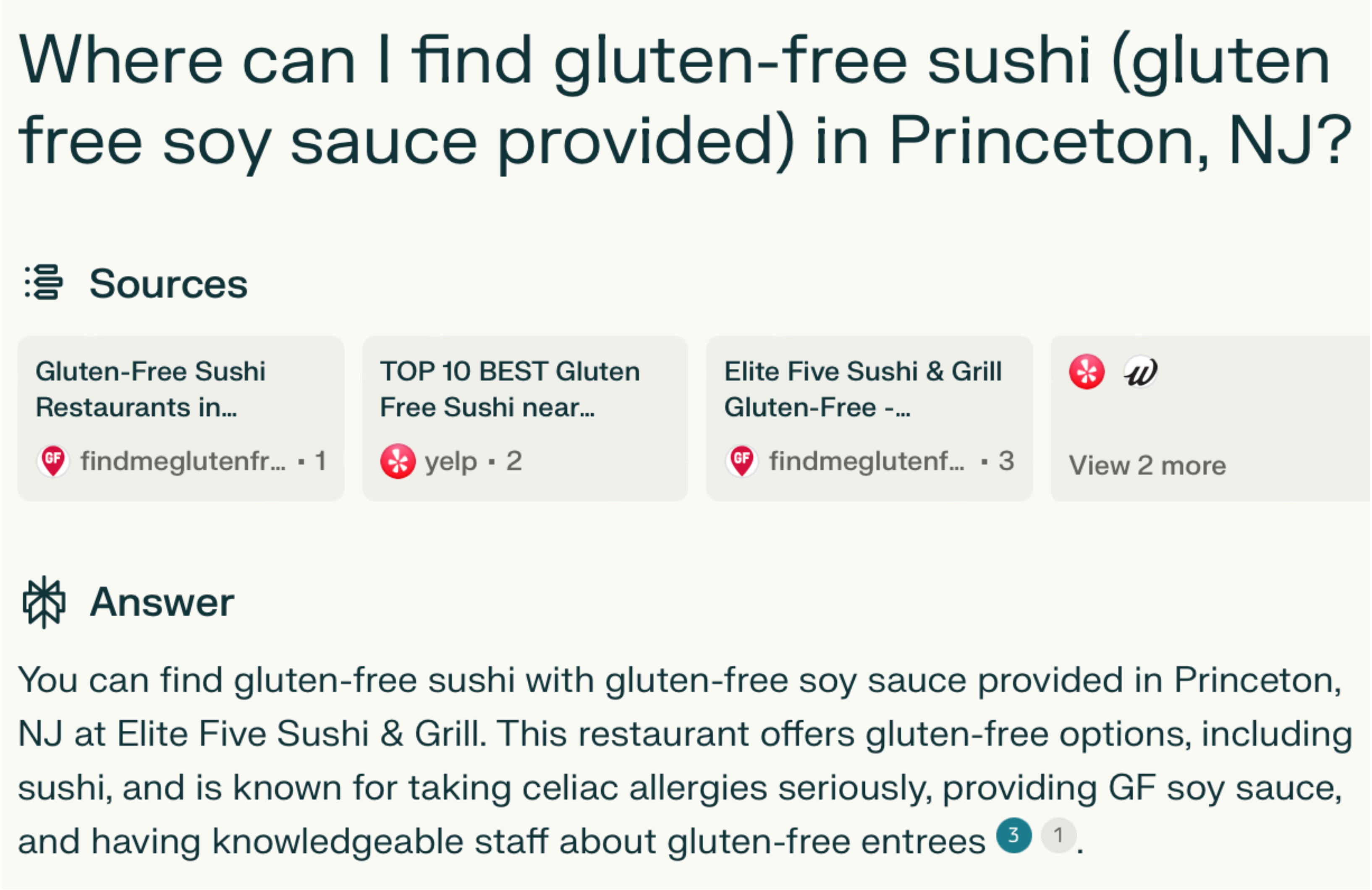
Retrieval-augmented



Q: Where can I find gluten-free sushi …

LM

2. The ability to provide “citations”



Language models: the new “search engines”

Language models: the new “search engines”

Challenge #1: the need for better retrieval/search

Language models: the new "search engines"

Challenge #1: the need for better retrieval/search

Traditional word-matching methods do not work well for "semantic search"

Language models: the new “search engines”

Challenge #1: the need for better retrieval/search

Traditional word-matching methods do not work well for “semantic search”

Query
Pre-trained language models are great.

BERT obtains new SoTA results on 11 NLP tasks. (Devlin et al., 2019)

Language models: the new “search engines”

Challenge #1: the need for better retrieval/search

Traditional word-matching methods do not work well for “semantic search”

Query
Pre-trained language models are great.

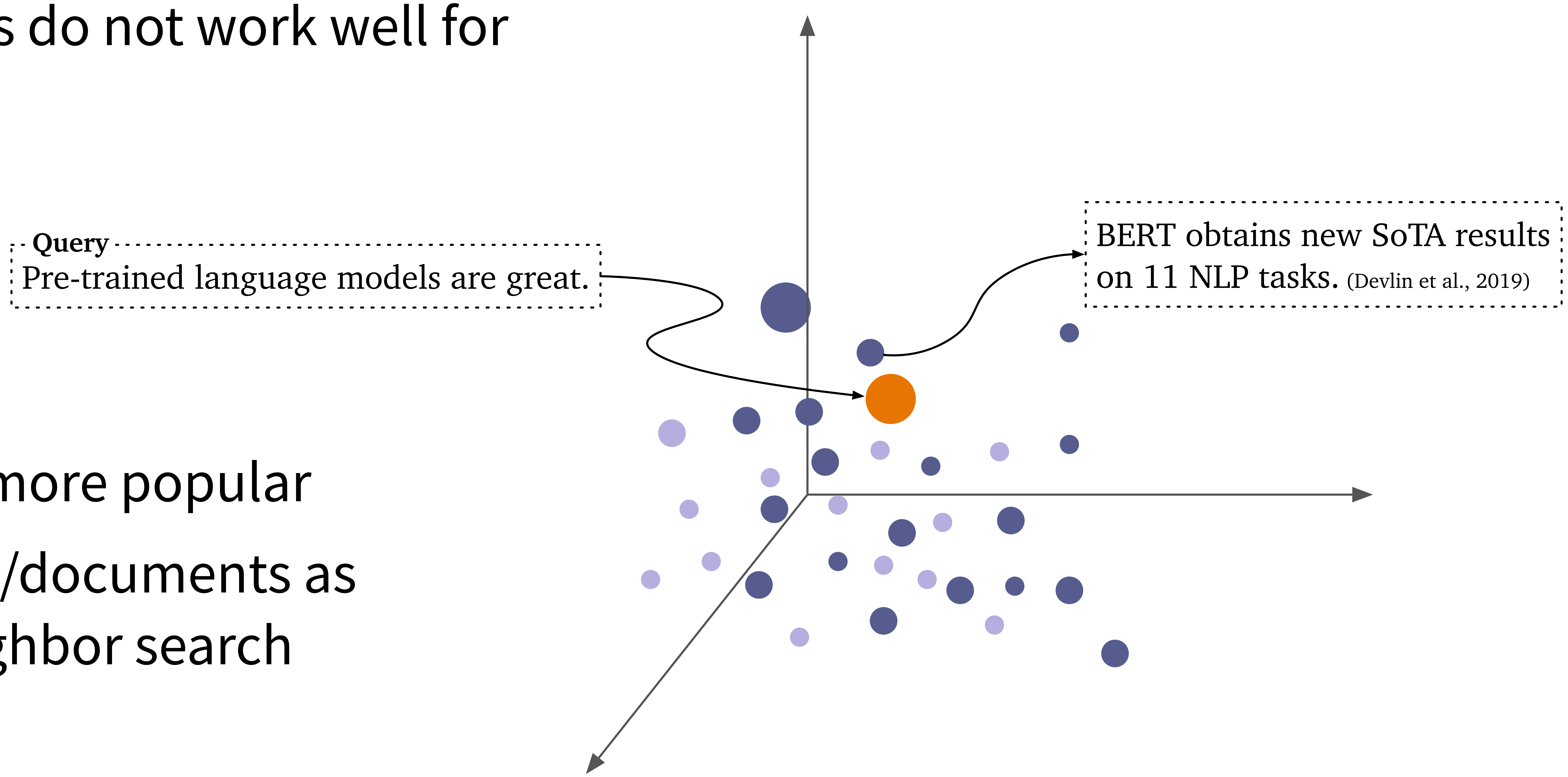
BERT obtains new SoTA results on 11 NLP tasks. (Devlin et al., 2019)

Dense retrieval becomes more and more popular

Language models: the new "search engines"

Challenge #1: the need for better retrieval/search

Traditional word-matching methods do not work well for "semantic search"



Dense retrieval becomes more and more popular

- Represent sentences/paragraphs/documents as *vectors* and perform nearest-neighbor search

Language models: the new “search engines”

Challenge #1: the need for better retrieval/search

Language models: the new "search engines"

Challenge #1: the need for better retrieval/search

SimCSE: Simple Contrastive Learning of Sentence Embeddings   

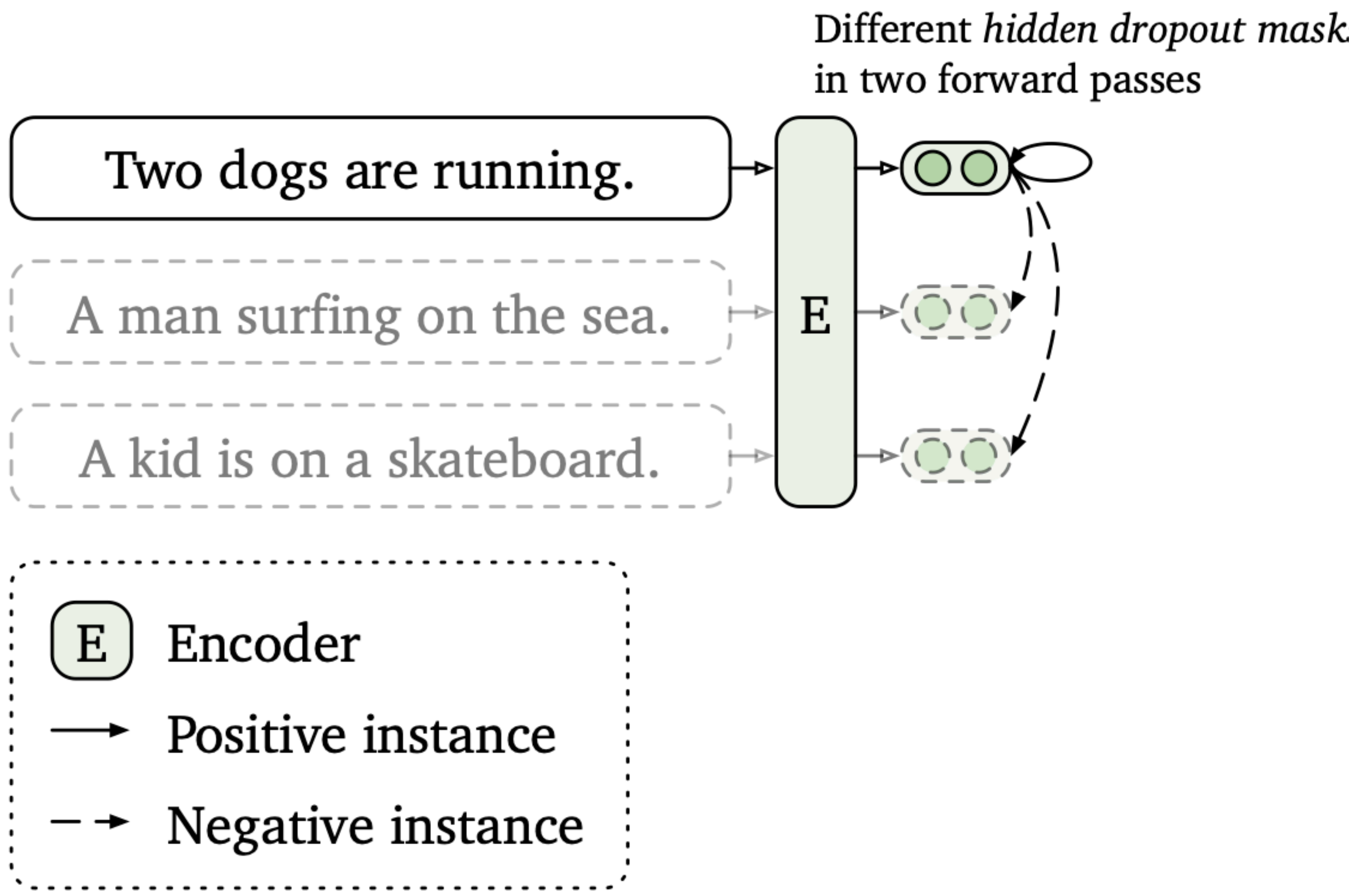
EMNLP 2021

Language models: the new "search engines"

Challenge #1: the need for better retrieval/search

SimCSE: Simple Contrastive Learning of Sentence Embeddings
EMNLP 2021

- Propose a simple *contrastive learning* framework for sentence embeddings

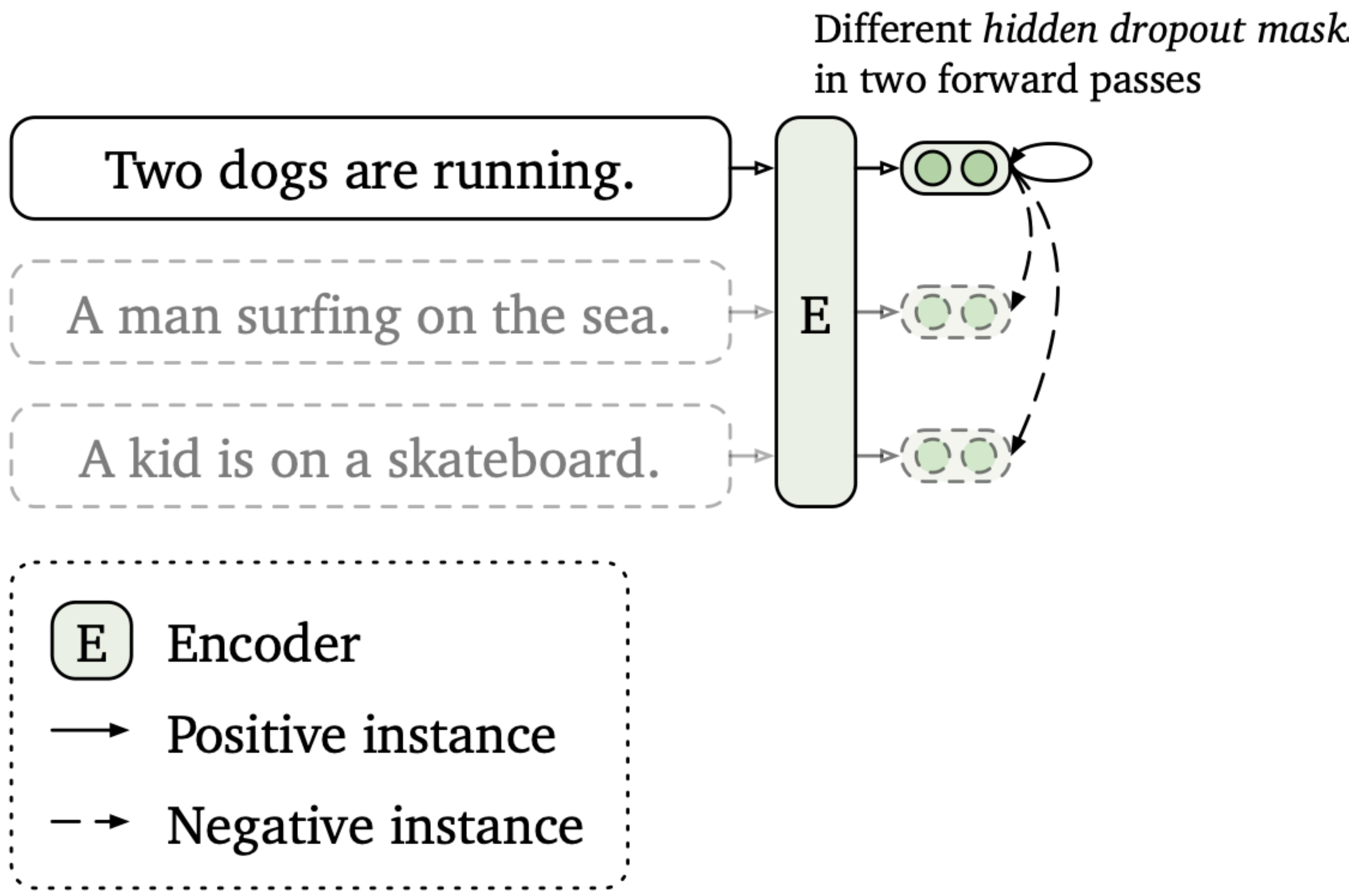


Language models: the new "search engines"

Challenge #1: the need for better retrieval/search

SimCSE: Simple Contrastive Learning of Sentence Embeddings
EMNLP 2021

- Propose a simple *contrastive learning* framework for sentence embeddings
    - A technique used by SoTA embedding tools (OpenAI, 2022; Su et al., 2023; Muennighoff et al., 2024)



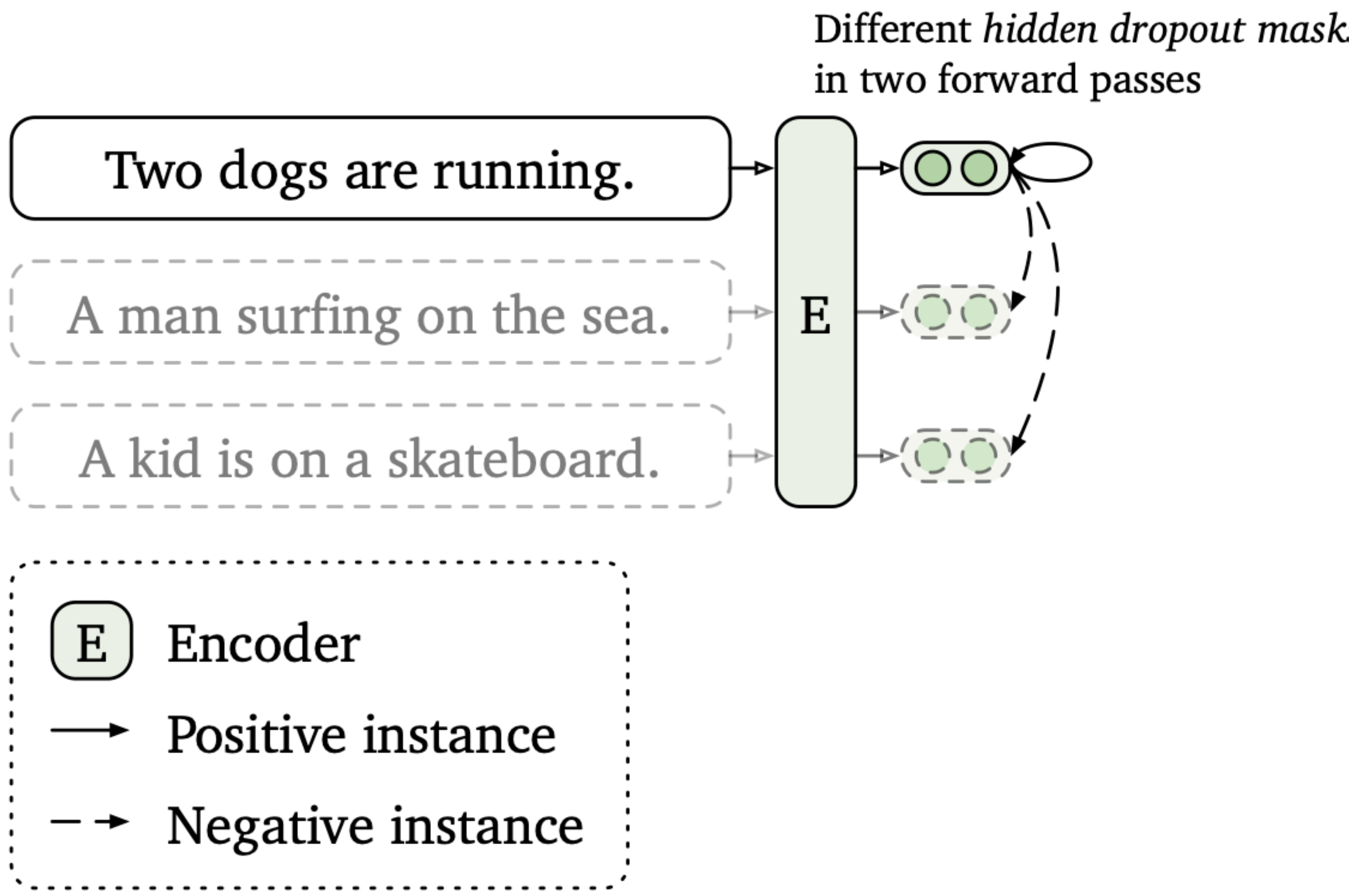
OpenAI, 2022. Text and Code Embeddings by Contrastive Pre-Training.
Su et al., 2023. One Embedder, Any Task: Instruction-Finetuned Text Embeddings.
Muennighoff et al., 2024. Generative Representational Instruction Tuning.

Language models: the new “search engines”

Challenge #1: the need for better retrieval/search

SimCSE: Simple Contrastive Learning of Sentence Embeddings
EMNLP 2021

- Propose a simple *contrastive learning* framework for sentence embeddings
  - A technique used by SoTA embedding tools (OpenAI, 2022; Su et al., 2023; Muennighoff et al., 2024)
- Downloaded by >*18M* times (from HuggingFace)



OpenAI, 2022. Text and Code Embeddings by Contrastive Pre-Training.
Su et al., 2023. One Embedder, Any Task: Instruction-Finetuned Text Embeddings.
Muennighoff et al., 2024. Generative Representational Instruction Tuning.

Language models: the new “search engines”

Language models: the new “search engines”

Challenge #2: how to evaluate?

Language models: the new “search engines”

Challenge #2: how to evaluate?

- There is no good evaluation for long text generation

Language models: the new “search engines”

Challenge #2: how to evaluate?

- There is no good evaluation for long text generation
- There is no automatic evaluation for generations with citations

Language models: the new “search engines”

Challenge #2: how to evaluate?

- There is no good evaluation for long text generation
- There is no automatic evaluation for generations with citations

Human evaluation?

Language models: the new "search engines"

Challenge #2: how to evaluate?

- There is no good evaluation for long text generation
- There is no automatic evaluation for generations with citations

Human evaluation? Slow, costly, unreliable

Language models: the new “search engines”

Challenge #2: how to evaluate?

Language models: the new "search engines"

Challenge #2: how to evaluate?

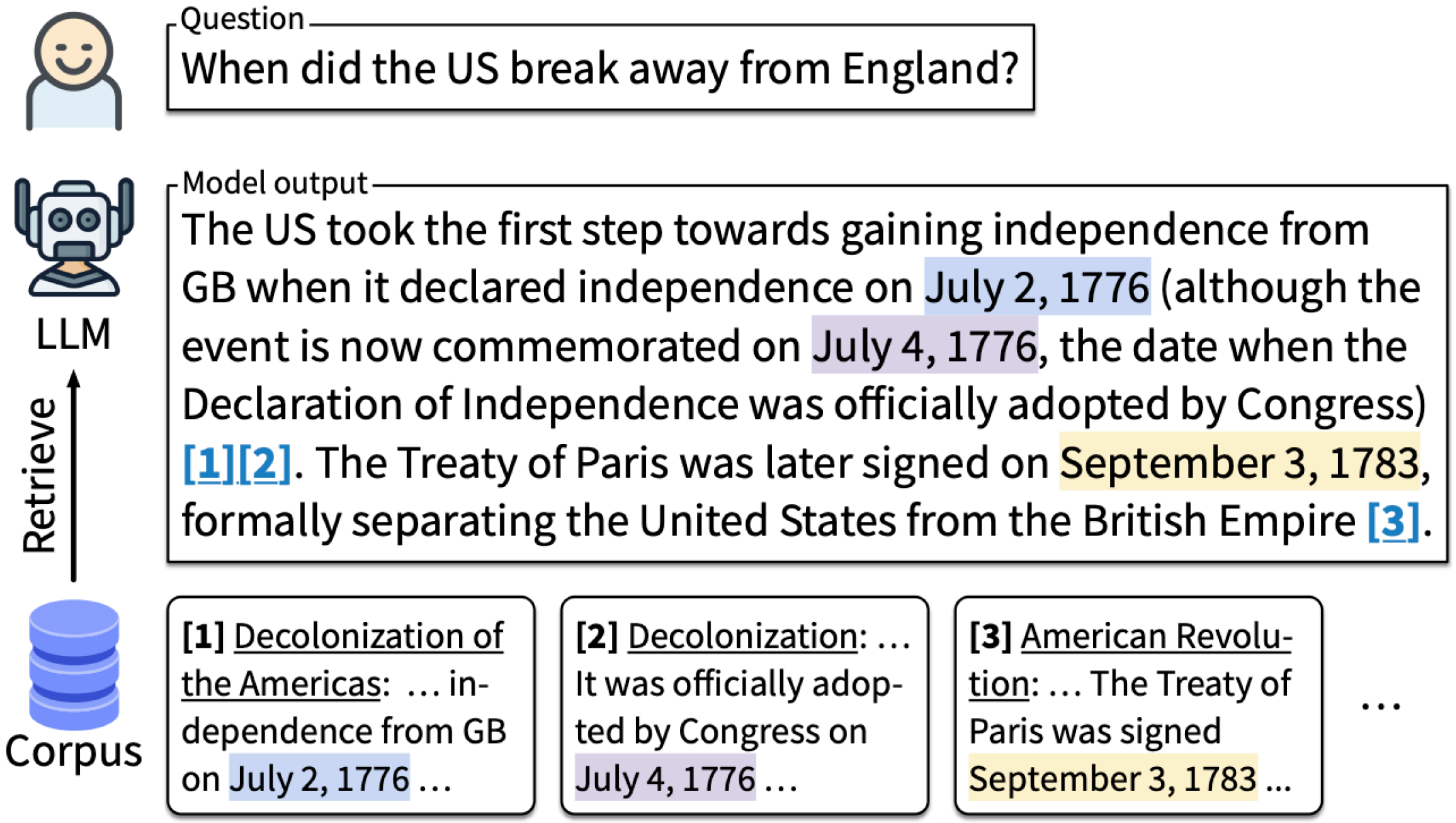
Enabling Large Language Models to Generate Text with Citations
EMNLP 2023

Language models: the new "search engines"

Challenge #2: how to evaluate?

Enabling Large Language Models to Generate Text with Citations
EMNLP 2023

ALCE: automatic LLM citation evaluation



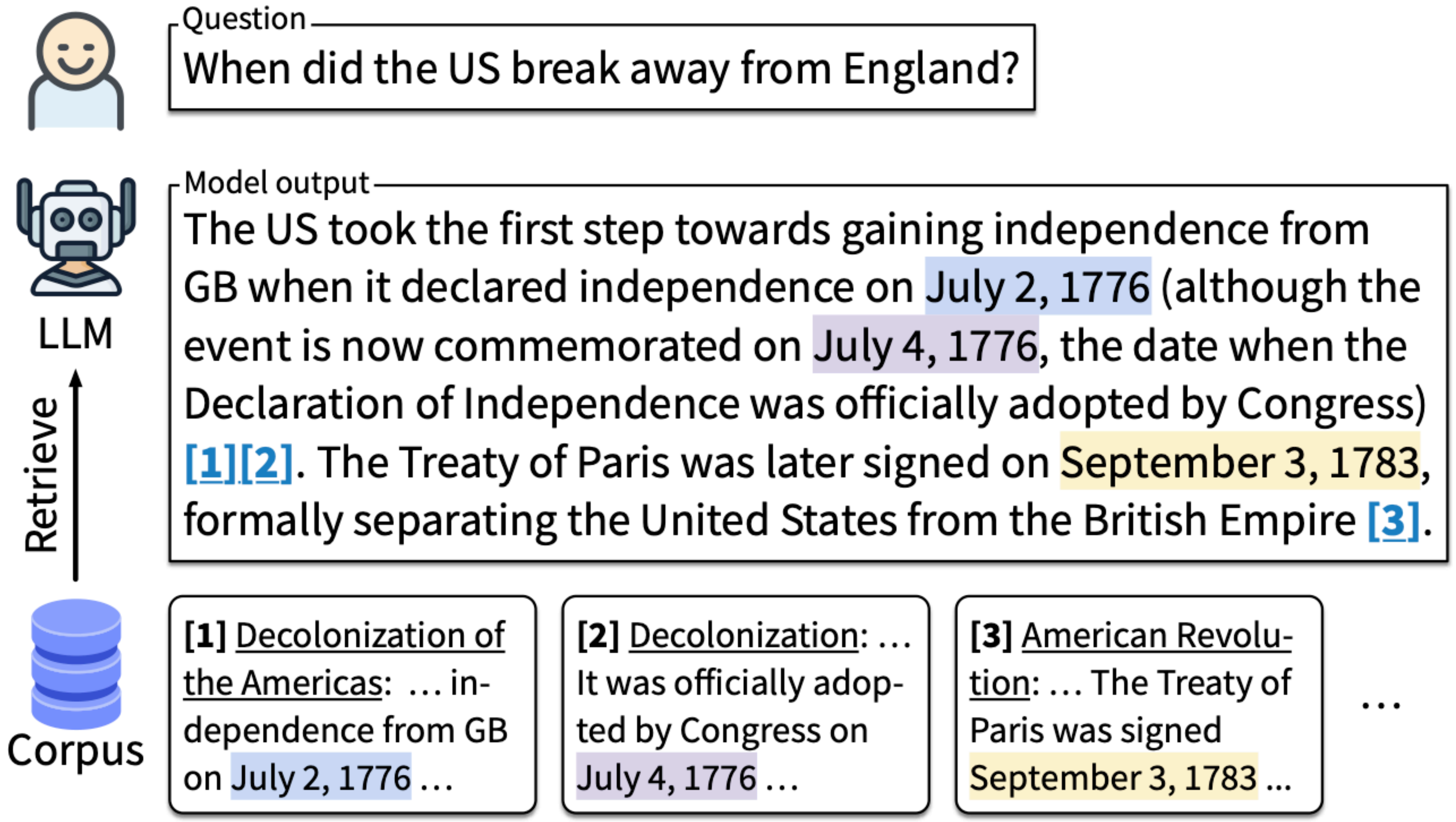
Language models: the new “search engines”

Challenge #2: how to evaluate?

Enabling Large Language Models to Generate Text with Citations
EMNLP 2023

ALCE: automatic LLM citation evaluation

- Given a **question**



Language models: the new “search engines”

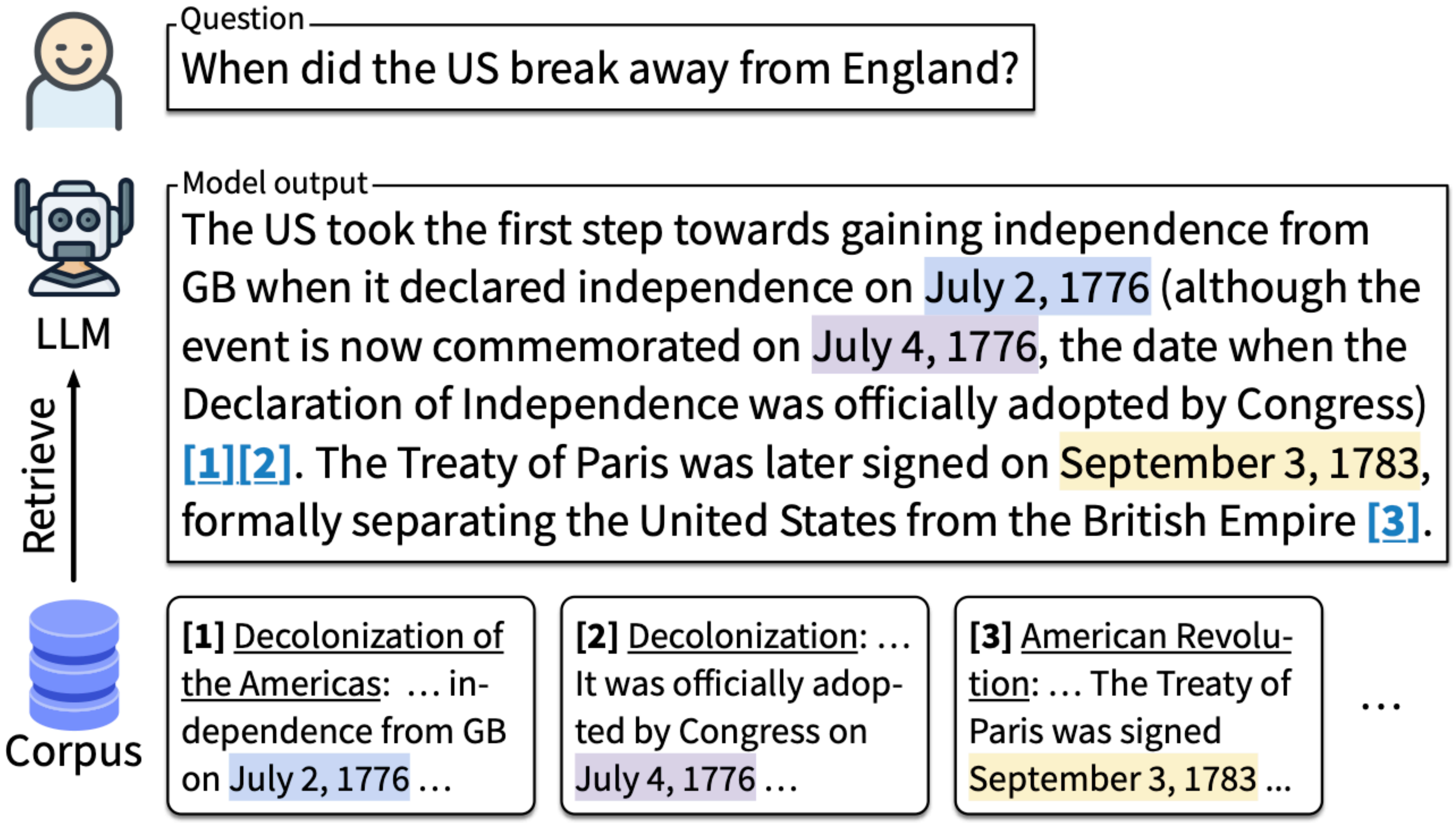
Challenge #2: how to evaluate?

Enabling Large Language Models to Generate Text with Citations
EMNLP 2023




ALCE: automatic LLM citation evaluation

- Given a **question**
- Given a **corpus**



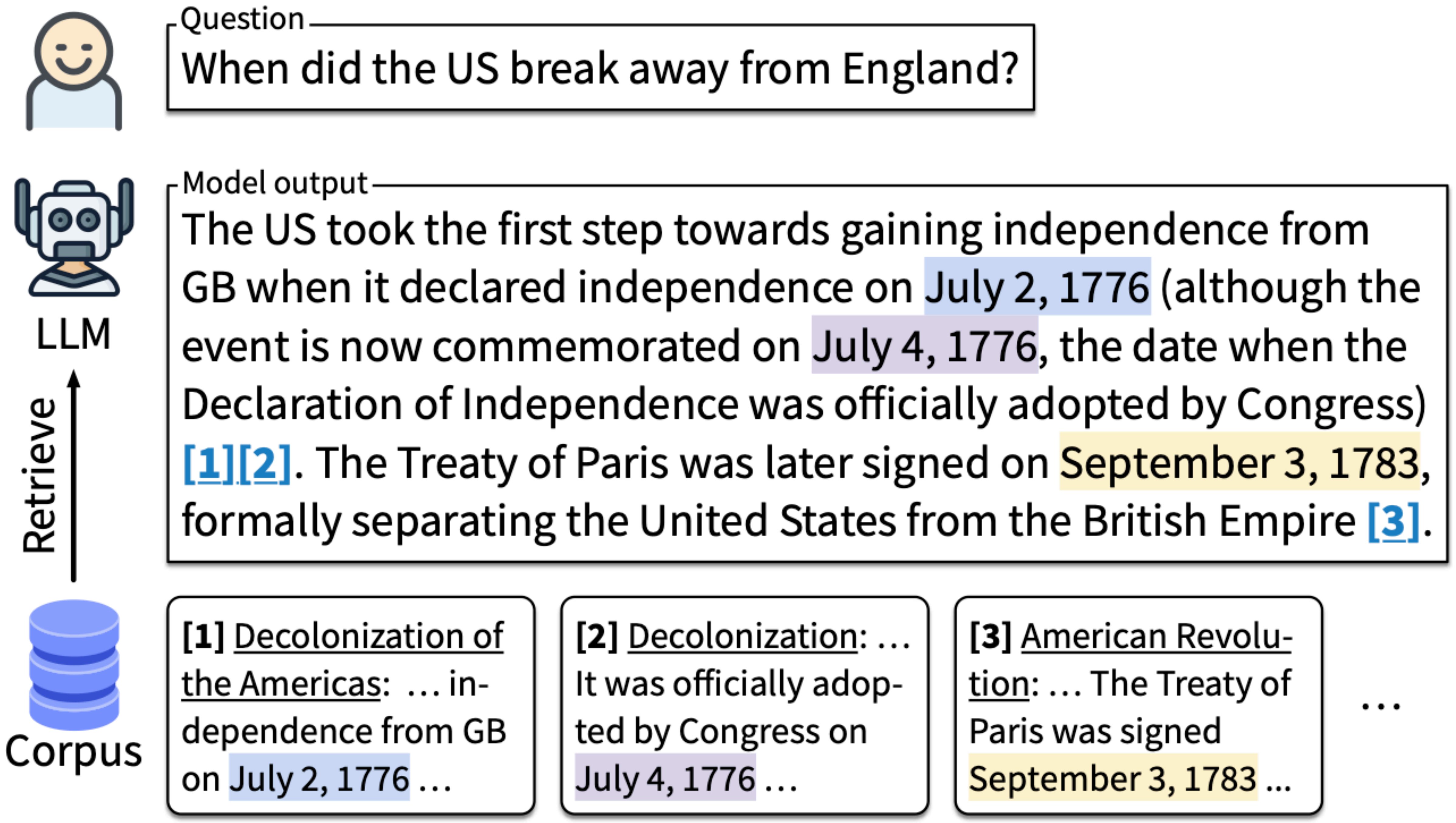
Language models: the new "search engines"

Challenge #2: how to evaluate?

Enabling Large Language Models to Generate Text with Citations
EMNLP 2023

ALCE: automatic LLM citation evaluation

- Given a **question**
- Given a **corpus**
- Requires end-to-end systems to



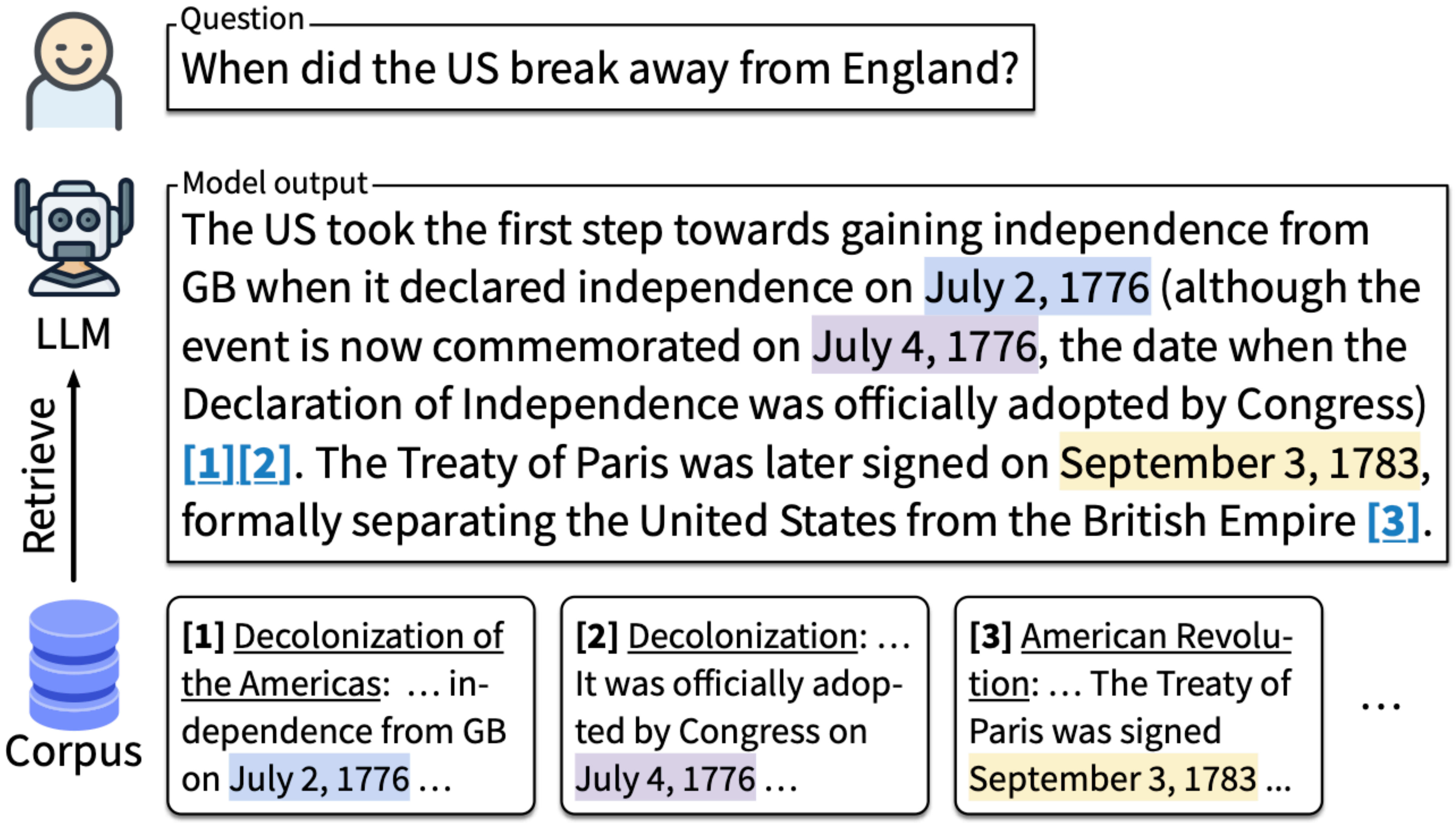
Language models: the new “search engines”

Challenge #2: how to evaluate?

Enabling Large Language Models to Generate Text with Citations
EMNLP 2023

ALCE: automatic LLM citation evaluation

- Given a **question**
- Given a **corpus**
- Requires end-to-end systems to
  - **Retrieve** passages from the corpus



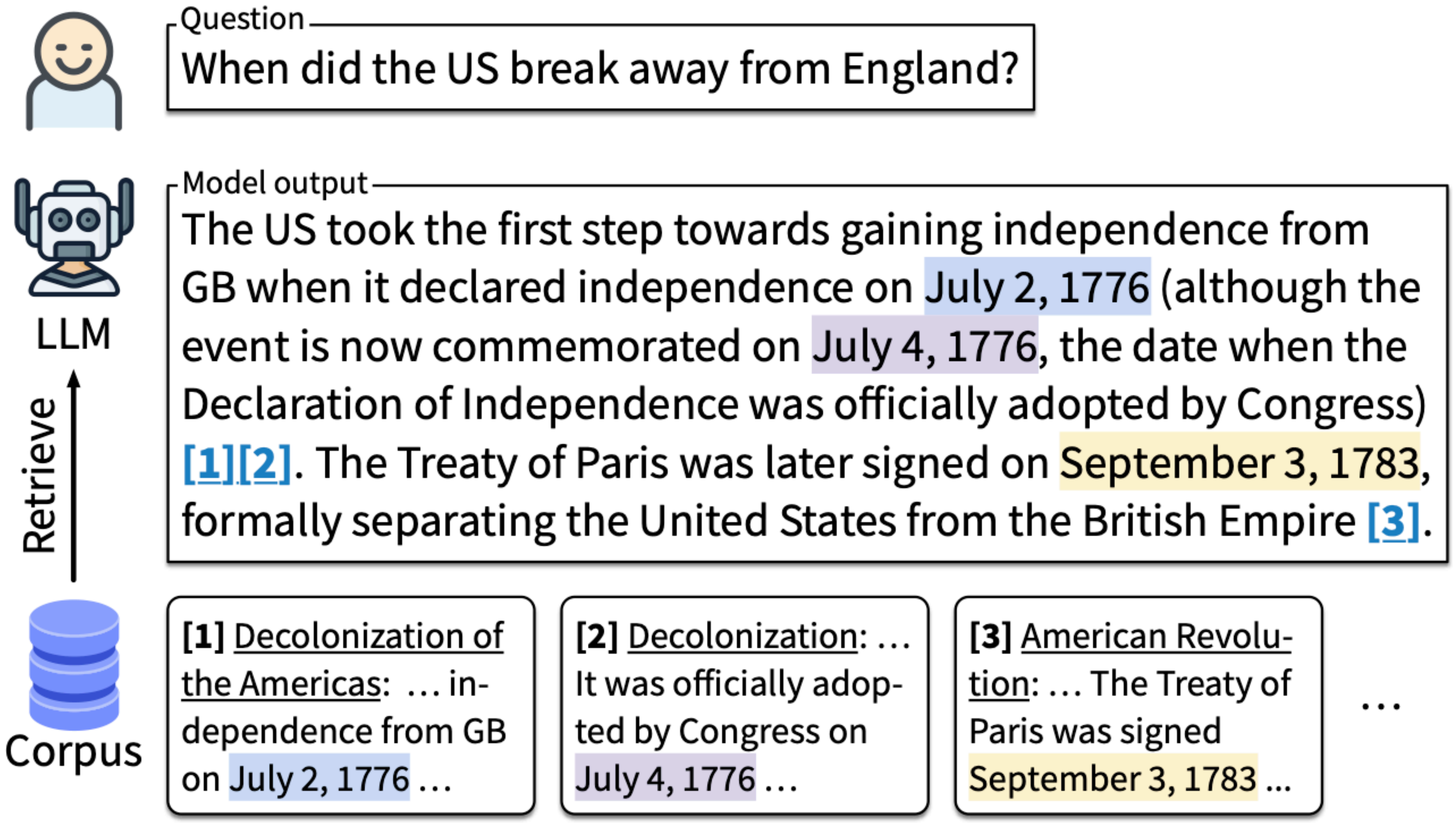
Language models: the new “search engines”

Challenge #2: how to evaluate?

Enabling Large Language Models to Generate Text with Citations
EMNLP 2023

ALCE: automatic LLM citation evaluation

- Given a **question**
- Given a **corpus**
- Requires end-to-end systems to
  - **Retrieve** passages from the corpus
  - **Generate** long-text responses



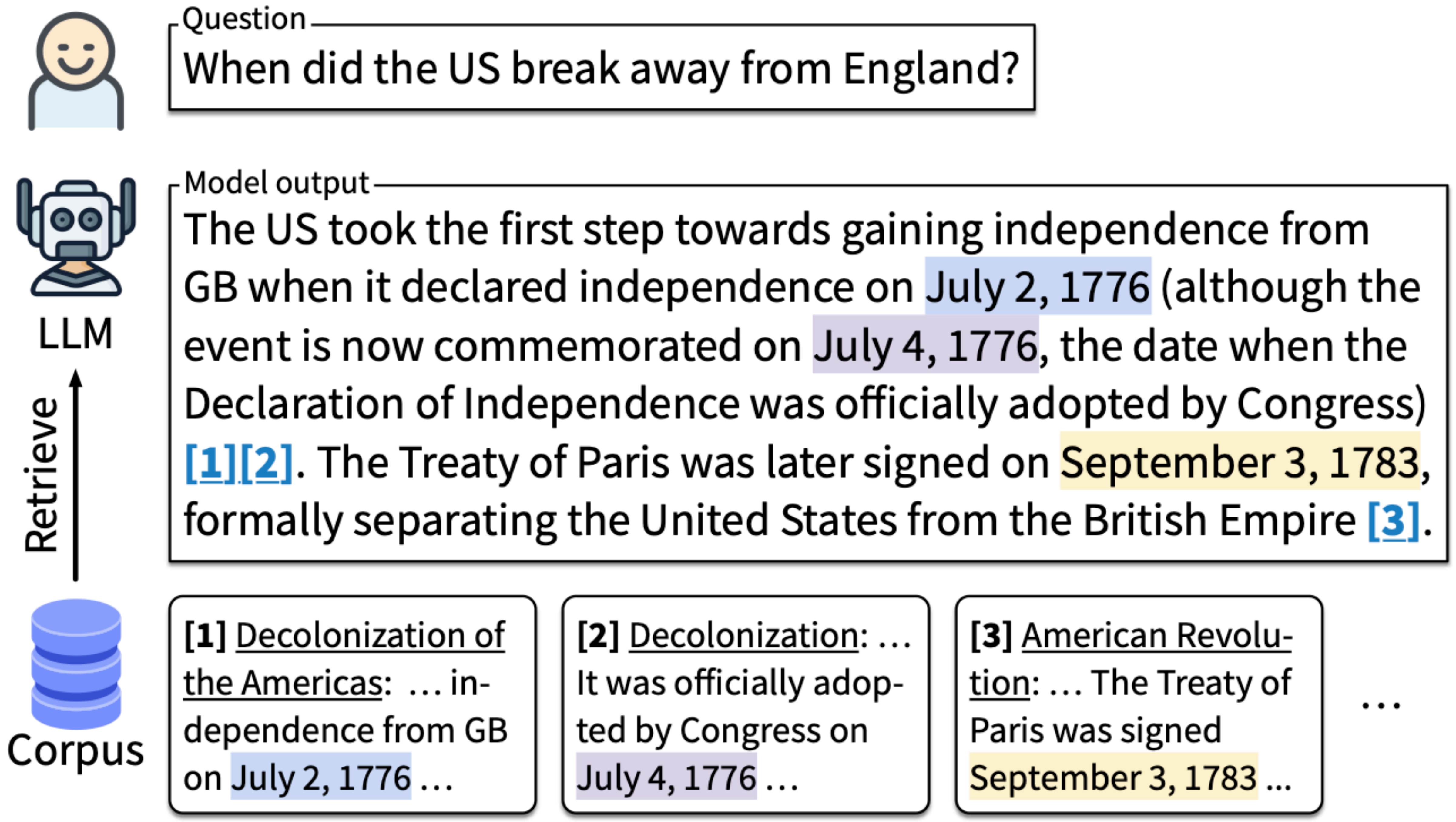
Language models: the new "search engines"

Challenge #2: how to evaluate?

Enabling Large Language Models to Generate Text with Citations
EMNLP 2023

ALCE: automatic LLM citation evaluation

- Given a **question**
- Given a **corpus**
- Requires end-to-end systems to
  - **Retrieve** passages from the corpus
  - **Generate** long-text responses
  - **Cite** supporting passages



Language models: the new "search engines"

Challenge #2: how to evaluate?

Language models: the new “search engines”

Challenge #2: how to evaluate?

ALCE automatically evaluates a long-text answer’s

Language models: the new “search engines”

Challenge #2: how to evaluate?

ALCE automatically evaluates a long-text answer’s

- Fluency

Language models: the new "search engines"

Challenge #2: how to evaluate?

ALCE automatically evaluates a long-text answer's

- Fluency
- Correctness

Language models: the new “search engines”

Challenge #2: how to evaluate?

ALCE automatically evaluates a long-text answer’s

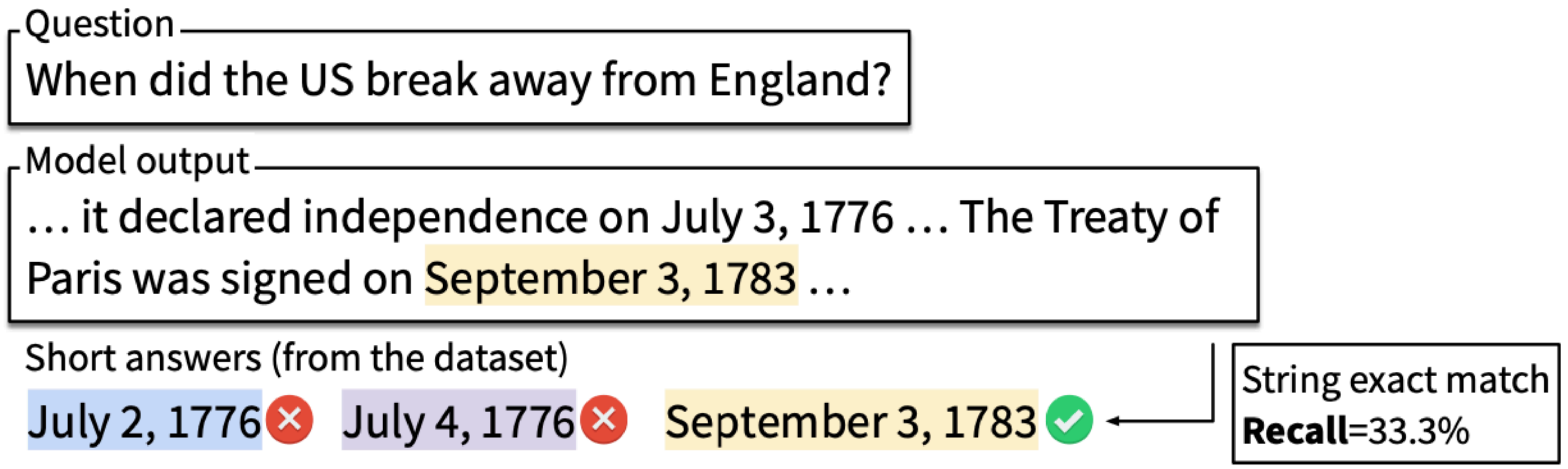
- Fluency
- Correctness
- Citation quality

Language models: the new "search engines"

Challenge #2: how to evaluate?

ALCE automatically evaluates a long-text answer's

- Fluency
- Correctness
- Citation quality



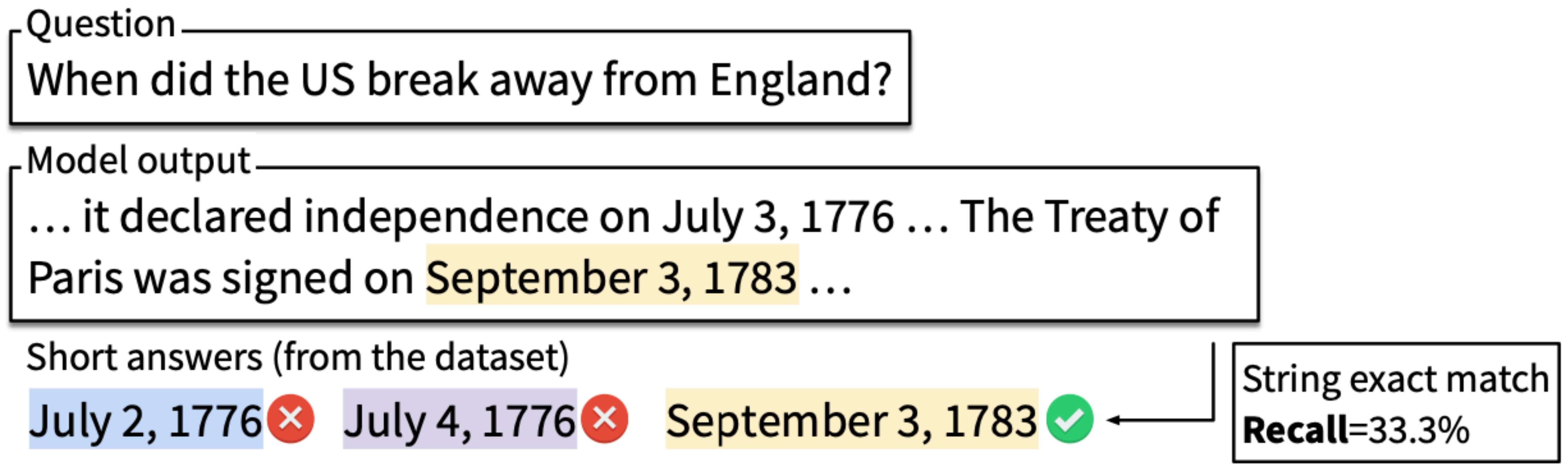
An example for correctness evaluation.

Language models: the new "search engines"

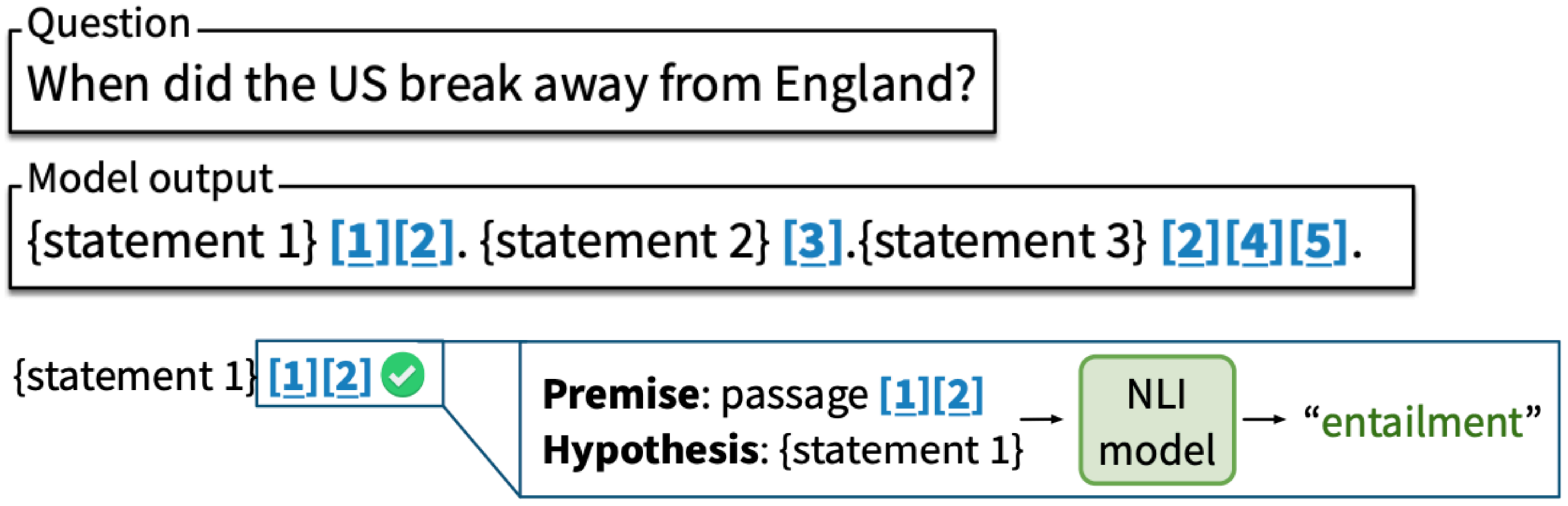
Challenge #2: how to evaluate?

ALCE automatically evaluates a long-text answer's

- Fluency
- Correctness
- Citation quality



An example for correctness evaluation.



An example for citation evaluation

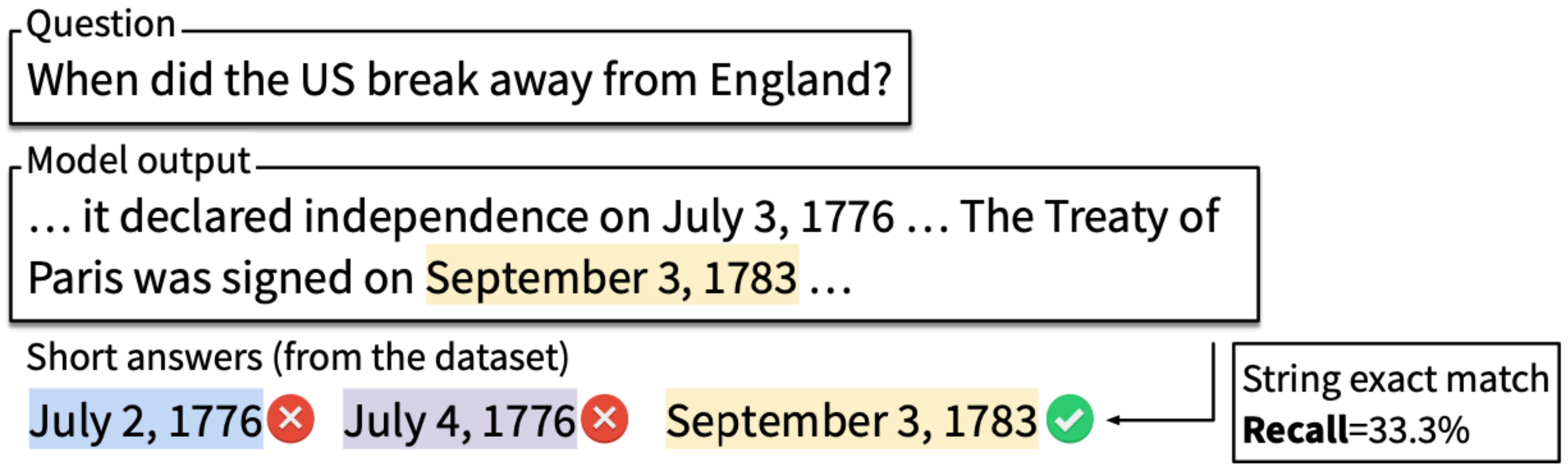
Language models: the new “search engines”

Challenge #2: how to evaluate?

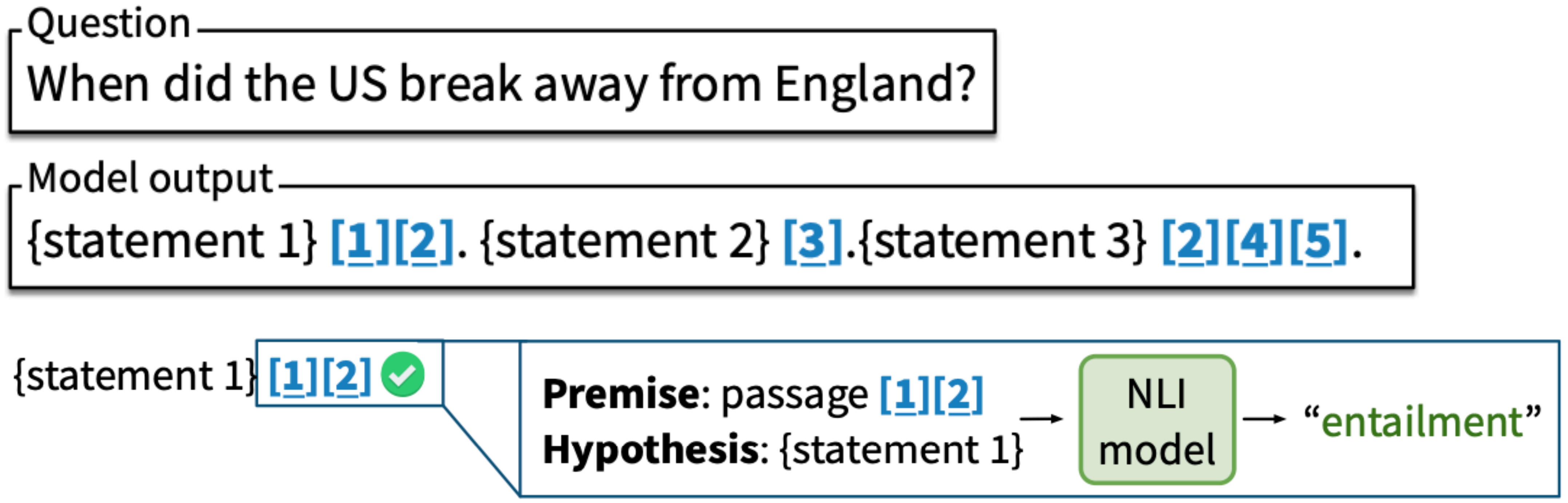
ALCE automatically evaluates a long-text answer’s

- Fluency
- Correctness
- Citation quality

We show that even **GPT-4** lacks complete citation support **50%** of the times



An example for correctness evaluation.



An example for citation evaluation

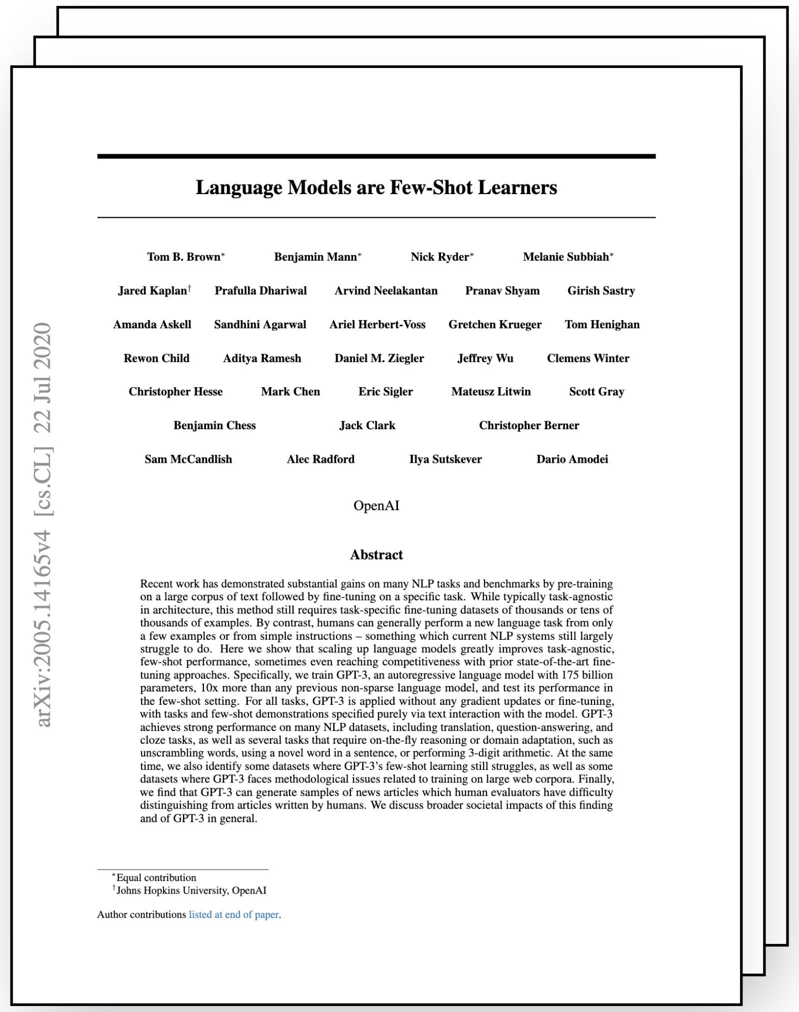
Language models: the new “search engines”

Language models: the new “search engines”

Challenge #3: how to fit the context

Language models: the new “search engines”

Challenge #3: how to fit the context


arXiv:2005.14165v4 [cs.CL] 22 Jul 2020

Language Models are Few-Shot Learners

Tom B. Brown* **Benjamin Mann*** **Nick Ryder*** **Melanie Subbiah***
Jared Kaplan† **Prafulla Dhariwal** **Arvind Neelakantan** **Pranav Shyam** **Girish Sastry**
Amanda Askell **Sandhini Agarwal** **Ariel Herbert-Voss** **Gretchen Krueger** **Tom Henighan**
Rewon Child **Aditya Ramesh** **Daniel M. Ziegler** **Jeffrey Wu** **Clemens Winter**
Christopher Hesse **Mark Chen** **Eric Sigler** **Mateusz Litwin** **Scott Gray**
Benjamin Chess **Jack Clark** **Christopher Berner**
Sam McCandlish **Alec Radford** **Ilya Sutskever** **Dario Amodei**

OpenAI

Abstract

Recent work has demonstrated substantial gains on many NLP tasks and benchmarks by pre-training on a large corpus of text followed by fine-tuning on a specific task. While typically task-agnostic in architecture, this method still requires task-specific fine-tuning datasets of thousands or tens of thousands of examples. By contrast, humans can generally perform a new language task from only a few examples or from simple instructions – something which current NLP systems still largely struggle to do. Here we show that scaling up language models greatly improves task-agnostic, few-shot performance, sometimes even reaching competitiveness with prior state-of-the-art fine-tuning approaches. Specifically, we train GPT-3, an autoregressive language model with 175 billion parameters, 10x more than any previous non-sparse language model, and test its performance in the few-shot setting. For all tasks, GPT-3 is applied without any gradient updates or fine-tuning, with tasks and few-shot demonstrations specified purely via text interaction with the model. GPT-3 achieves strong performance on many NLP datasets, including translation, question-answering, and cloze tasks, as well as several tasks that require on-the-fly reasoning or domain adaptation, such as unscrambling words, using a novel word in a sentence, or performing 3-digit arithmetic. At the same time, we also identify some datasets where GPT-3's few-shot learning still struggles, as well as some datasets where GPT-3 faces methodological issues related to training on large web corpora. Finally, we find that GPT-3 can generate samples of news articles which human evaluators have difficulty distinguishing from articles written by humans. We discuss broader societal impacts of this finding and of GPT-3 in general.

*Equal contribution
†Johns Hopkins University, OpenAI

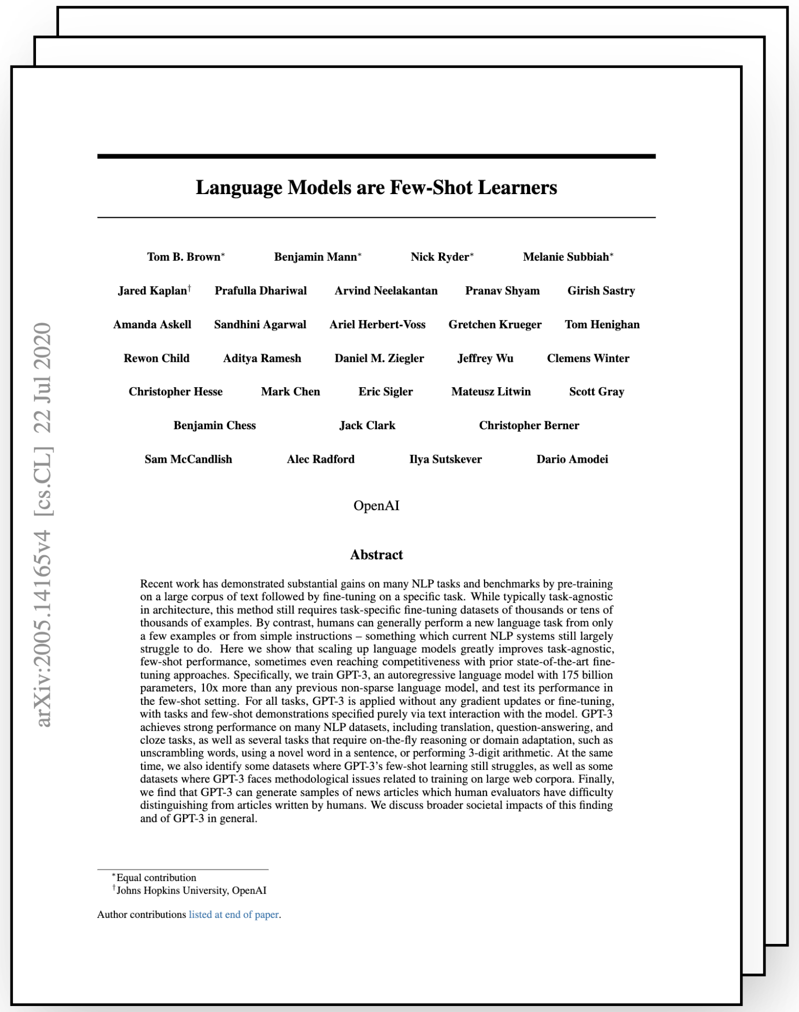
Author contributions listed at end of paper.

The GPT-3 paper

(~**75K** tokens)

Language models: the new "search engines"

Challenge #3: how to fit the context


arXiv:2005.14165v4 [cs.CL] 22 Jul 2020

Language Models are Few-Shot Learners

Tom B. Brown* **Benjamin Mann*** **Nick Ryder*** **Melanie Subbiah***
Jared Kaplan† **Prafulla Dhariwal** **Arvind Neelakantan** **Pranav Shyam** **Girish Sastry**
Amanda Askell **Sandhini Agarwal** **Ariel Herbert-Voss** **Gretchen Krueger** **Tom Henighan**
Rewon Child **Aditya Ramesh** **Daniel M. Ziegler** **Jeffrey Wu** **Clemens Winter**
Christopher Hesse **Mark Chen** **Eric Sigler** **Mateusz Litwin** **Scott Gray**
Benjamin Chess **Jack Clark** **Christopher Berner**
Sam McCandlish **Alec Radford** **Ilya Sutskever** **Dario Amodei**

OpenAI

Abstract

Recent work has demonstrated substantial gains on many NLP tasks and benchmarks by pre-training on a large corpus of text followed by fine-tuning on a specific task. While typically task-agnostic in architecture, this method still requires task-specific fine-tuning datasets of thousands or tens of thousands of examples. By contrast, humans can generally perform a new language task from only a few examples or from simple instructions – something which current NLP systems still largely struggle to do. Here we show that scaling up language models greatly improves task-agnostic, few-shot performance, sometimes even reaching competitiveness with prior state-of-the-art fine-tuning approaches. Specifically, we train GPT-3, an autoregressive language model with 175 billion parameters, 10x more than any previous non-sparse language model, and test its performance in the few-shot setting. For all tasks, GPT-3 is applied without any gradient updates or fine-tuning, with tasks and few-shot demonstrations specified purely via text interaction with the model. GPT-3 achieves strong performance on many NLP datasets, including translation, question-answering, and cloze tasks, as well as several tasks that require on-the-fly reasoning or domain adaptation, such as unscrambling words, using a novel word in a sentence, or performing 3-digit arithmetic. At the same time, we also identify some datasets where GPT-3's few-shot learning still struggles, as well as some datasets where GPT-3 faces methodological issues related to training on large web corpora. Finally, we find that GPT-3 can generate samples of news articles which human evaluators have difficulty distinguishing from articles written by humans. We discuss broader societal impacts of this finding and of GPT-3 in general.

*Equal contribution
†Johns Hopkins University, OpenAI

Author contributions listed at end of paper.

The GPT-3 paper

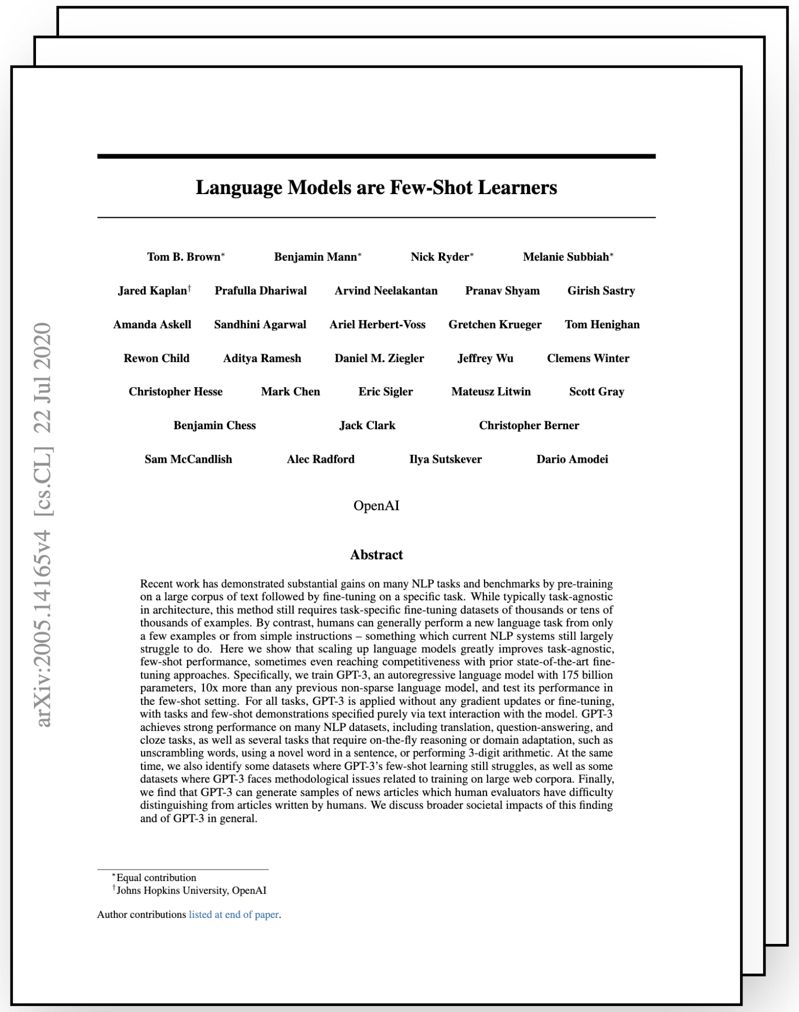
(~**75K** tokens)

The Dune series

(~**1M** words)

Language models: the new "search engines"

Challenge #3: how to fit the context


Language Models are Few-Shot Learners

Tom B. Brown* Benjamin Mann* Nick Ryder* Melanie Subbiah*
Jared Kaplan† Prafulla Dhariwal Arvind Neelakantan Pranav Shyam Girish Sastry
Amanda Askell Sandhini Agarwal Ariel Herbert-Voss Gretchen Krueger Tom Henighan
Rewon Child Aditya Ramesh Daniel M. Ziegler Jeffrey Wu Clemens Winter
Christopher Hesse Mark Chen Eric Sigler Mateusz Litwin Scott Gray
Benjamin Chess Jack Clark Christopher Berner
Sam McCandlish Alec Radford Ilya Sutskever Dario Amodei

OpenAI

Abstract

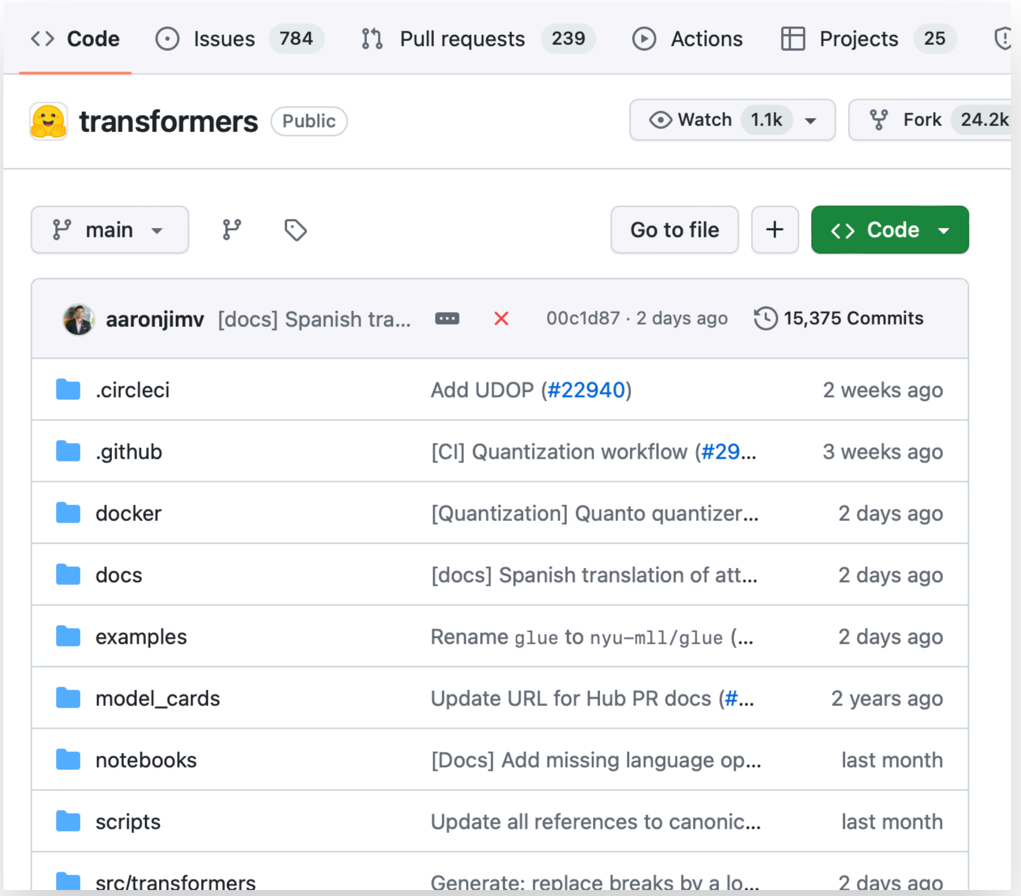
arXiv:2005.14165v4 [cs.CL] 22 Jul 2020

The GPT-3 paper

(~**75K** tokens)

The Dune series

(~**1M** words)

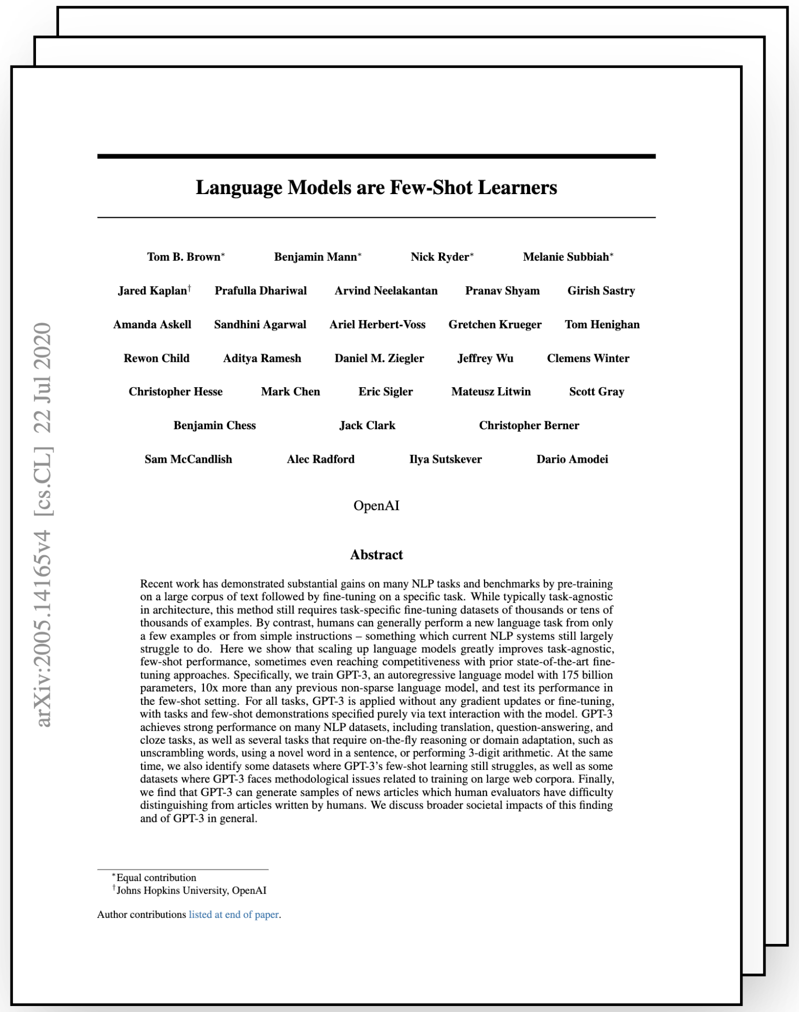


The Transformers package

(~**10M** tokens)

Language models: the new "search engines"

Challenge #3: how to fit the context

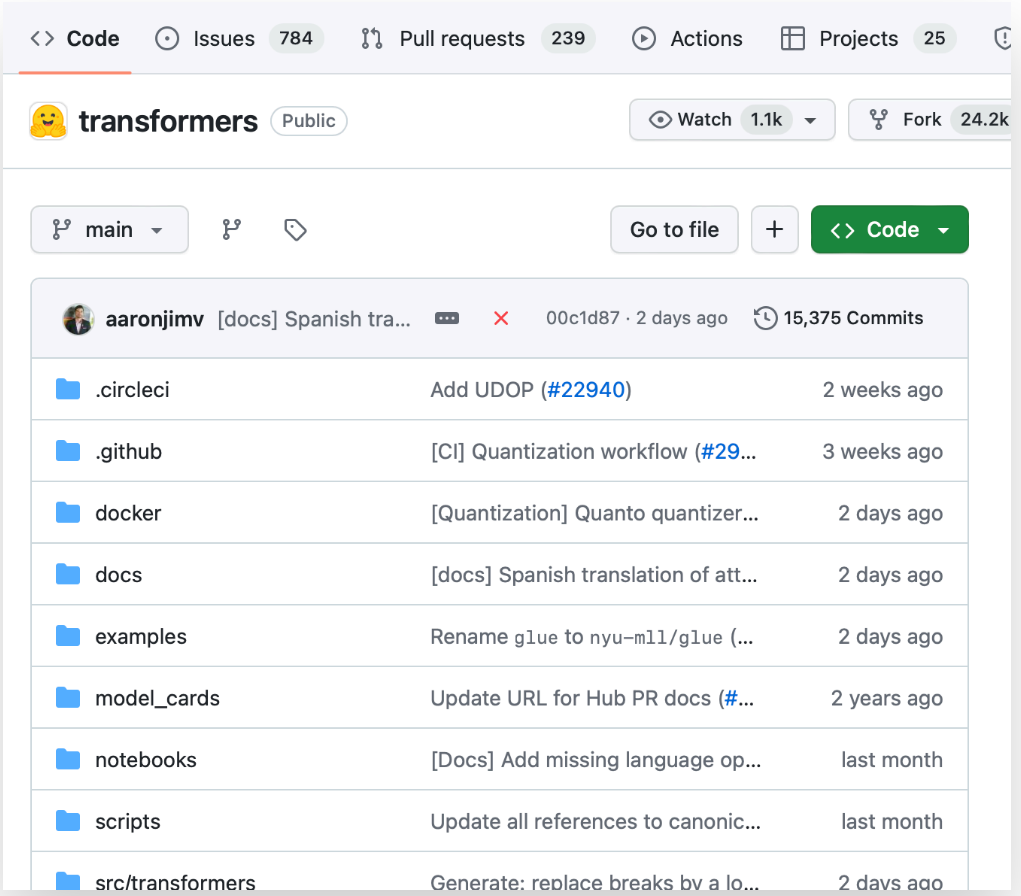


The GPT-3 paper

(~**75K** tokens)

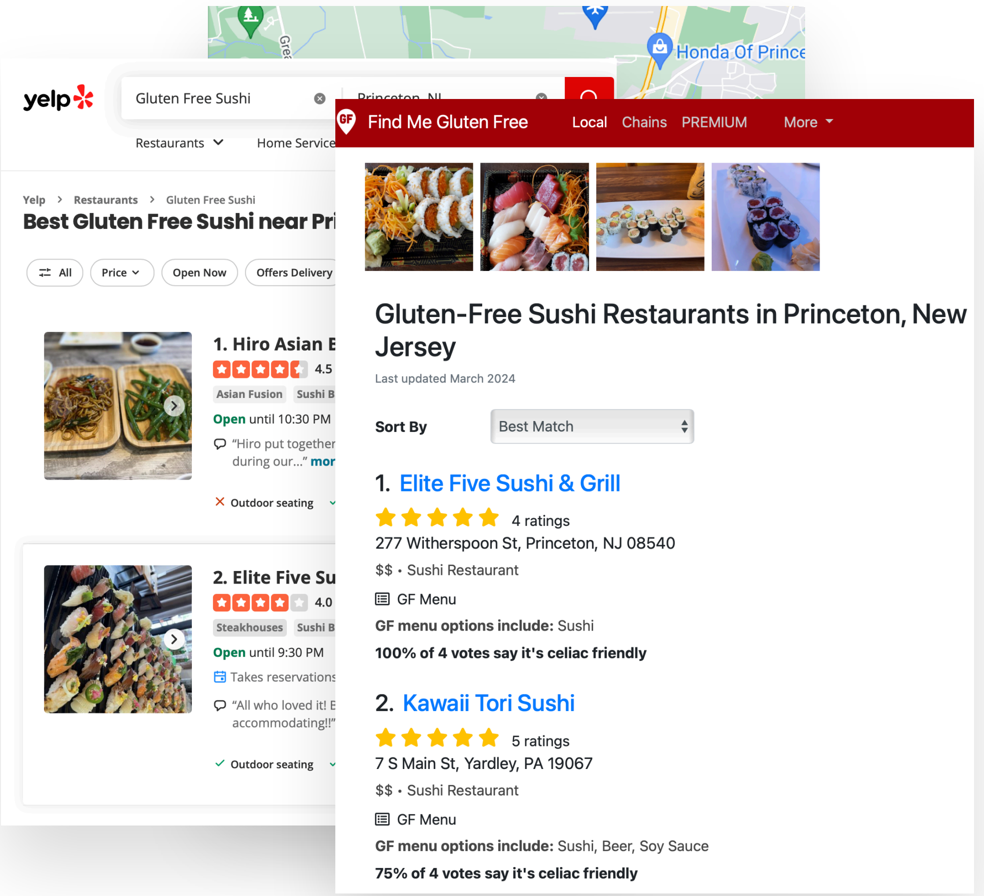
The Dune series

(~**1M** words)



The Transformers package

(~**10M** tokens)

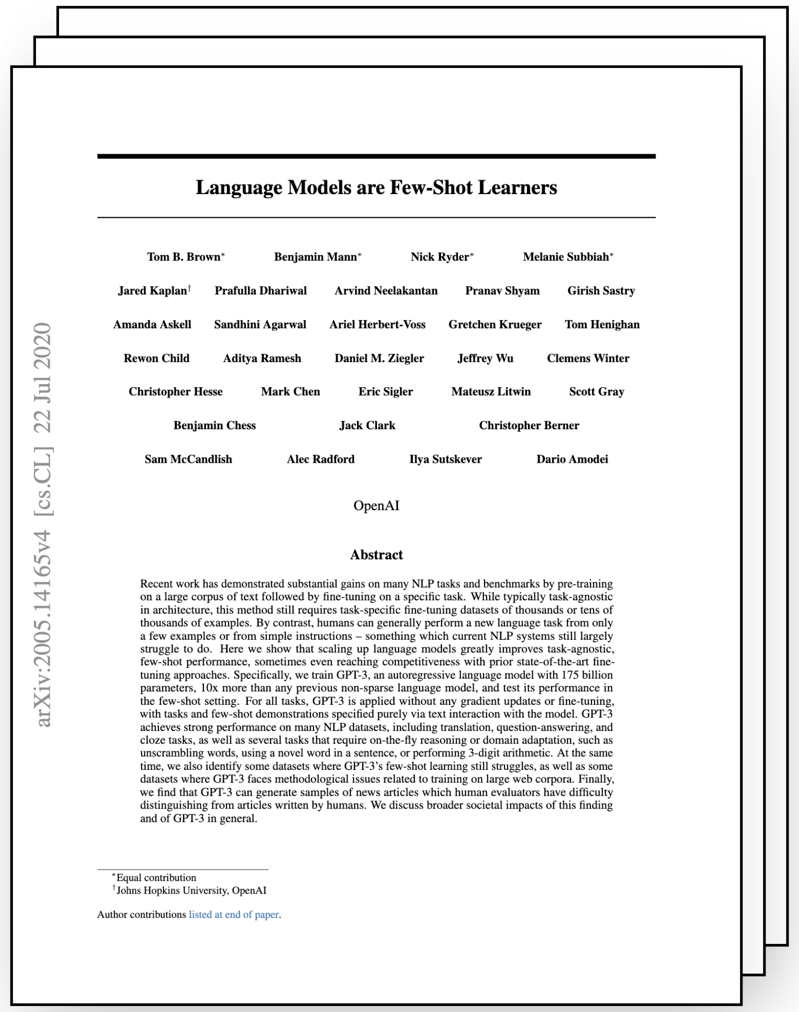


100 web pages

(~**100K** tokens)

Language models: the new “search engines”

Challenge #3: how to fit the context

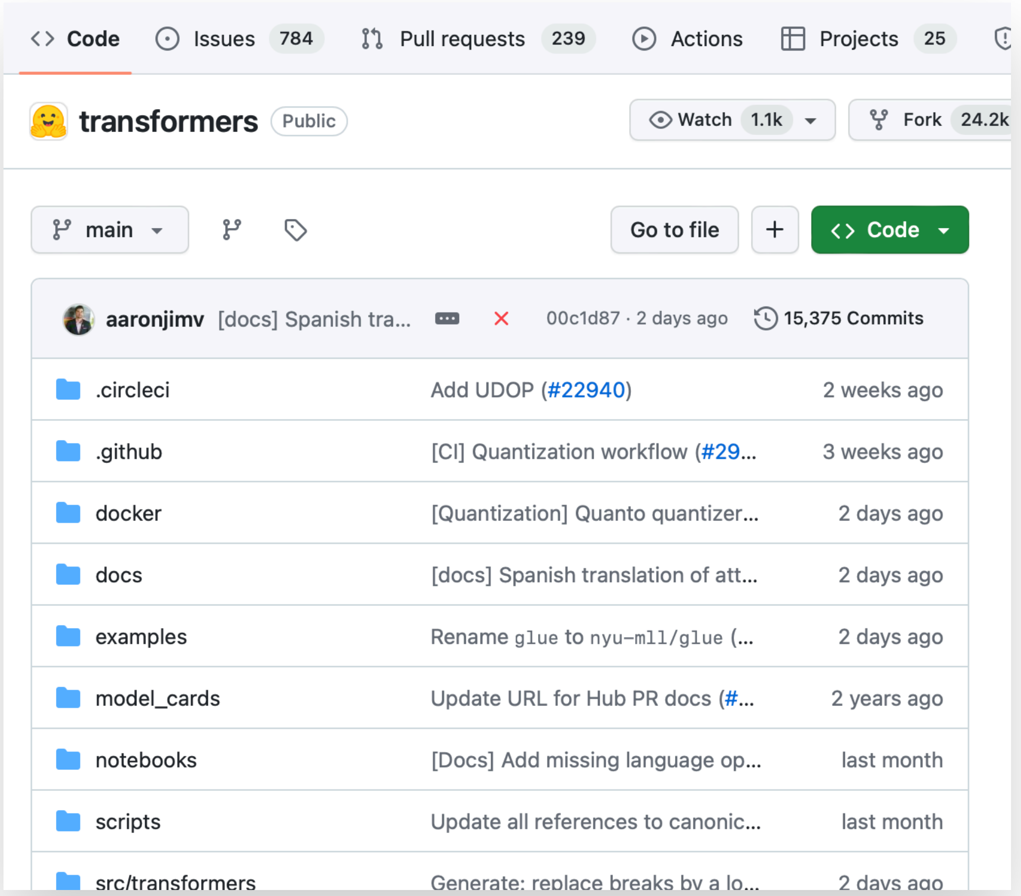


The GPT-3 paper

(~**75K** tokens)

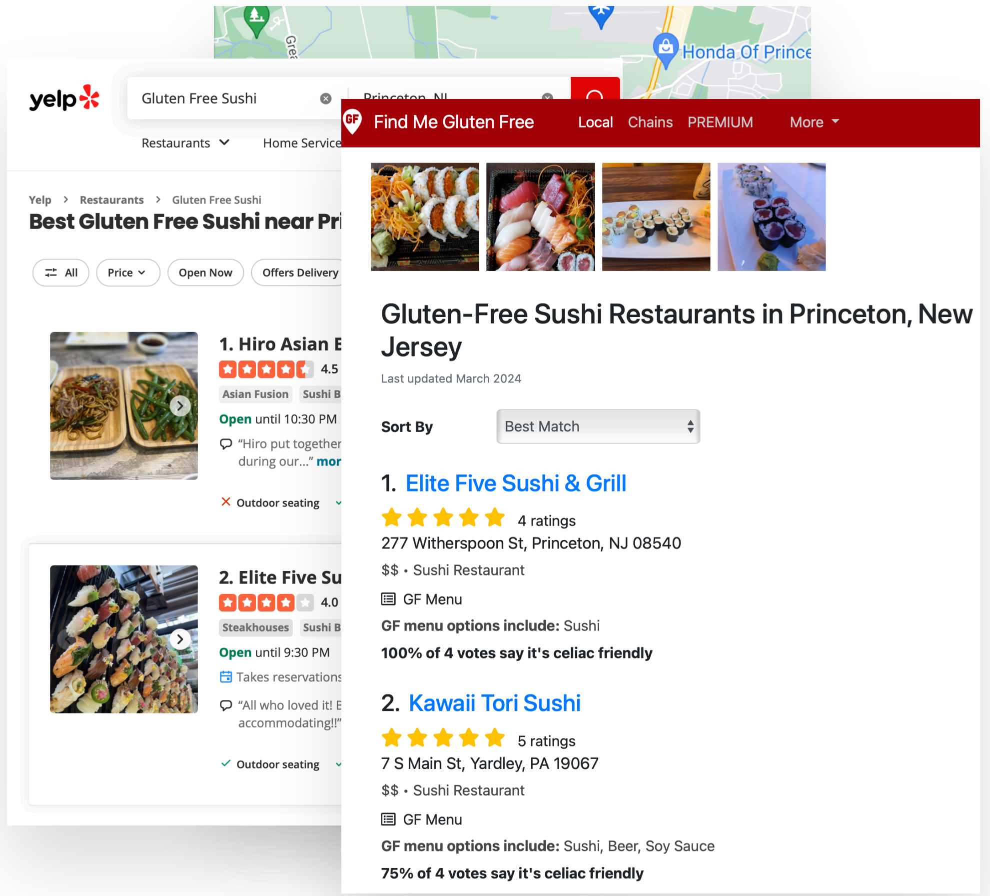
The Dune series

(~**1M** words)



The Transformers package

(~**10M** tokens)



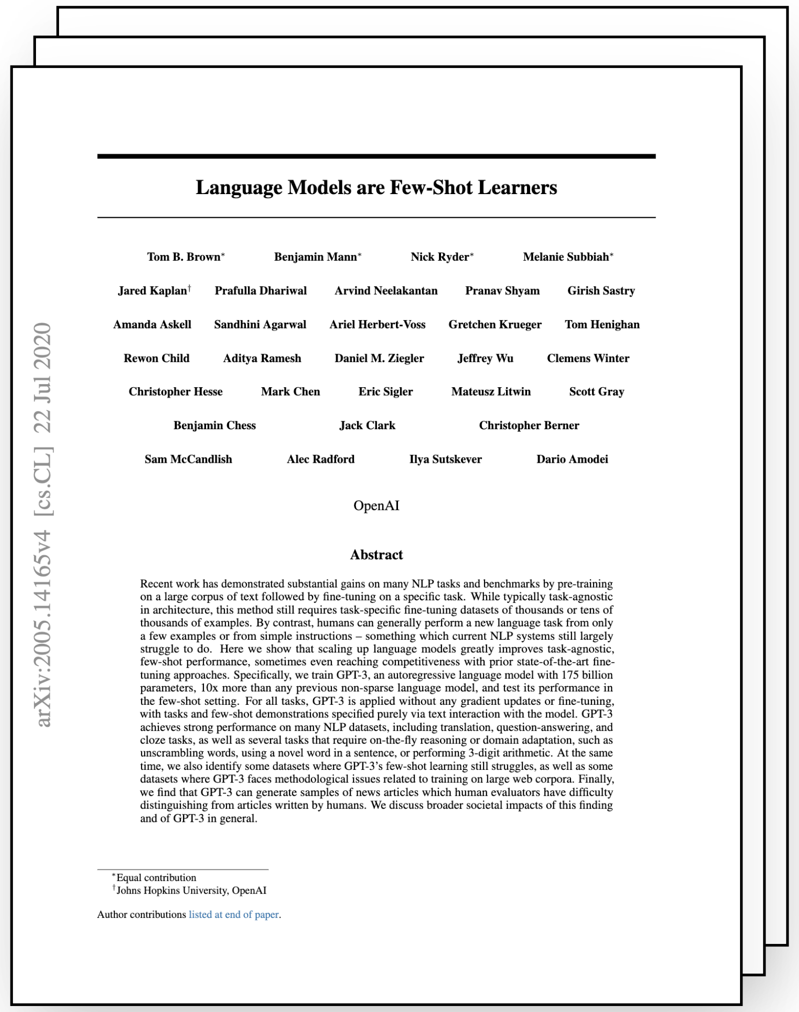
100 web pages

(~**100K** tokens)

Meta LLaMA-2: 4K

Language models: the new "search engines"

Challenge #3: how to fit the context

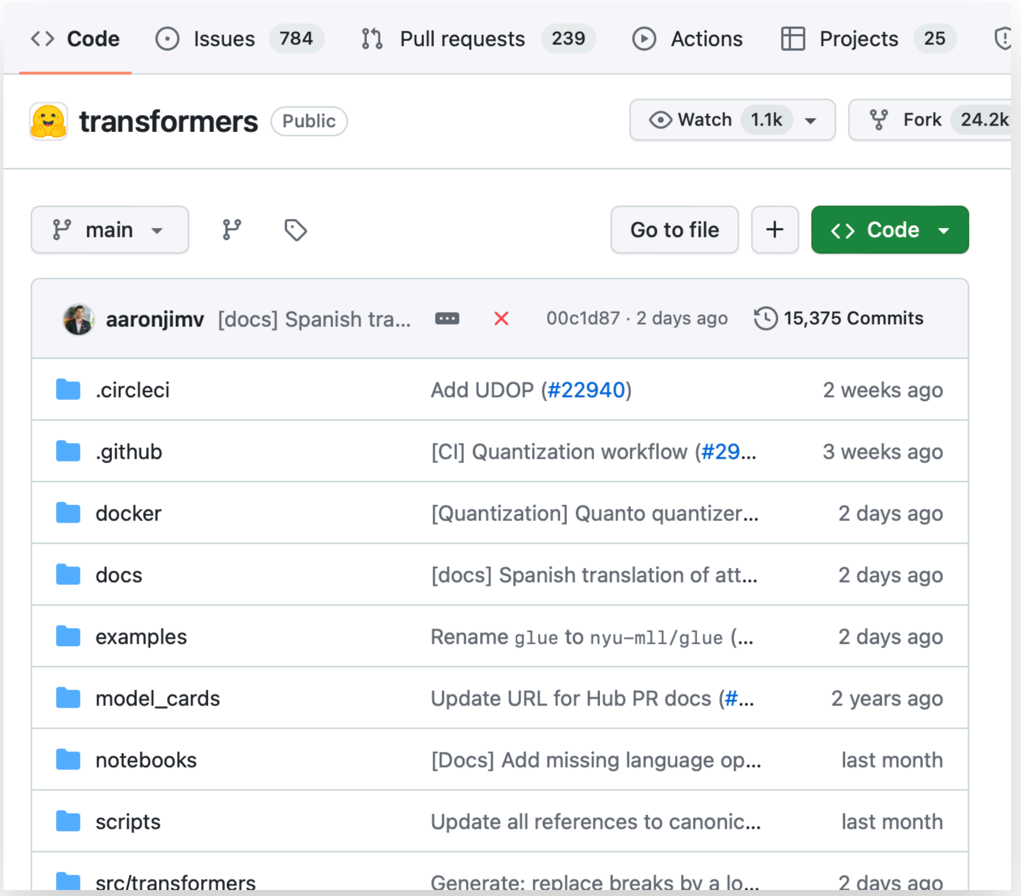


The GPT-3 paper

(~**75K** tokens)

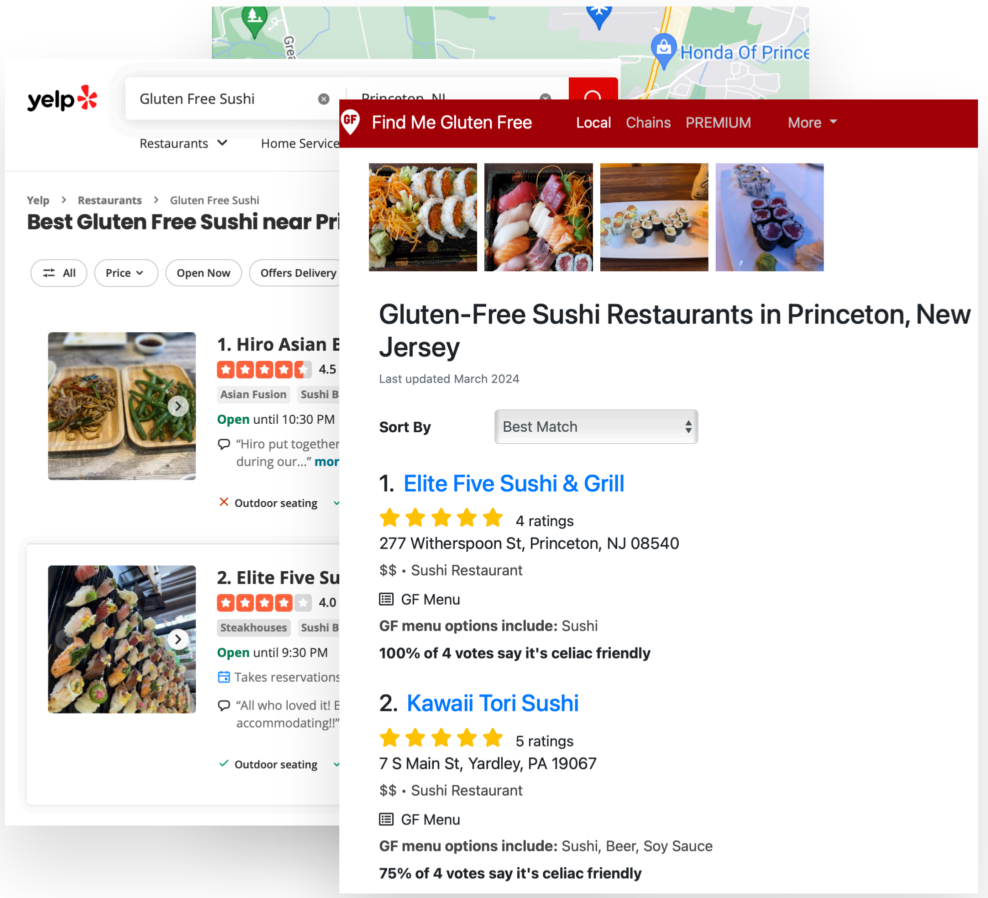
The Dune series

(~**1M** words)



The Transformers package

(~**10M** tokens)



100 web pages

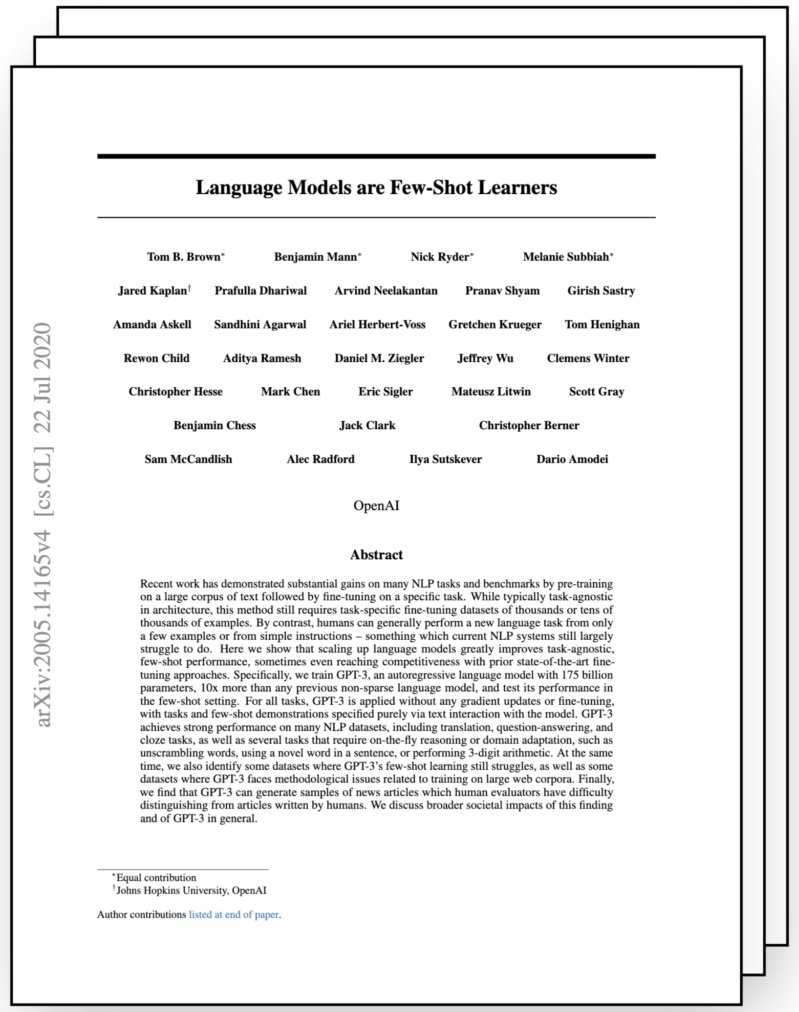
(~**100K** tokens)

Meta LLaMA-2: 4K

MistralAI Mixtral: 32K

Language models: the new "search engines"

Challenge #3: how to fit the context

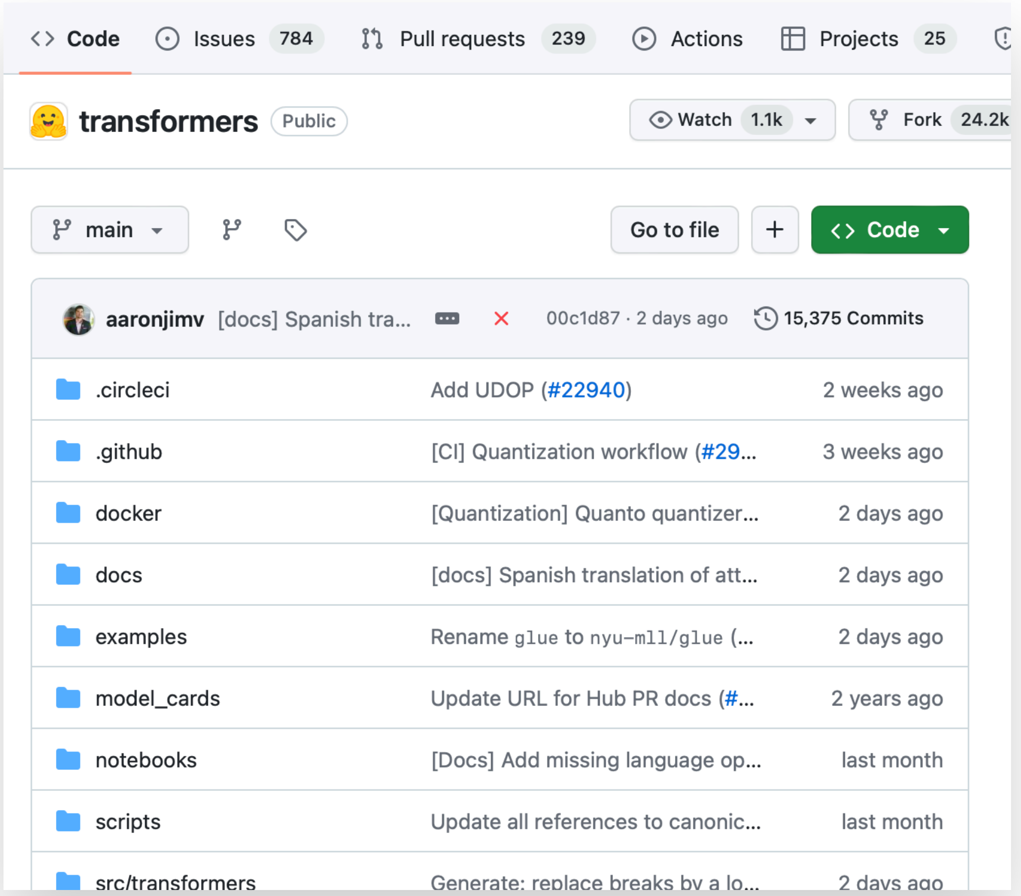


The GPT-3 paper

(~**75K** tokens)

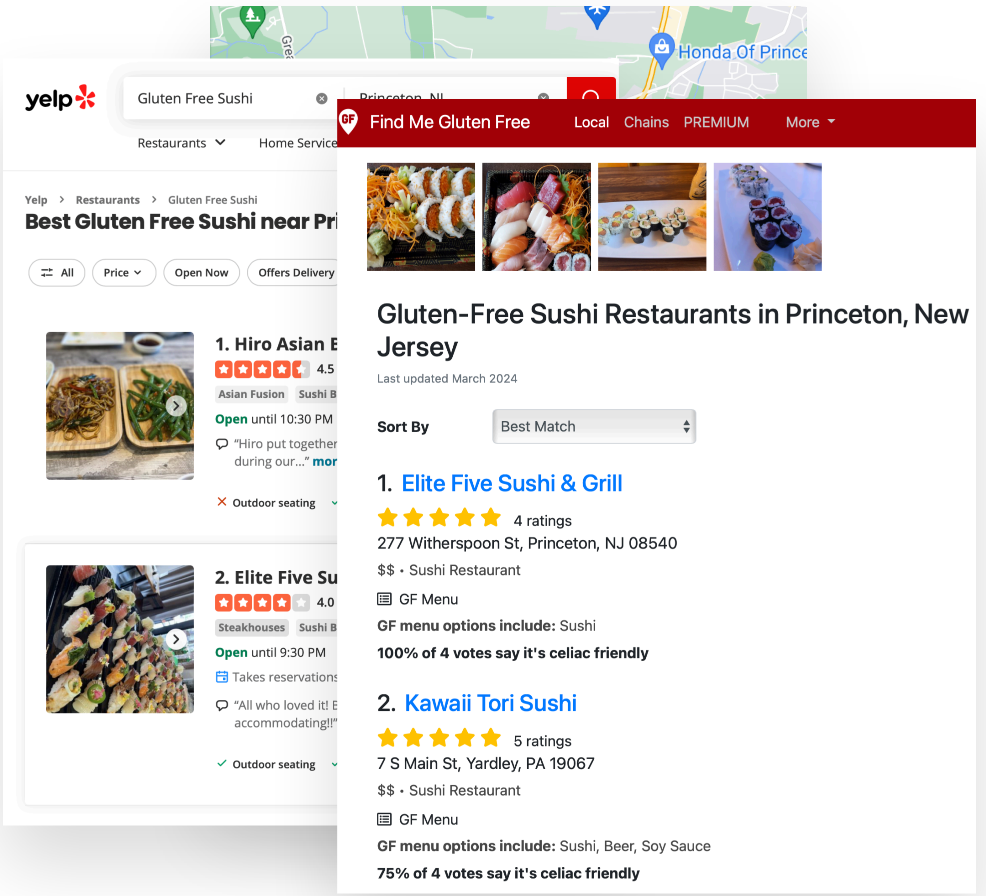
The Dune series

(~**1M** words)



The Transformers package

(~**10M** tokens)



100 web pages

(~**100K** tokens)

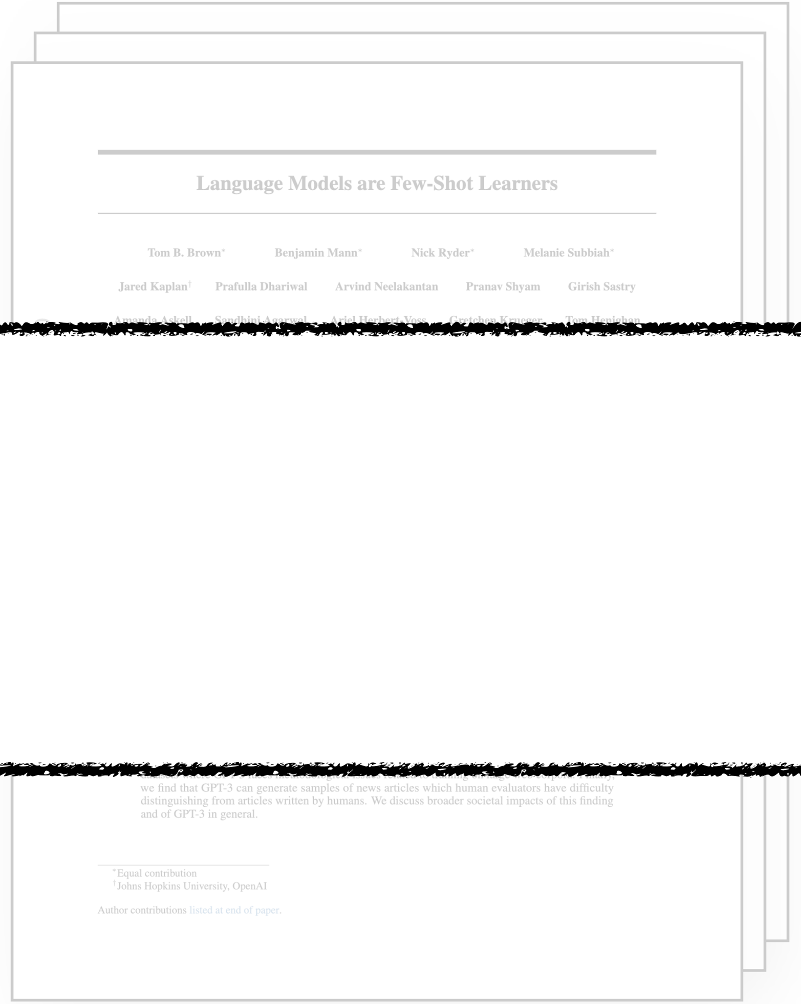
Meta LLaMA-2: 4K

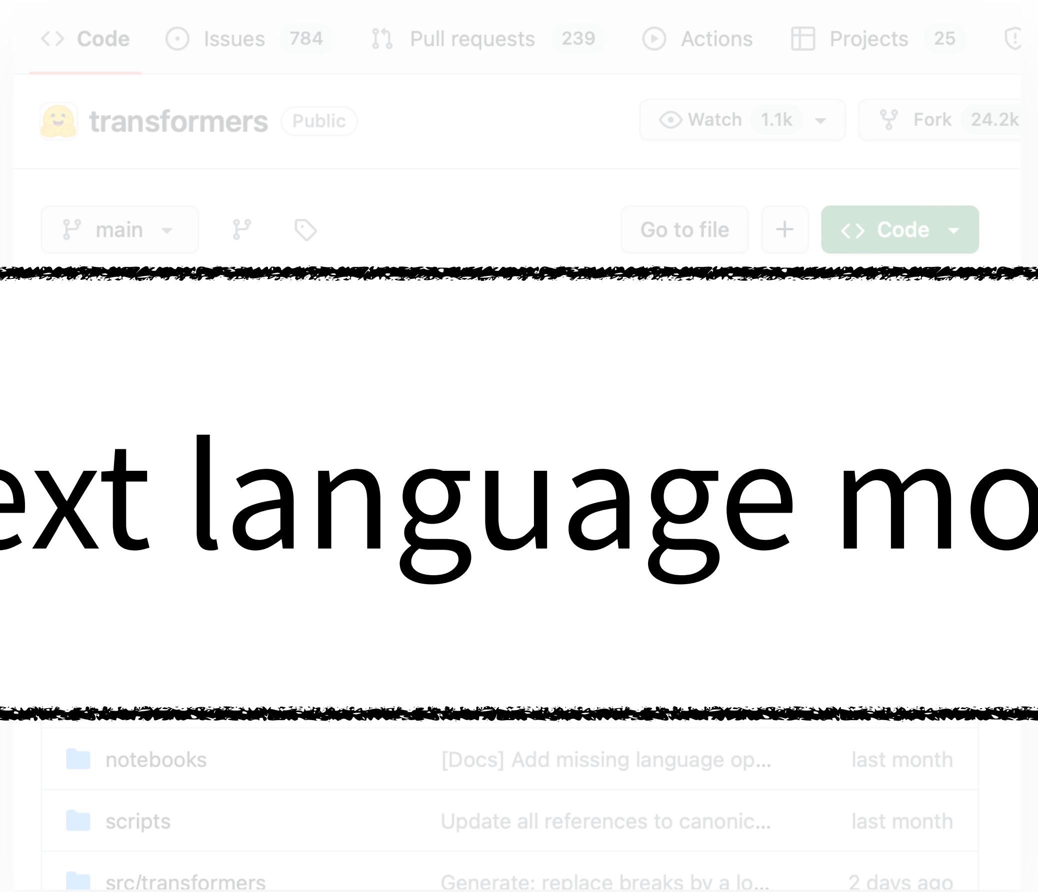
Mistral AI Mixtral: 32K

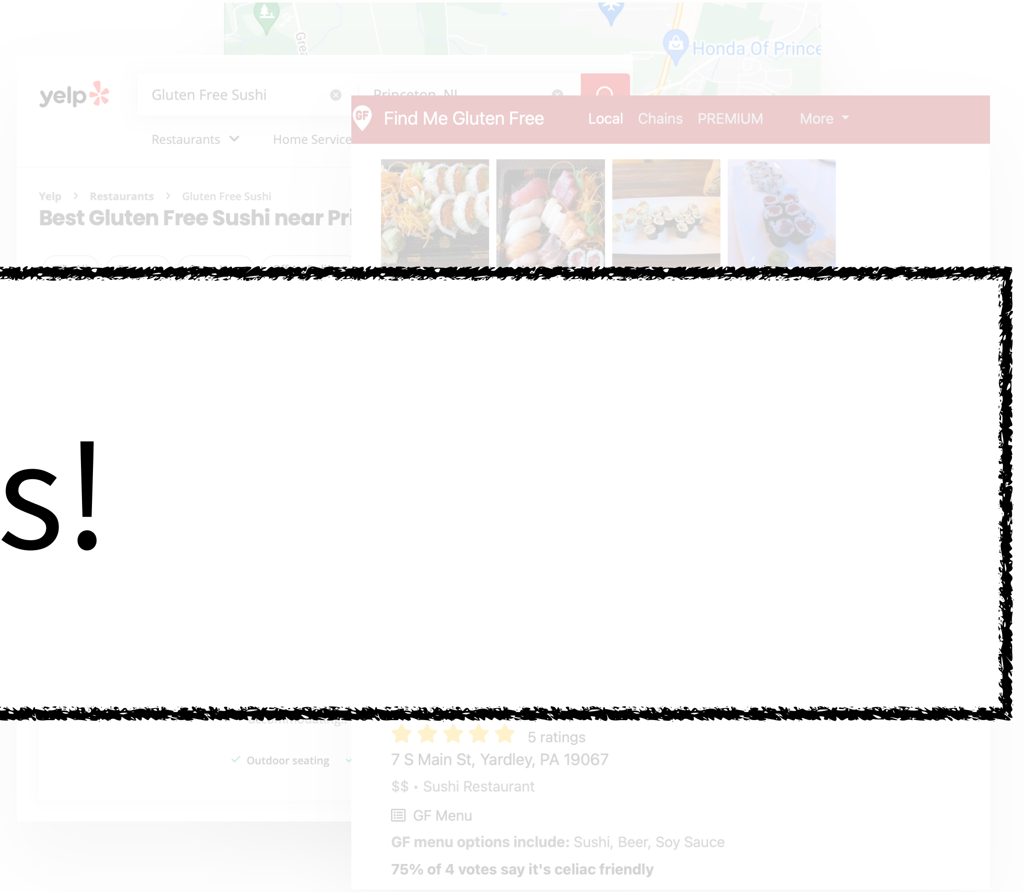
GPT-4: 128K

Language models: the new “search engines”

Challenge #3: how to fit the context







We need long-context language models!

The GPT-3 paper (~**75K** tokens)

The Dune series (~**1M** words)

The Transformers package (~**10M** tokens)

100 web pages (~**100K** tokens)

Meta LLaMA-2: 4K

Mistral AI Mixtral: 32K

GPT-4: 128K

Long-context hurdle #1: Transformers

Vaswani et al., 2017. Attention is all you need.

Long-context hurdle #1: Transformers

Transformers are costly (both computation and memory)

Vaswani et al., 2017. Attention is all you need.

Long-context hurdle #1: Transformers

Transformers are costly (both computation and memory)

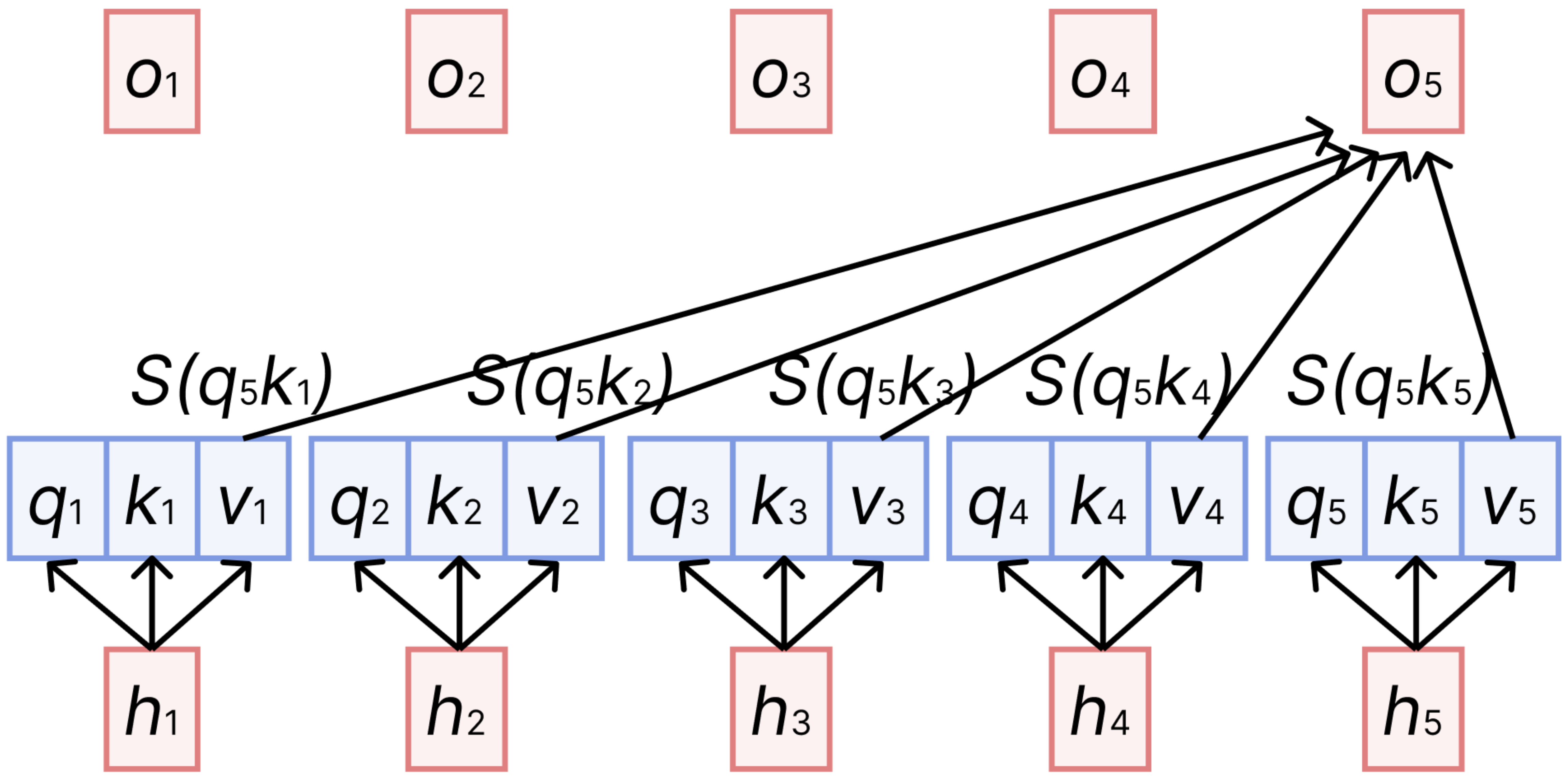
Multi-head attention: the representation of every word is a *weighted sum* of *all* previous words

Vaswani et al., 2017. Attention is all you need.

Long-context hurdle #1: Transformers

Transformers are costly (both computation and memory)

Multi-head attention: the representation of every word is a *weighted sum* of *all* previous words

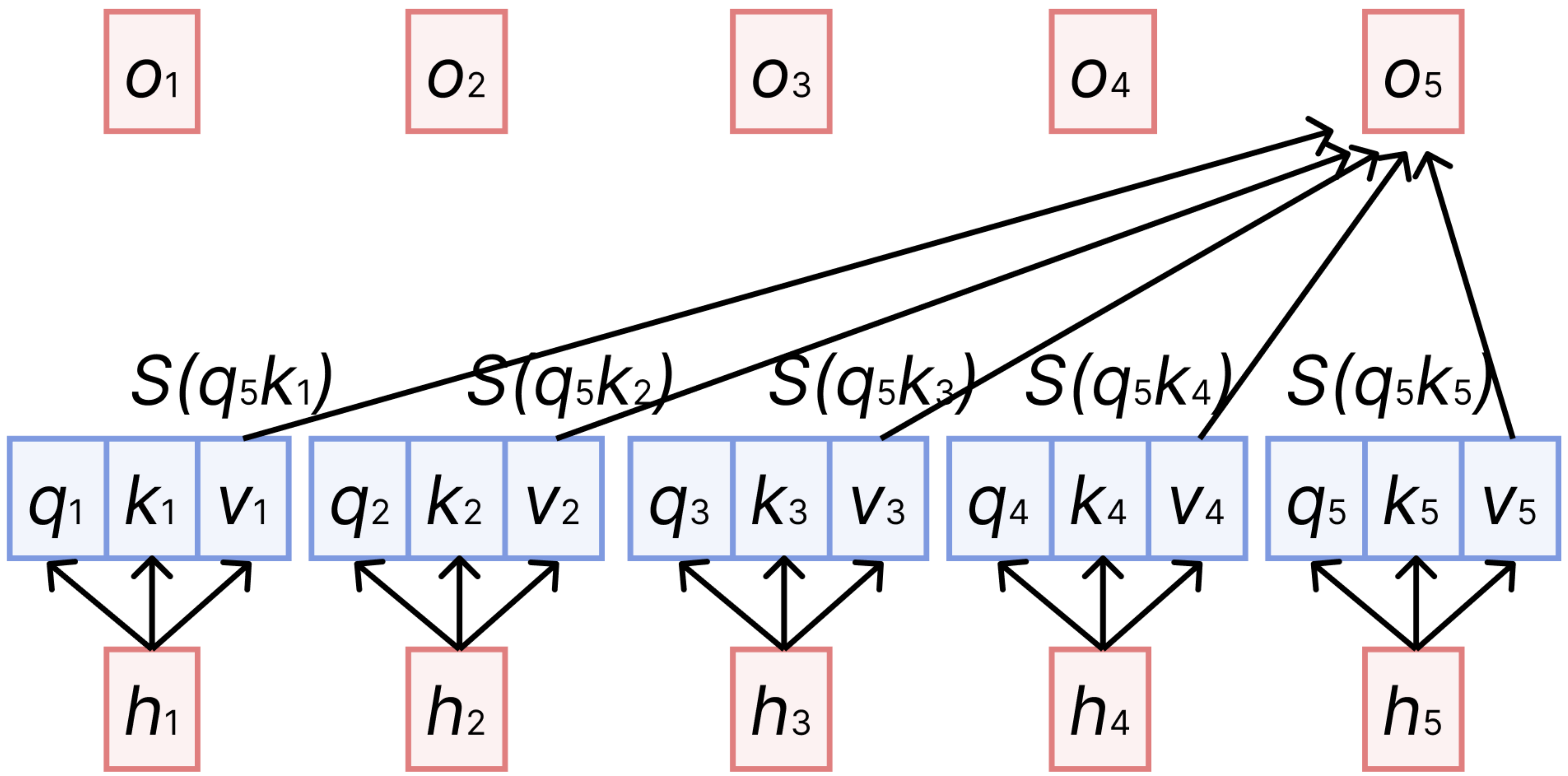


Vaswani et al., 2017. Attention is all you need.

Long-context hurdle #1: Transformers

Transformers are costly (both computation and memory)

Multi-head attention: the representation of every word is a *weighted sum* of *all* previous words



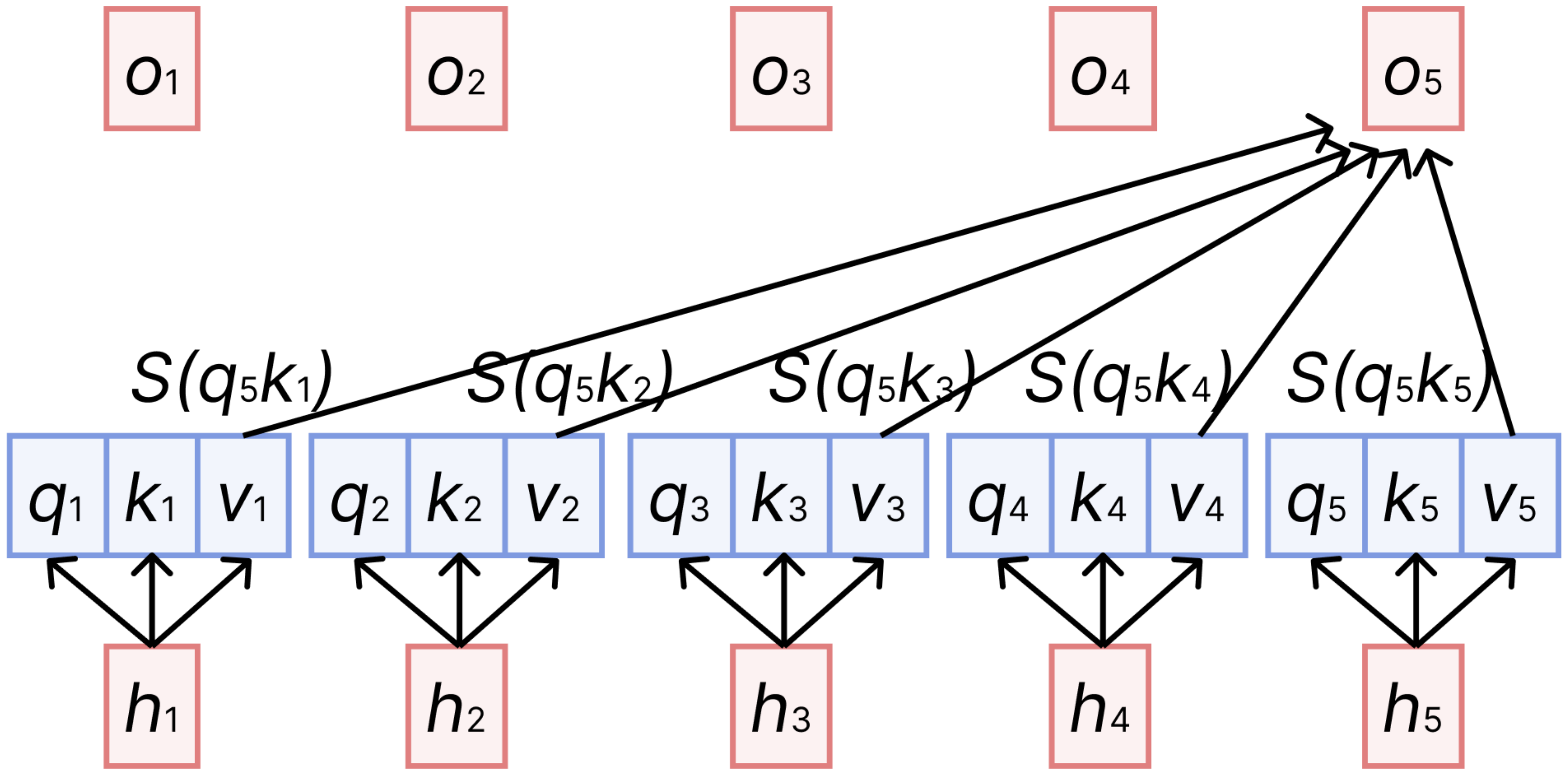
To encode a context of n words
$\mathcal{O}(n^2)$ compute complexity

Vaswani et al., 2017. Attention is all you need.

Long-context hurdle #1: Transformers

Transformers are costly (both computation and memory)

Multi-head attention: the representation of every word is a *weighted sum* of *all* previous words



To encode a context of n words
$\mathcal{O}(n^2)$ compute complexity

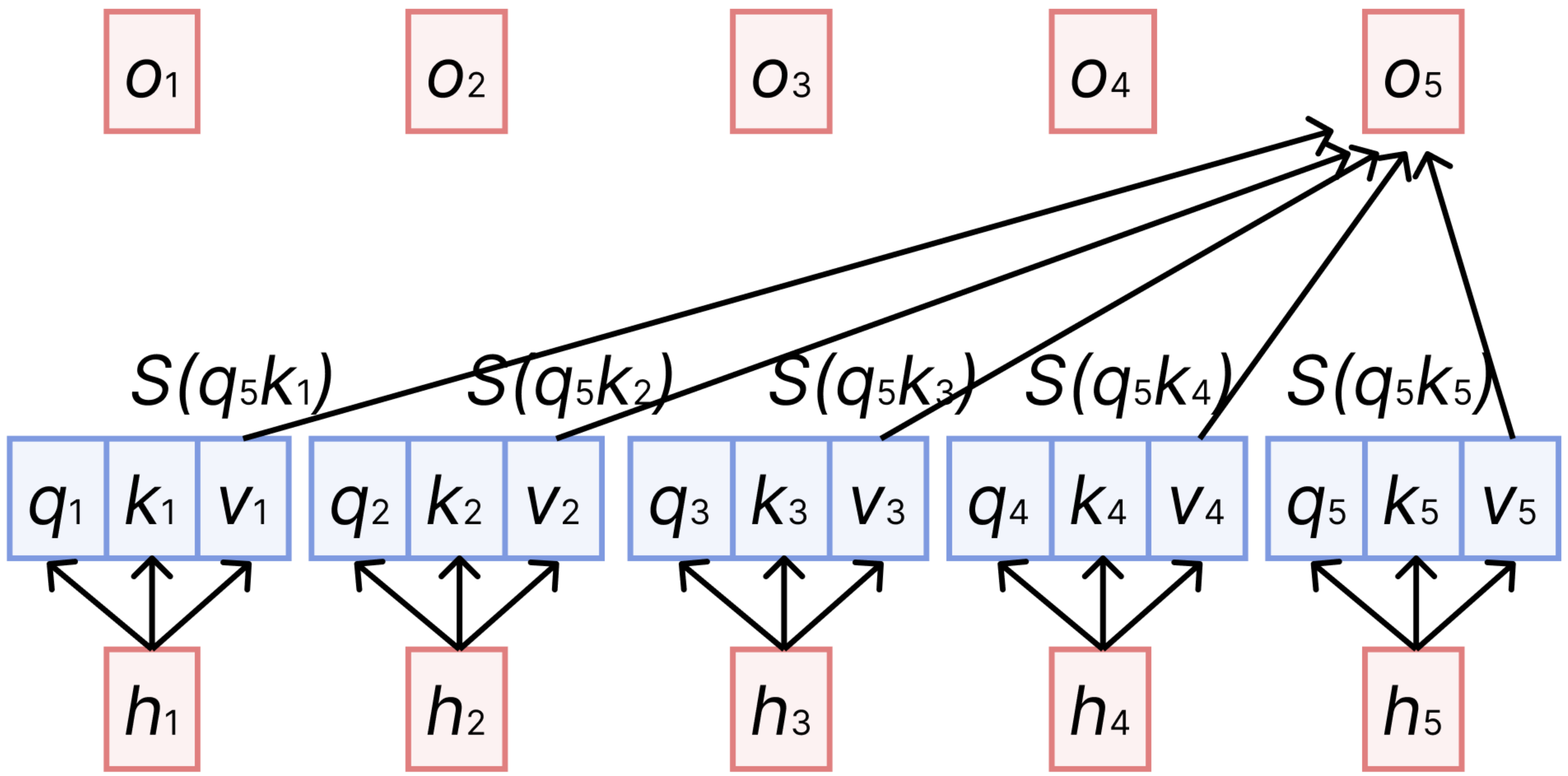
To predict the next word
$\mathcal{O}(n)$ memory complexity

Vaswani et al., 2017. Attention is all you need.

Long-context hurdle #1: Transformers

Transformers are costly (both computation and memory)

Multi-head attention: the representation of every word is a *weighted sum* of *all* previous words



To encode a context of n words
$\mathcal{O}(n^2)$ compute complexity

To predict the next word
$\mathcal{O}(n)$ memory complexity

A **1M** token context would cost **164GB** memory!
(LLaMA-70B, FP16)

Vaswani et al., 2017. Attention is all you need.

Long-context hurdle #2: Positional encodings

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.

Long-context hurdle #2: Positional encodings

Popular positional encodings are not generalizable

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.

Long-context hurdle #2: Positional encodings

Popular positional encodings are not generalizable

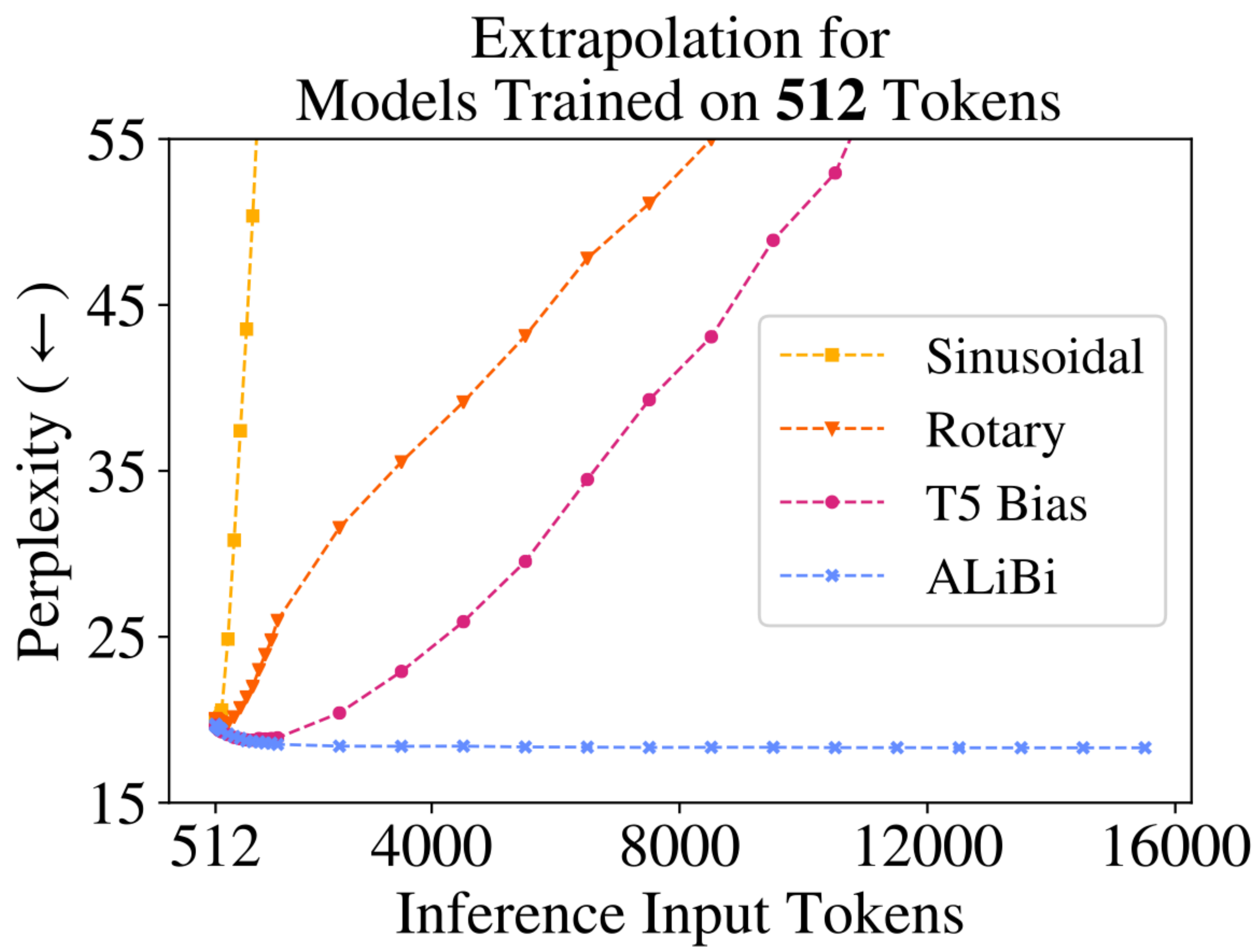
The most popular positional encoding method, RoPE (Su et al., 2021), cannot generalize beyond the training length.

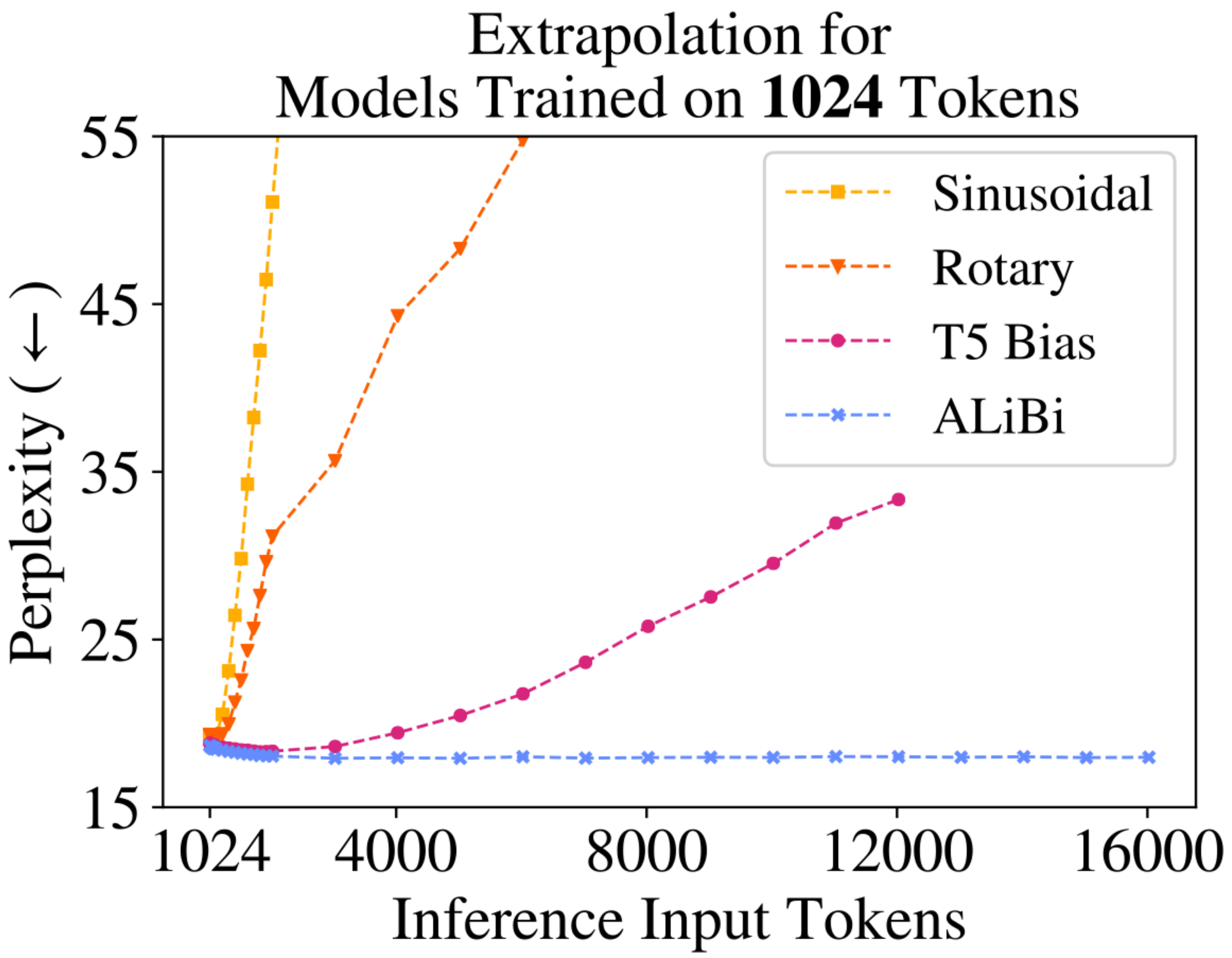
Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.

Long-context hurdle #2: Positional encodings

Popular positional encodings are not generalizable

The most popular positional encoding method, RoPE (Su et al., 2021), cannot generalize beyond the training length.





Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.

Long-context hurdle #3: Data scarcity

Together. 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.

Long-context hurdle #3: Data scarcity

High-quality long-context data are hard to find

Together. 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.

Long-context hurdle #3: Data scarcity

High-quality long-context data are hard to find

Average length of domains from **RedPajama** (a open-source pre-training data collection)

- Wikipedia: 0.5K tokens
- C4 (webpages): 0.5K tokens
- Arxiv: 20K tokens
- Books: 147K tokens

Average length of **instruction-tuning/chat** data: <1K tokens

Together. 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.

Long-context hurdle #3: Data scarcity

High-quality long-context data are hard to find

Average length of domains from **RedPajama** (a open-source pre-training data collection)

- Wikipedia: 0.5K tokens
- C4 (webpages): 0.5K tokens
- Arxiv: 20K tokens
- Books: 147K tokens

Average length of **instruction-tuning/chat** data: <1K tokens

How can we train a model that can continually generalize to longer length?

Together. 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.

Long-Context Language Modeling with Parallel Context Encoding

Long-Context Language Modeling with Parallel Context Encoding

- We propose 🍄**CEPE**: **Context Expansion with Parallel Encoding**

Long-Context Language Modeling with Parallel Context Encoding

- We propose 🍄**CEPE**: **Context Expansion with Parallel Encoding**
    - A lightweight framework that can extend any decoder-only LM’s context length

Long-Context Language Modeling with Parallel Context Encoding

- We propose 🍄**CEPE**: **Context Expansion with Parallel Encoding**
  - A lightweight framework that can extend any decoder-only LM’s context length
- Why CEPE

Long-Context Language Modeling with Parallel Context Encoding

- We propose 🍄**CEPE**: **Context Expansion with Parallel Encoding**
  - A lightweight framework that can extend any decoder-only LM's context length
- Why CEPE
  - **Length generalization** (train on 8K, generalize to 128K)

Long-Context Language Modeling with Parallel Context Encoding

- We propose 🍄**CEPE**: **Context Expansion with Parallel Encoding**
    - A lightweight framework that can extend any decoder-only LM's context length
- Why CEPE
    - **Length generalization** (train on 8K, generalize to 128K)
    - **Efficiency** (10x throughput, 1/256 memory for every extra token* compared to LLaMA-7B)

Long-Context Language Modeling with Parallel Context Encoding

- We propose 🍄**CEPE**: **Context Expansion with Parallel Encoding**
  - A lightweight framework that can extend any decoder-only LM's context length
- Why CEPE
  - **Length generalization** (train on 8K, generalize to 128K)
  - **Efficiency** (10x throughput, 1/256 memory for every extra token* compared to LLaMA-7B)
  - **Low training cost** (extending a LLaMA-7B can be done by 1 A100)

Long-Context Language Modeling with Parallel Context Encoding

- We propose 🍄**CEPE**: **Context Expansion with Parallel Encoding**
  - A lightweight framework that can extend any decoder-only LM's context length
- Why CEPE
  - **Length generalization** (train on 8K, generalize to 128K)
  - **Efficiency** (10x throughput, 1/256 memory for every extra token* compared to LLaMA-7B)
  - **Low training cost** (extending a LLaMA-7B can be done by 1 A100)
  - Also work for **chat** models

Long-Context Language Modeling with Parallel Context Encoding

- We propose 🍄**CEPE**: **Context Expansion with Parallel Encoding**
  - A lightweight framework that can extend any decoder-only LM's context length
- Why CEPE
  - **Length generalization** (train on 8K, generalize to 128K)
  - **Efficiency** (10x throughput, 1/256 memory for every extra token* compared to LLaMA-7B)
  - **Low training cost** (extending a LLaMA-7B can be done by 1 A100)
  - Also work for **chat** models
- CEPE achieves great performance on both **long-context** and **retrieval-augmented** applications

How do decoder-only LMs handle long context?

How do decoder-only LMs handle long context?

The Dune series

Q: Who betrayed the Atreides?

How do decoder-only LMs handle long context?

Decoder-only language model

The Dune series

Q: Who betrayed the Atreides?

How do decoder-only LMs handle long context?

Decoder-only language model

The Dune series

Q: Who betrayed the Atreides?

$\mathcal{O}(n)$ memory cost

$\mathcal{O}(n^2)$ computational cost

How do decoder-only LMs handle long context?

Decoder-only language model

The Dune series

Q: Who betrayed the Atreides?

$\mathcal{O}(n)$ memory cost

$\mathcal{O}(n^2)$ computational cost

n ~ 10K to 1M

How does CEPE handle long context?

The Dune series

Q: Who betrayed the Atreides?

How does CEPE handle long context?

The Dune series

Additional context

Q: Who betrayed the Atreides?

How does CEPE handle long context?

The Dune series

Additional context

Q: Who betrayed the Atreides?

Main input

How does CEPE handle long context?

The Dune series

Q: Who betrayed the Atreides?

Additional context

Main input

- Can be processed by “chunks”
- Order does not matter much

How does CEPE handle long context?

The Dune series

Q: Who betrayed the Atreides?

Additional context

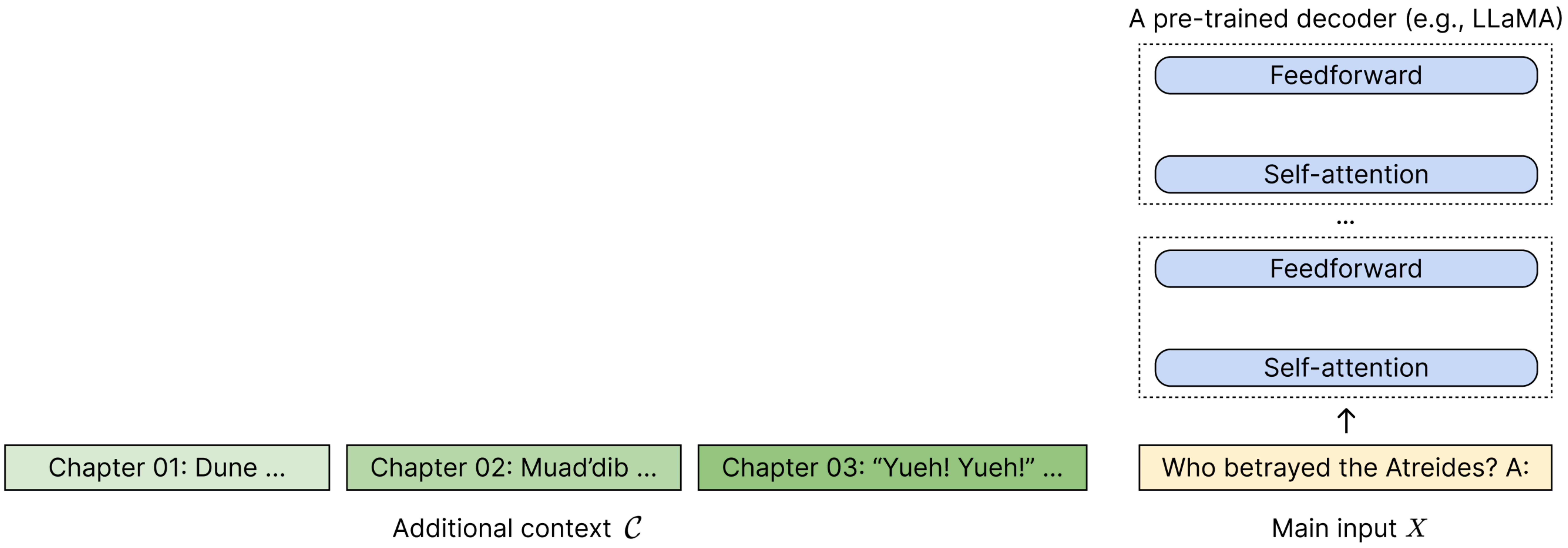
- Can be processed by “chunks”
- Order does not matter much

Main input

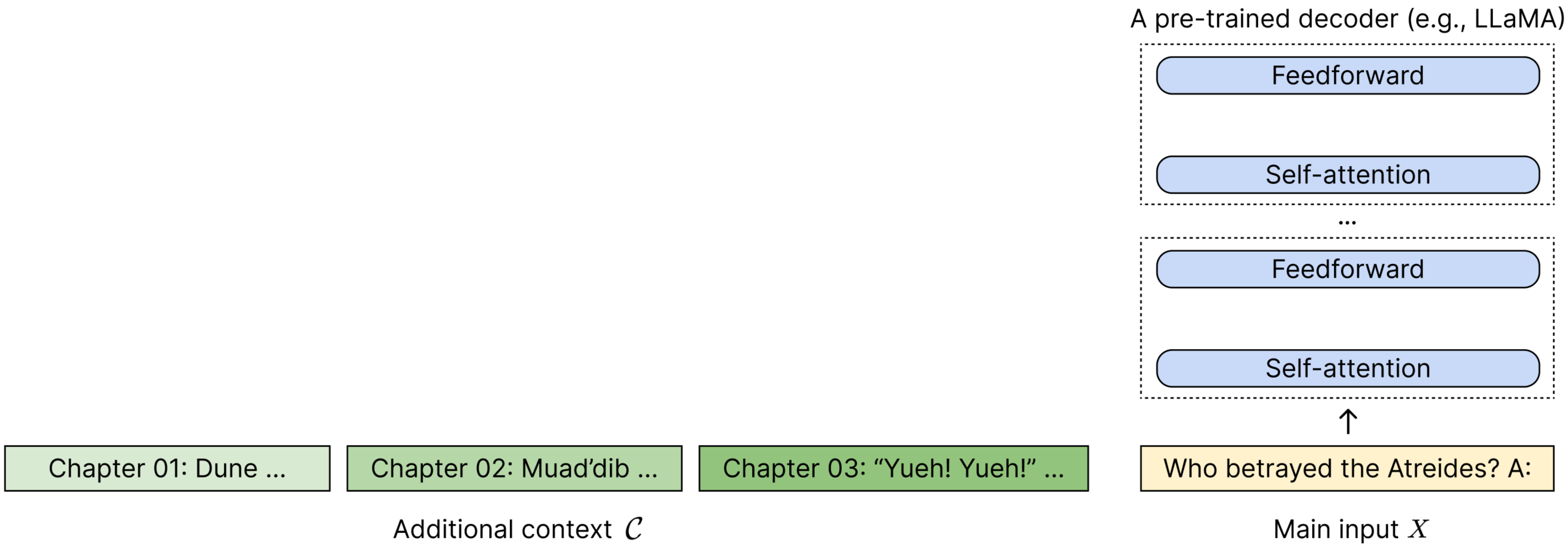
- Should be processed with additional context
- Directly related to the generation

Context Expansion with Parallel Encoding (CEPE)

Context Expansion with Parallel Encoding (CEPE)

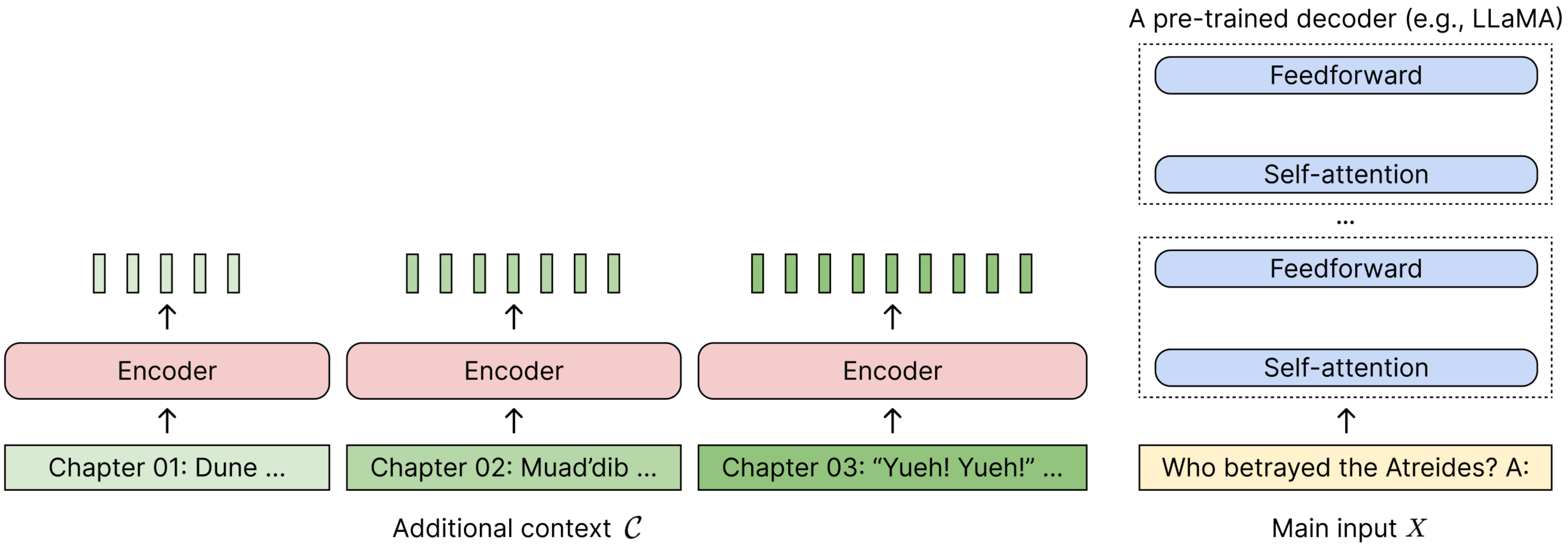


Context Expansion with Parallel Encoding (CEPE)



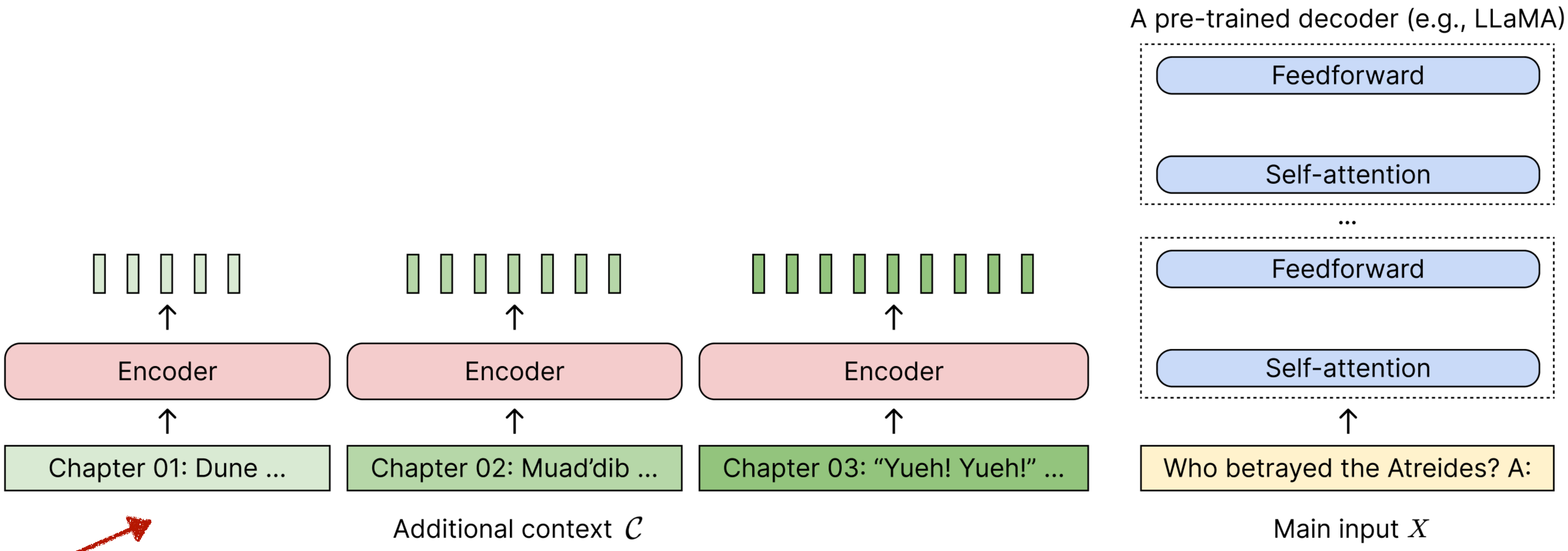
We use an existing decoder-only model (e.g., LLaMA-7B) to process the main input.

Context Expansion with Parallel Encoding (CEPE)



We use a small bidirectional encoder (435M) to encode the additional context *by chunks*.

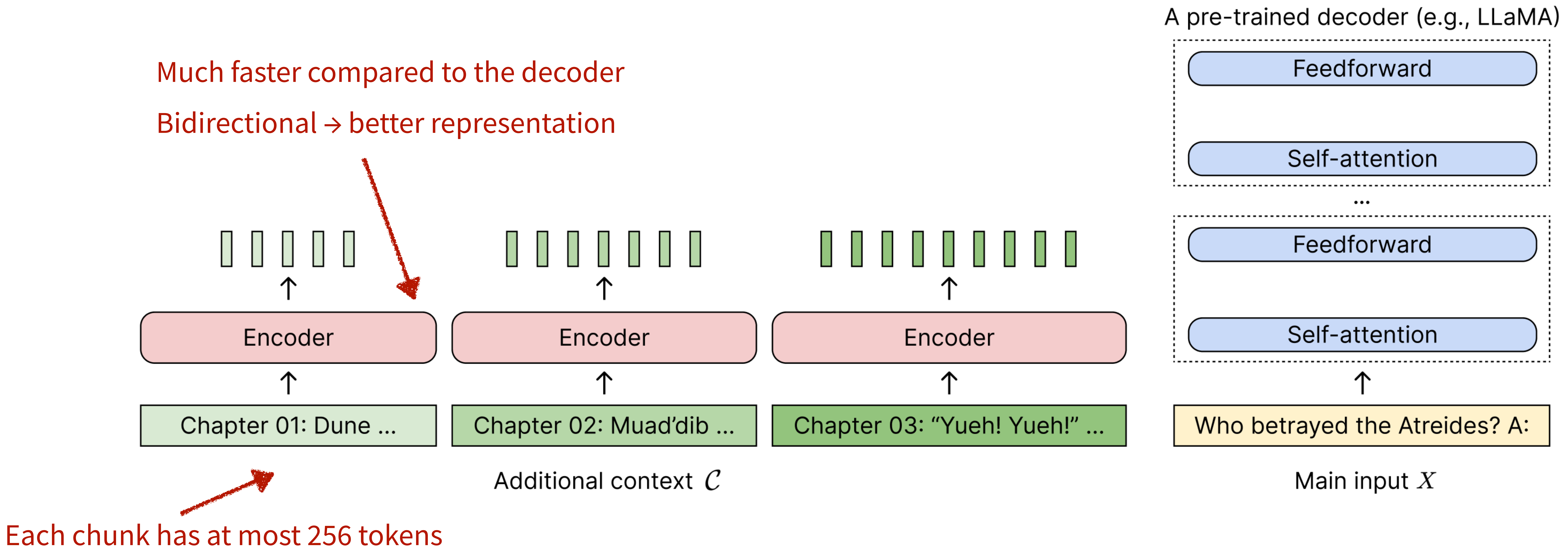
Context Expansion with Parallel Encoding (CEPE)



Each chunk has at most 256 tokens

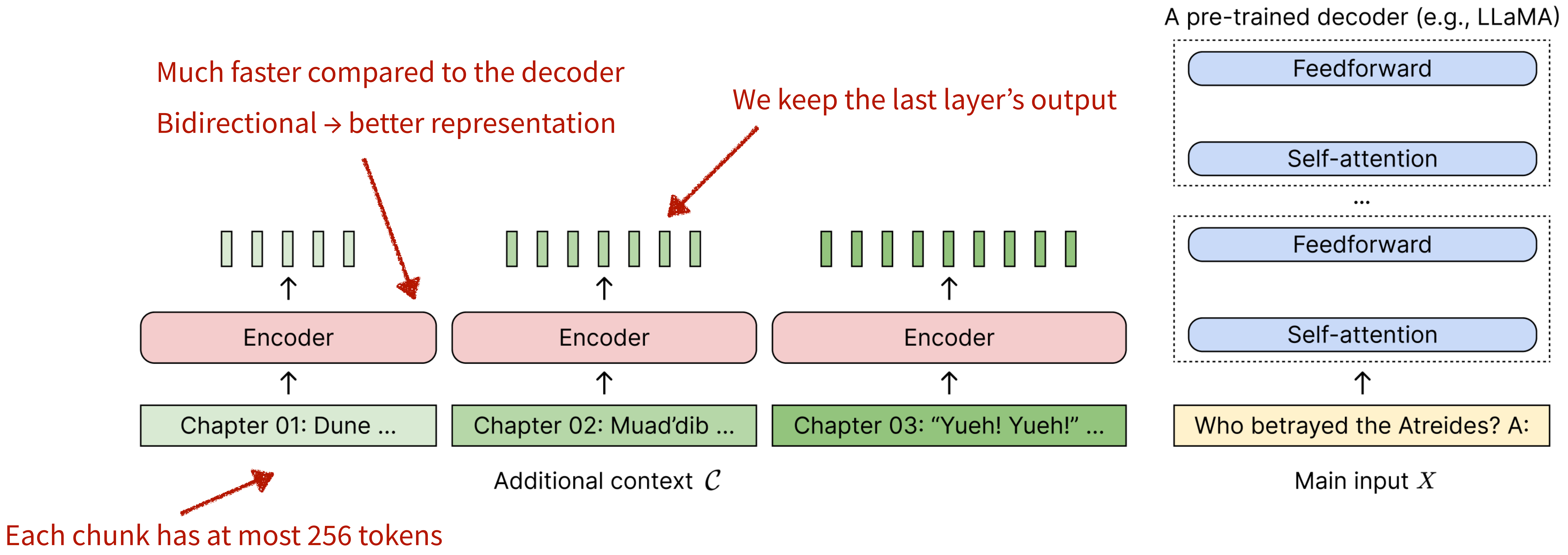
We use a small bidirectional encoder (435M) to encode the additional context *by chunks*.

Context Expansion with Parallel Encoding (CEPE)



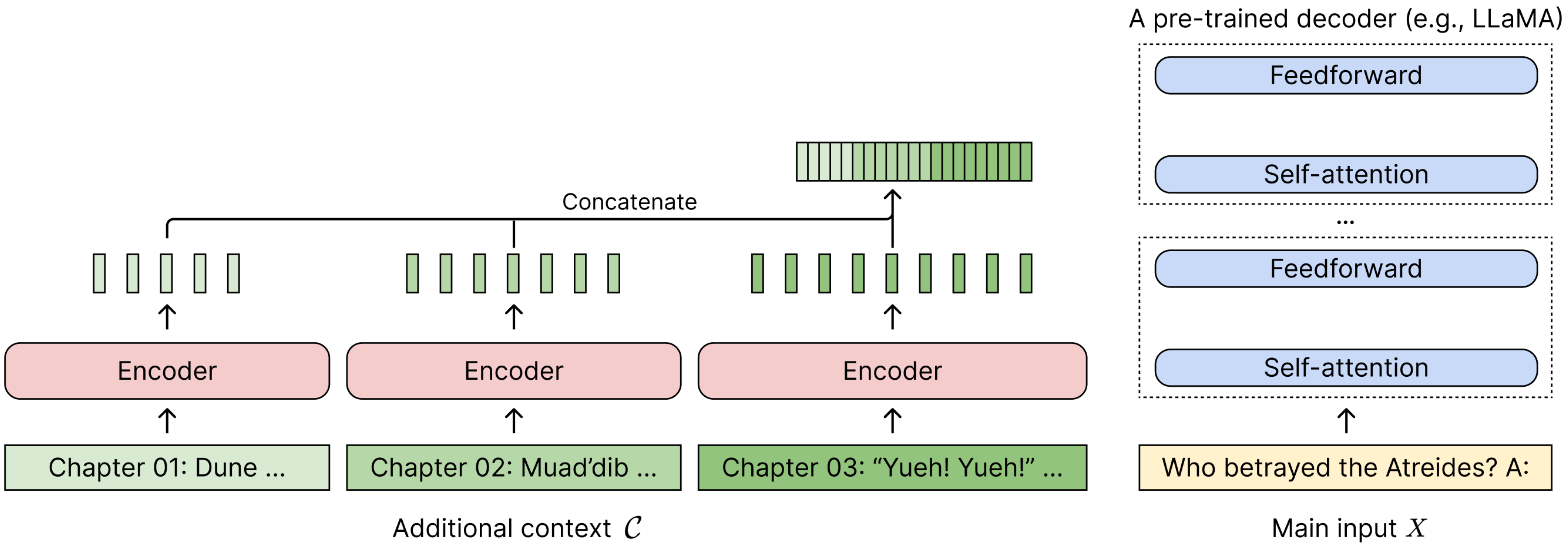
We use a small bidirectional encoder (435M) to encode the additional context *by chunks*.

Context Expansion with Parallel Encoding (CEPE)



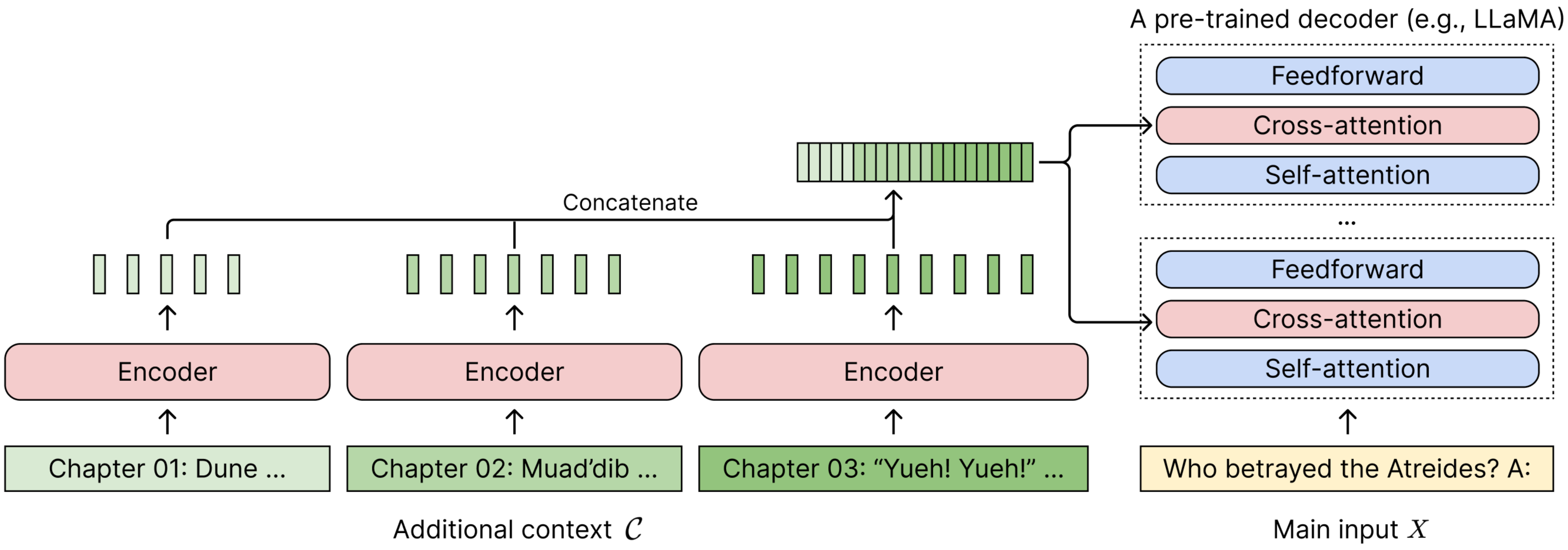
We use a small bidirectional encoder (435M) to encode the additional context *by chunks*.

Context Expansion with Parallel Encoding (CEPE)



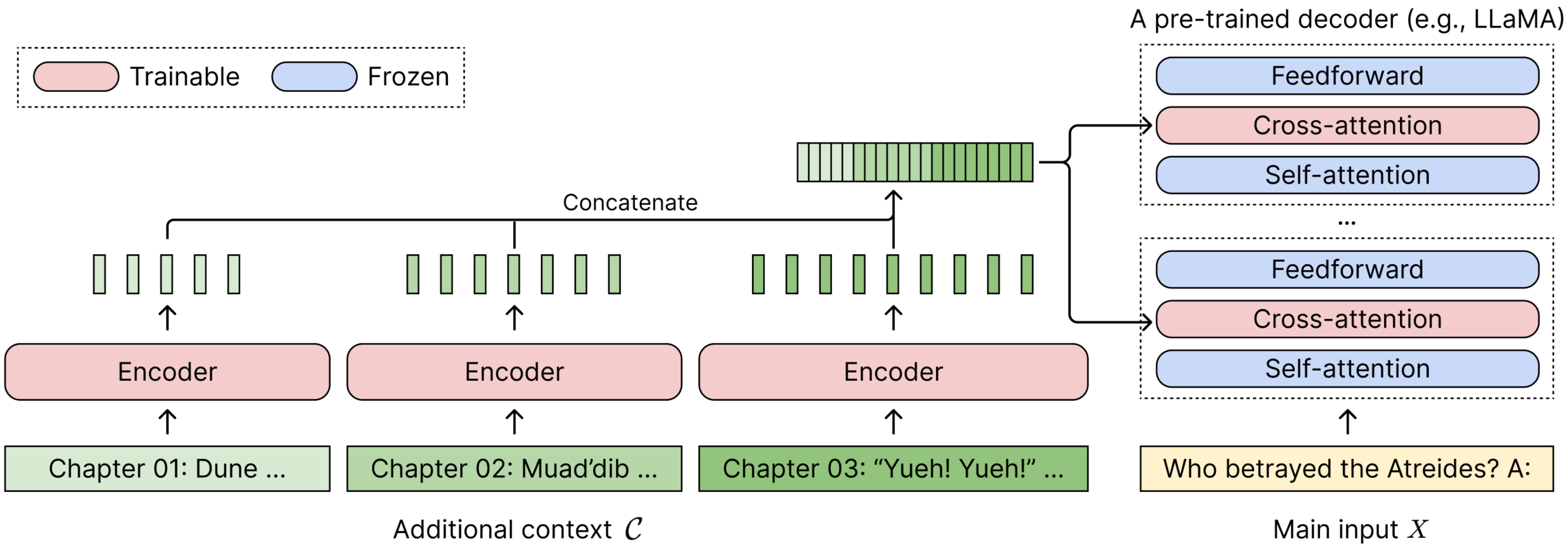
All the encoder outputs are concatenated as the representation for the additional context.

Context Expansion with Parallel Encoding (CEPE)



We insert *cross-attention* into every layer of the decoder, which attends to the additional context.

Context Expansion with Parallel Encoding (CEPE)



We freeze the decoder and only tune the small encoder and the cross-attention modules.

Benefit #1: Length generalization

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.
Chen et al., 2023. Extending Context Window of Large Language Models via Positional Interpolation.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Benefit #1: Length generalization

For the decoder-only model (assume trained on **4K** length):

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.
Chen et al., 2023. Extending Context Window of Large Language Models via Positional Interpolation.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Benefit #1: Length generalization

For the decoder-only model (assume trained on **4K** length):

Dune. Chapter 01: ... Chapter 02: ... Chapter 03: ...	Who betrayed the Atreides? A:

Position: 0 1 2 ... 4094 4095 4096 4097 ... 800,012 800,013 800,014 800,015 ...

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.
Chen et al., 2023. Extending Context Window of Large Language Models via Positional Interpolation.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Benefit #1: Length generalization

For the decoder-only model (assume trained on **4K** length):

Dune. Chapter 01: ... Chapter 02: ... Chapter 03: ... | Who betrayed the Atreides? A:

Position: 0 1 2 ... 4094 4095 4096 4097 ... 800,012 800,013 800,014 800,015 ...

The model **does not generalize well beyond its training length**

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.
Chen et al., 2023. Extending Context Window of Large Language Models via Positional Interpolation.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Benefit #1: Length generalization

For the decoder-only model (assume trained on **4K** length):

Dune. Chapter 01: ... Chapter 02: ... Chapter 03: ...	Who betrayed the Atreides? A:

Position: 0 1 2 ... 4094 4095 4096 4097 ... 800,012 800,013 800,014 800,015 ...

The model **does not generalize well beyond its training length**

- RoPE embeddings can generalize infinitely in theory (Su et al., 2021)

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.
Chen et al., 2023. Extending Context Window of Large Language Models via Positional Interpolation.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Benefit #1: Length generalization

For the decoder-only model (assume trained on **4K** length):

Dune. Chapter 01: ... Chapter 02: ... Chapter 03: ...	Who betrayed the Atreides? A:
Position: 0 1 2 ... 4094 4095 4096 4097 ... 800,012 800,013	800,014 800,015 ...

The model **does not generalize well beyond its training length**

- RoPE embeddings can generalize infinitely in theory (Su et al., 2021)
- … but the model does not perform well on beyond-training lengths (Press et al., 2021)

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.
Chen et al., 2023. Extending Context Window of Large Language Models via Positional Interpolation.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Benefit #1: Length generalization

For the decoder-only model (assume trained on **4K** length):

Dune. Chapter 01: ... Chapter 02: ... Chapter 03: ... | Who betrayed the Atreides? A:

Position: 0 1 2 ... 4094 4095 4096 4097 ... 800,012 800,013 800,014 800,015 ...

The model **does not generalize well beyond its training length**

- RoPE embeddings can generalize infinitely in theory (Su et al., 2021)
- … but the model does not perform well on beyond-training lengths (Press et al., 2021)
- There exists positional interpolation methods (Chen et al., 2023)

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.
Chen et al., 2023. Extending Context Window of Large Language Models via Positional Interpolation.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Benefit #1: Length generalization

For the decoder-only model (assume trained on **4K** length):

Dune. Chapter 01: ... Chapter 02: ... Chapter 03: ...	Who betrayed the Atreides? A:

Position: 0 1 2 ... 4094 4095 4096 4097 ... 800,012 800,013 800,014 800,015 ...

The model **does not generalize well beyond its training length**

- RoPE embeddings can generalize infinitely in theory (Su et al., 2021)
- … but the model does not perform well on beyond-training lengths (Press et al., 2021)
- There exists positional interpolation methods (Chen et al., 2023)
- … but good performance on long context still requires extensive training on the target length (Fu et al., 2024)

Su et al., 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding.
Press et al., 2021. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation.
Chen et al., 2023. Extending Context Window of Large Language Models via Positional Interpolation.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

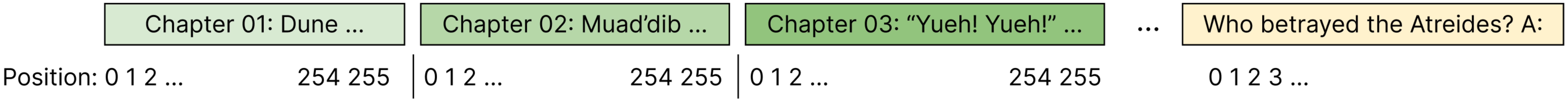
Benefit #1: Length generalization

Benefit #1: Length generalization

For **CEPE**

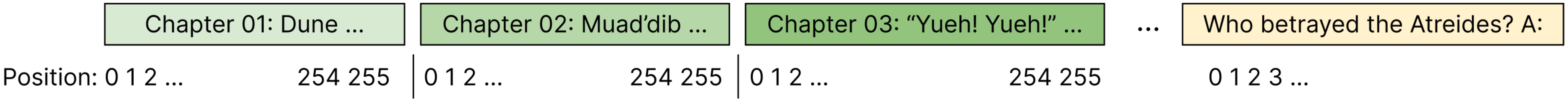
Benefit #1: Length generalization

For **CEPE**



Benefit #1: Length generalization

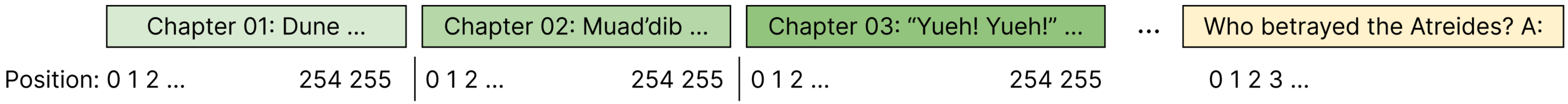
For **CEPE**



Each encoder chunk/the decoder window has its own positional encodings.

Benefit #1: Length generalization

For **CEPE**

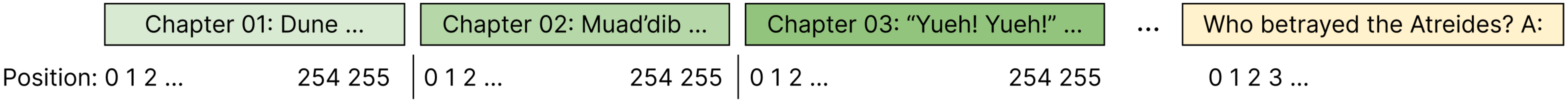


Each encoder chunk/the decoder window has its own positional encodings.

- Generalize to longer length → generalize to *more chunks*

Benefit #1: Length generalization

For **CEPE**



Each encoder chunk/the decoder window has its own positional encodings.

- Generalize to longer length → generalize to *more chunks*
- Trained on 16 chunks, CEPE can generalize to (at least) 128 chunks

Benefit #2: Efficiency

Benefit #2: Efficiency

Throughput

Benefit #2: Efficiency

Throughput

- Full attention: $\mathcal{O}(m^2)$,

Benefit #2: Efficiency

Throughput

#tokens

- Full attention: $\mathcal{O}(m^2)$,

Benefit #2: Efficiency

Throughput

#tokens

- Full attention: $\mathcal{O}(m^2)$,
- Attention by chunks: $\mathcal{O}(kc^2), m = kc$

Benefit #2: Efficiency

Throughput #tokens

- Full attention: $\mathcal{O}(m^2)$, #chunks
- Attention by chunks: $\mathcal{O}(kc^2)$, $m = kc$

Benefit #2: Efficiency

Throughput

#tokens

- Full attention: $\mathcal{O}(m^2)$, #chunks length of the chunk
- Attention by chunks: $\mathcal{O}(kc^2), m = kc$

Benefit #2: Efficiency

Throughput

#tokens

- Full attention: $\mathcal{O}(m^2)$, #chunks length of the chunk
- Attention by chunks: $\mathcal{O}(kc^2), m = kc$
- Using a small encoder to encode most context

Benefit #2: Efficiency

Throughput

- Full attention: $\mathcal{O}(\underset{\text{\#tokens}}{m}^2)$,
- Attention by chunks: $\mathcal{O}(\underset{\text{\#chunks}}{k}\underset{\text{length of the chunk}}{c}^2)$, $m = kc$
- Using a small encoder to encode most context

Memory

Benefit #2: Efficiency

Throughput

- Full attention: $\mathcal{O}(\underset{\text{\#tokens}}{m}^2)$,
- Attention by chunks: $\mathcal{O}(\underset{\text{\#chunks}}{k}\underset{\text{length of the chunk}}{c}^2), m = kc$
- Using a small encoder to encode most context

Memory

- Additional context: m tokens. Main input: n tokens.

Benefit #2: Efficiency

Throughput

- Full attention: $\mathcal{O}(\overset{\text{\#tokens}}{m}{}^2)$,
- Attention by chunks: $\mathcal{O}(\underset{\text{\#chunks}}{k}\,\underset{\text{length of the chunk}}{c}{}^2), m = kc$
- Using a small encoder to encode most context

Memory

- Additional context: m tokens. Main input: n tokens.
- Decoder-only: $\mathcal{O}((m+n)Ld_{\text{dec}})$

Benefit #2: Efficiency

Throughput

#tokens

- Full attention: $\mathcal{O}(m^2)$, #chunks length of the chunk
- Attention by chunks: $\mathcal{O}(kc^2)$, $m = kc$
- Using a small encoder to encode most context

Memory

- Additional context: m tokens. Main input: n tokens.
- Decoder-only: $\mathcal{O}((m+n)Ld_{\text{dec}})$

#decoder layers

Benefit #2: Efficiency

Throughput

#tokens

- Full attention: $\mathcal{O}(m^2)$, #chunks length of the chunk
- Attention by chunks: $\mathcal{O}(kc^2)$, $m = kc$
- Using a small encoder to encode most context

Memory

- Additional context: m tokens. Main input: n tokens.
- Decoder-only: $\mathcal{O}((m+n)Ld_{\text{dec}})$
- Ours: $\mathcal{O}(md_{\text{enc}} + nLd_{\text{dec}})$

#decoder layers

Benefit #2: Efficiency

Throughput

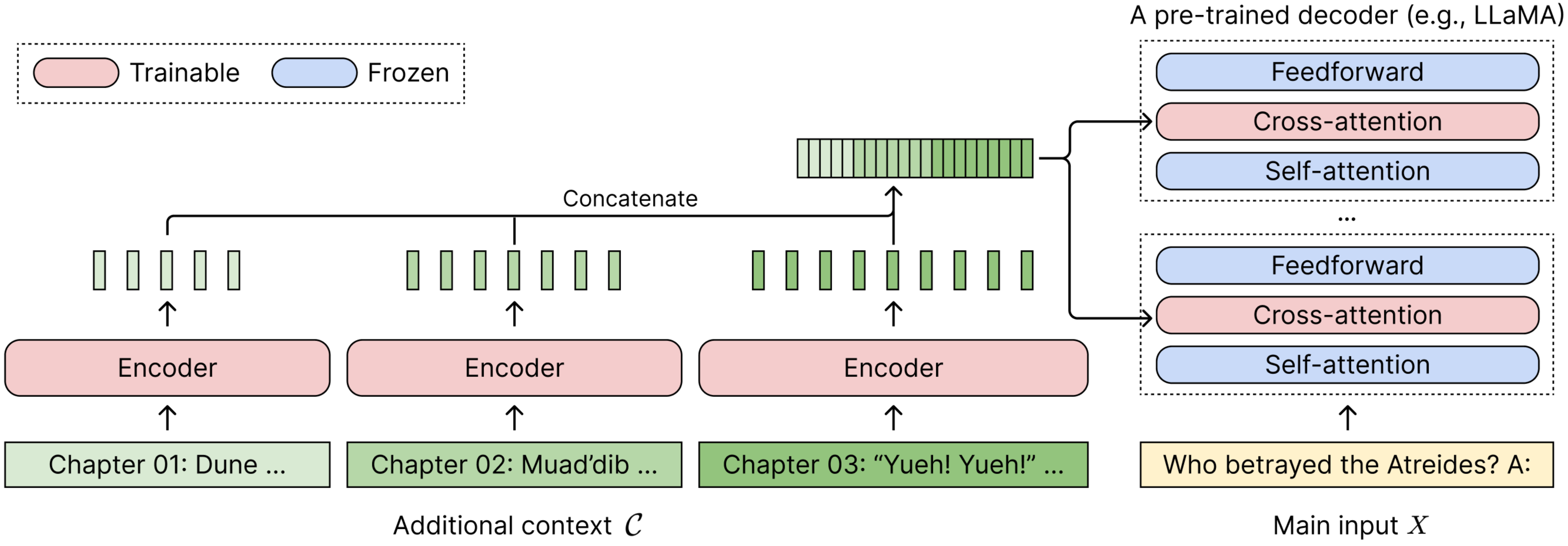
#tokens

- Full attention: $\mathcal{O}(m^2)$, #chunks length of the chunk
- Attention by chunks: $\mathcal{O}(kc^2)$, $m = kc$
- Using a small encoder to encode most context

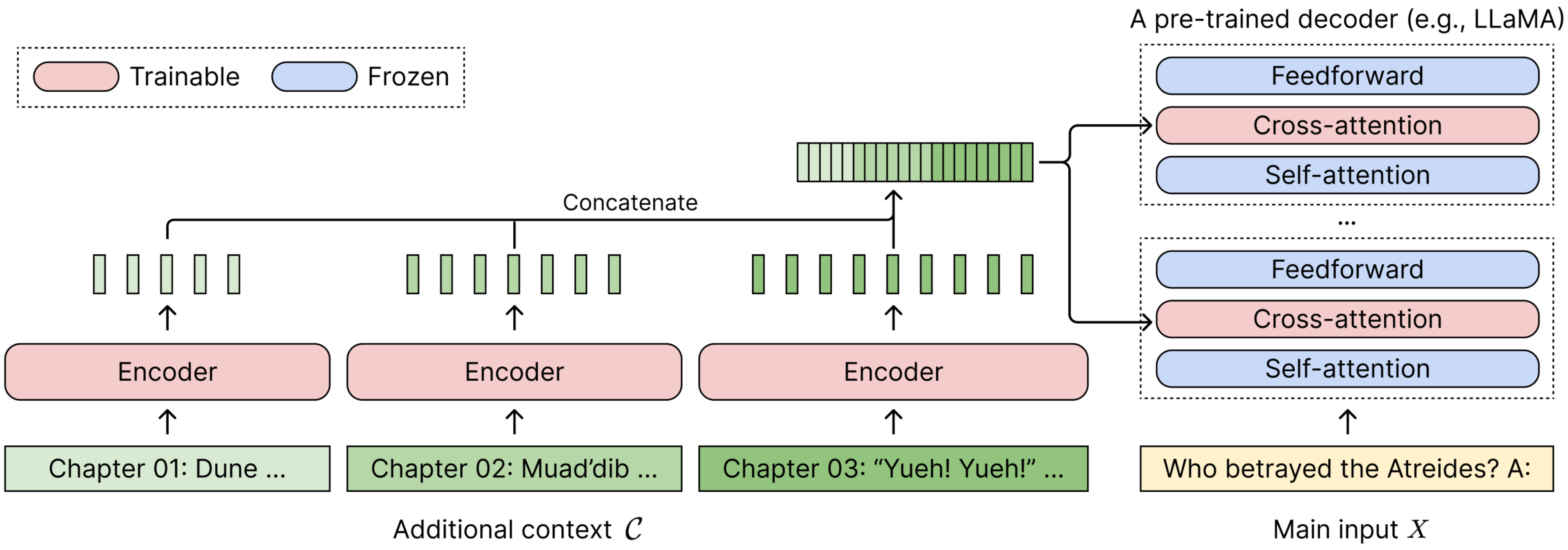
Memory

- Additional context: m tokens. Main input: n tokens.
- Decoder-only: $\mathcal{O}((m+n)Ld_{\text{dec}})$
- Ours: $\mathcal{O}(md_{\text{enc}} + nLd_{\text{dec}})$ #decoder layers
- $m \gg n, d_{\text{dec}} \gg d_{\text{enc}}$

Benefit #3: Low training cost

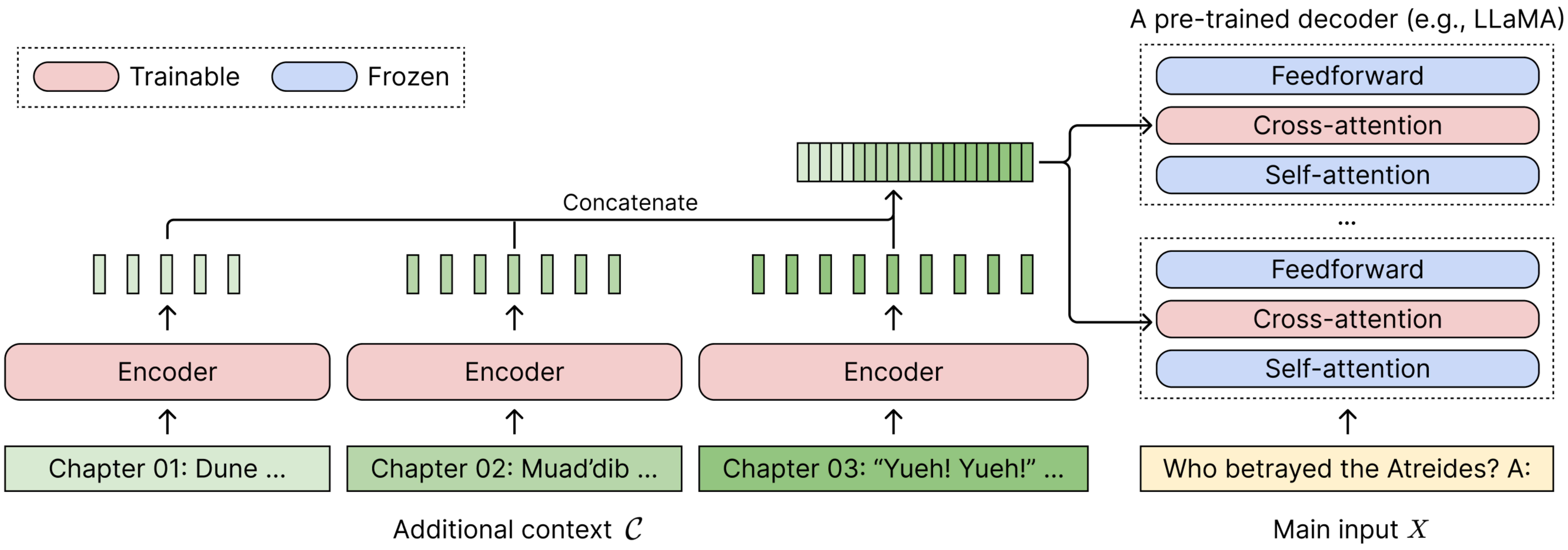


Benefit #3: Low training cost



We only fine-tune the encoder and the cross-attention → can fit on 1 80GB GPU

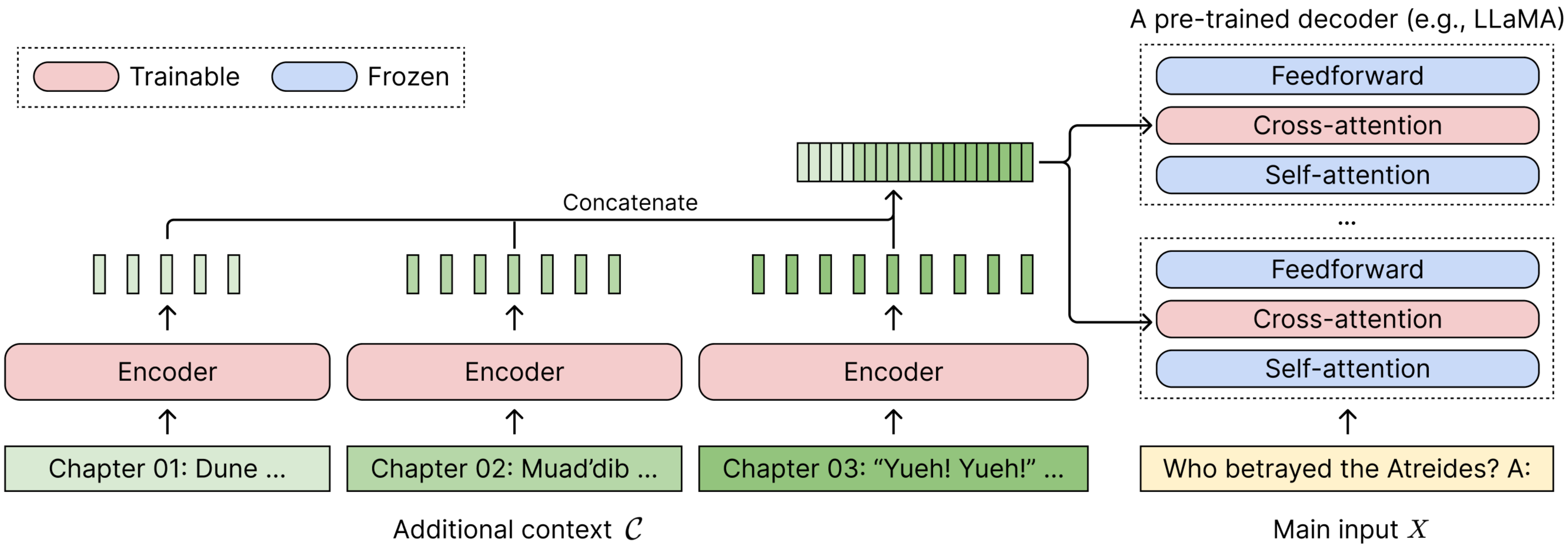
Benefit #3: Low training cost



We only fine-tune the encoder and the cross-attention → can fit on 1 80GB GPU

- Fine-tuning a 7B decoder → at least 4 80GB GPUs

Benefit #3: Low training cost



We only fine-tune the encoder and the cross-attention → can fit on 1 80GB GPU

- Fine-tuning a 7B decoder → at least 4 80GB GPUs

We only fine-tune on 8K sequence length → generalize to 128K

Training data

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)
  - Good diversity

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)
  - Good diversity
  - Lack long-range dependency

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)
  - Good diversity
  - Lack long-range dependency

Domain	Avg #tokens
Arxiv	**16K**
Book	**142K**
C4 (clean web)	0.5K
CC (web)	2K
Github	2K
StackExchange	0.5K
Wikipedia	0.5K

Different domains from RP.

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)
  - Good diversity
  - Lack long-range dependency
- **Length-filter recipe**: long documents from Arxiv and Book (RP-filter)

Domain	Avg #tokens
Arxiv	**16K**
Book	**142K**
C4 (clean web)	0.5K
CC (web)	2K
Github	2K
StackExchange	0.5K
Wikipedia	0.5K

Different domains from RP.

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)
  - Good diversity
  - Lack long-range dependency
- **Length-filter recipe**: long documents from Arxiv and Book (RP-filter)
  - Strong long-range dependency

Domain	Avg #tokens
Arxiv	**16K**
Book	**142K**
C4 (clean web)	0.5K
CC (web)	2K
Github	2K
StackExchange	0.5K
Wikipedia	0.5K

Different domains from RP.

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)
  - Good diversity
  - Lack long-range dependency
- **Length-filter recipe**: long documents from Arxiv and Book (RP-filter)
  - Strong long-range dependency
  - Lack diversity

Domain	Avg #tokens
Arxiv	**16K**
Book	**142K**
C4 (clean web)	0.5K
CC (web)	2K
Github	2K
StackExchange	0.5K
Wikipedia	0.5K

Different domains from RP.

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)
  - Good diversity
  - Lack long-range dependency
- **Length-filter recipe**: long documents from Arxiv and Book (RP-filter)
  - Strong long-range dependency
  - Lack diversity

Domain	Avg #tokens
Arxiv	**16K**
Book	**142K**
C4 (clean web)	0.5K
CC (web)	2K
Github	2K
StackExchange	0.5K
Wikipedia	0.5K

Different domains from RP.

We use a 2:1 mixture of RP-filter and RP-concat,

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training data

We use RedPajama (RP; Together, 2023).

- **Standard recipe:** concatenate the documents to the target length (RP-concat)
  - Good diversity
  - Lack long-range dependency
- **Length-filter recipe**: long documents from Arxiv and Book (RP-filter)
  - Strong long-range dependency
  - Lack diversity

Domain	Avg #tokens
Arxiv	**16K**
Book	**142K**
C4 (clean web)	0.5K
CC (web)	2K
Github	2K
StackExchange	0.5K
Wikipedia	0.5K

Different domains from RP.

We use a 2:1 mixture of RP-filter and RP-concat,
a design choice echoing Together, 2023; Fu et al., 2024.

Together, 2023. RedPajama, a project to create leading open-source models, starts by reproducing LLaMA training dataset of over 1.2 trillion tokens.
Together, 2023. Preparing for the era of 32K context: Early learnings and explorations.
Fu et al., 2024. Data Engineering for Scaling Language Models to 128K Context.

Training details

Training details

Encoder

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training CEPE

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training CEPE

- We train CEPE with LLaMA-2-7B for 20B tokens (1% of LLaMA-2's pre-training budget)

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training CEPE

- We train CEPE with LLaMA-2-7B for 20B tokens (1% of LLaMA-2's pre-training budget)
- Decoder input = **4K** tokens

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training CEPE

- We train CEPE with LLaMA-2-7B for 20B tokens (1% of LLaMA-2's pre-training budget)
- Decoder input = **4K** tokens
- Encoder input = **16 x 256** tokens

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training CEPE

- We train CEPE with LLaMA-2-7B for 20B tokens (1% of LLaMA-2's pre-training budget)
- Decoder input = **4K** tokens
- Encoder input = **16 x 256** tokens

Warmup

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training CEPE

- We train CEPE with LLaMA-2-7B for 20B tokens (1% of LLaMA-2's pre-training budget)
- Decoder input = **4K** tokens
- Encoder input = **16 x 256** tokens

Warmup

We first train the cross-attention to learn to copy

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training CEPE

- We train CEPE with LLaMA-2-7B for 20B tokens (1% of LLaMA-2's pre-training budget)
- Decoder input = **4K** tokens
- Encoder input = **16 x 256** tokens

Warmup

We first train the cross-attention to learn to copy

- We set the encoder and the decoder input to be the same (for 131M tokens)

Training details

Encoder

- A train-from-scratch **435M** bidirectional encoder using masked language modeling
- Same vocabulary as LLaMA

Training CEPE

- We train CEPE with LLaMA-2-7B for 20B tokens (1% of LLaMA-2's pre-training budget)
- Decoder input = **4K** tokens
- Encoder input = **16 x 256** tokens

Warmup

We first train the cross-attention to learn to copy

- We set the encoder and the decoder input to be the same (for 131M tokens)

CEPE for instruction-tuned models

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)

Who is the president of the United States?
A: Donald Trump

Q: What are three scientific advancements?
A: Airline travel, cars, and space travel

Q: Where were the first humans found?
A: East Afrrica

Q: Where did humans emigrate to prior to finding the new world?
A: Asia

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)

Who is the president of the United States?
A: Donald Trump

Q: What are three scientific advancements?
A: Airline travel, cars, and space travel

Q: Where were the first humans found?
A: East Afrrica

Q: Where did humans emigrate to prior to finding the new world?
A: Asia

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)

Who is the president of the United States?
A: Donald Trump

Q: What are three scientific advancements?
A: Airline travel, cars, and space travel

Q: Where were the first humans found?
A: East Afrrica

Q: Where did humans emigrate to prior to finding the new world?
A: Asia

USER	Who is the president of the United States?
ASSISTANT	As of my last update in April 2023, Joe Biden is the President of the United States.

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)
- How to turn a long-context LM to a long-context ***chat*** LM?

Who is the president of the United States?
A: Donald Trump

Q: What are three scientific advancements?
A: Airline travel, cars, and space travel

Q: Where were the first humans found?
A: East Afrrica

Q: Where did humans emigrate to prior to finding the new world?
A: Asia

USER	Who is the president of the United States?
ASSISTANT	As of my last update in April 2023, Joe Biden is the President of the United States.

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)
- How to turn a long-context LM to a long-context ***chat*** LM?
  - Tune a chat LM on long-context data

Who is the president of the United States?
A: Donald Trump

Q: What are three scientific advancements?
A: Airline travel, cars, and space travel

Q: Where were the first humans found?
A: East Afrrica

Q: Where did humans emigrate to prior to finding the new world?
A: Asia

→

USER	Who is the president of the United States?
ASSISTANT	As of my last update in April 2023, Joe Biden is the President of the United States.

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)
- How to turn a long-context LM to a long-context ***chat*** LM?
  - Tune a chat LM on long-context data → lose chat abilities

Who is the president of the United States?
A: Donald Trump

Q: What are three scientific advancements?
A: Airline travel, cars, and space travel

Q: Where were the first humans found?
A: East Afrrica

Q: Where did humans emigrate to prior to finding the new world?
A: Asia

→

USER	Who is the president of the United States?
ASSISTANT	As of my last update in April 2023, Joe Biden is the President of the United States.

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)
- How to turn a long-context LM to a long-context ***chat*** LM?
  - Tune a chat LM on long-context data → lose chat abilities
  - Tune a long-context LM on chat data

Who is the president of the United States?
A: Donald Trump

Q: What are three scientific advancements?
A: Airline travel, cars, and space travel

Q: Where were the first humans found?
A: East Afrrica

Q: Where did humans emigrate to prior to finding the new world?
A: Asia

USER	Who is the president of the United States?
ASSISTANT	As of my last update in April 2023, Joe Biden is the President of the United States.

CEPE for instruction-tuned models

Instruction-tuning / chat-tuning

- Most useful models are fine-tuned on *chat-like data* (often proprietary)
- How to turn a long-context LM to a long-context ***chat*** LM?
  - Tune a chat LM on long-context data → lose chat abilities
  - Tune a long-context LM on chat data → no proprietary data; no long-context chat data

Who is the president of the United States?
A: Donald Trump

Q: What are three scientific advancements?
A: Airline travel, cars, and space travel

Q: Where were the first humans found?
A: East Afrrica

Q: Where did humans emigrate to prior to finding the new world?
A: Asia

USER	Who is the president of the United States?
ASSISTANT	As of my last update in April 2023, Joe Biden is the President of the United States.

CEPE for instruction-tuned models

CEPE for instruction-tuned models

CEPE-Distilled (CEPED)

CEPE for instruction-tuned models

CEPE-Distilled (CEPED)

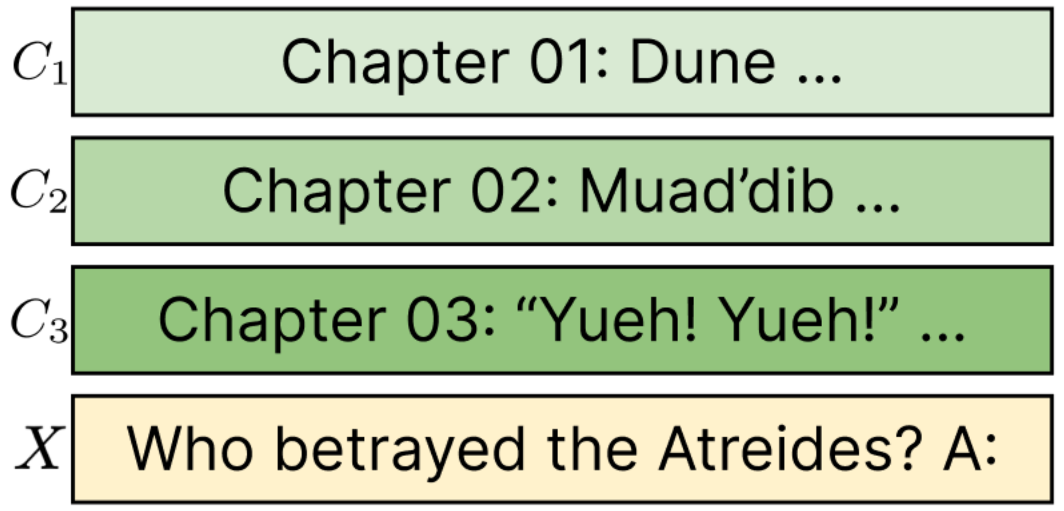
- Use only *unsupervised* long-context data, we can turn a chat model to a *long-context* model

CEPE for instruction-tuned models

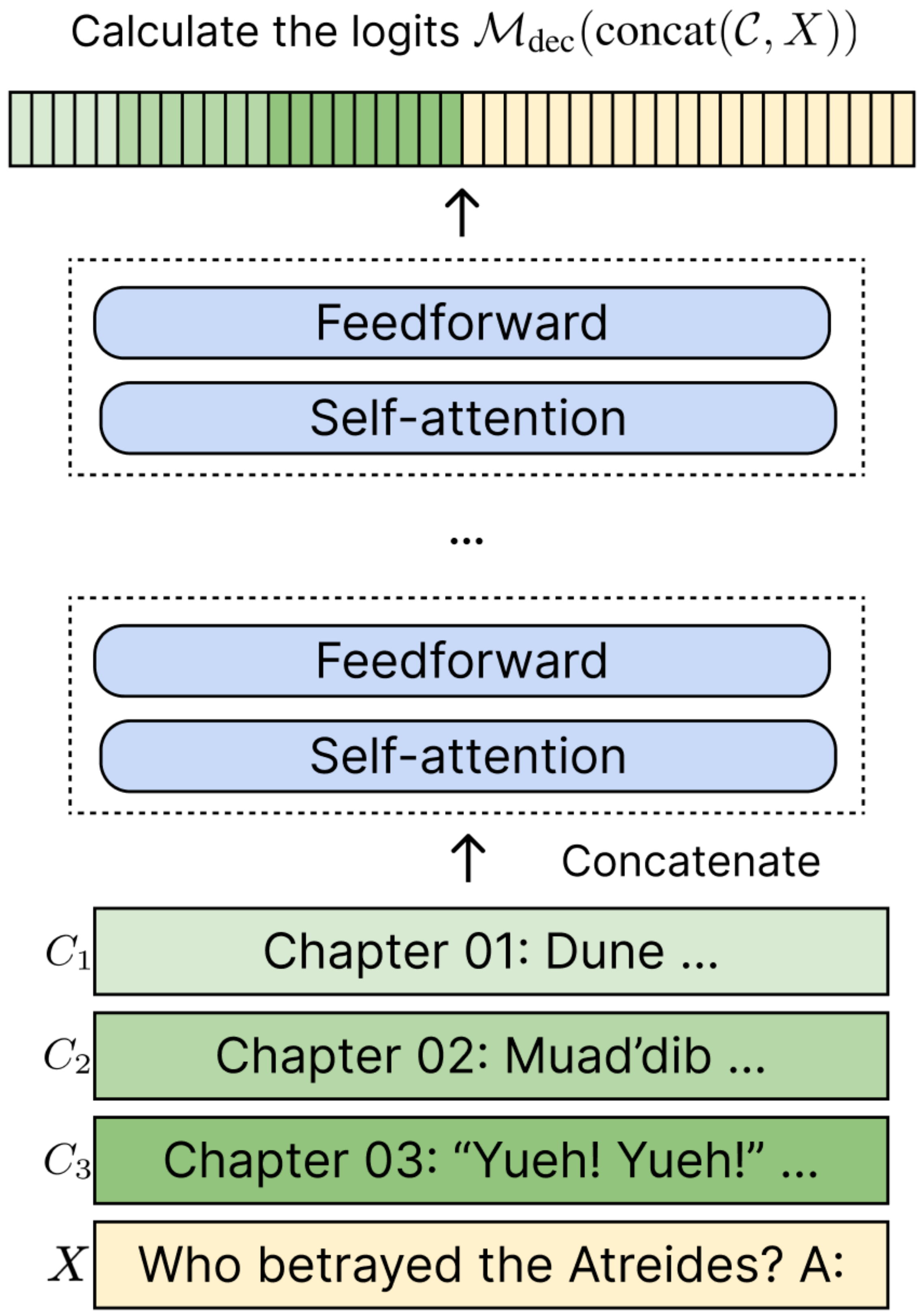
CEPE-Distilled (CEPED)

- Use only *unsupervised* long-context data, we can turn a chat model to a *long-context* model
- Add **an auxiliary distillation loss** to maintain the chat model's chat ability

CEPE-Distilled (CEPED)

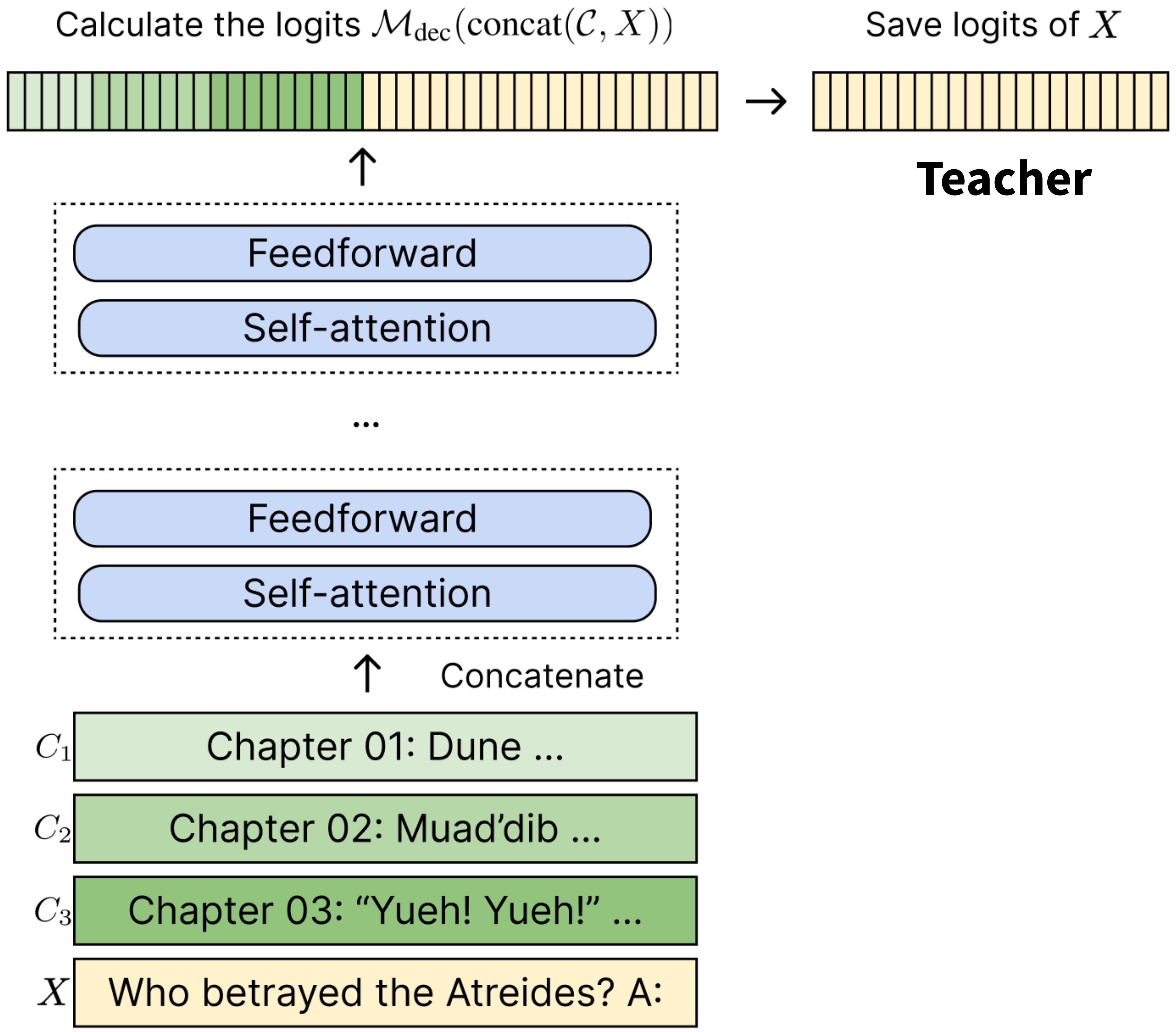


CEPE-Distilled (CEPED)



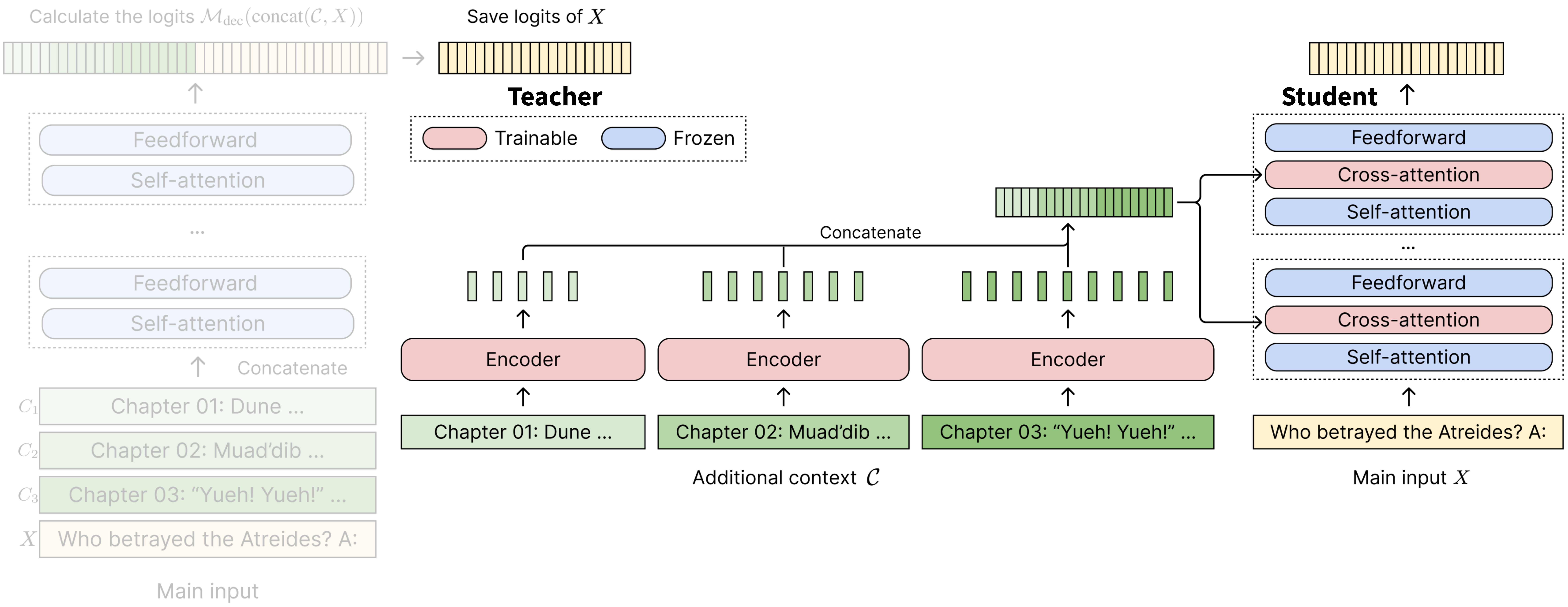
1. Run forward passes w/ the original chat model

CEPE-Distilled (CEPED)



1. Run forward passes w/ the original chat model

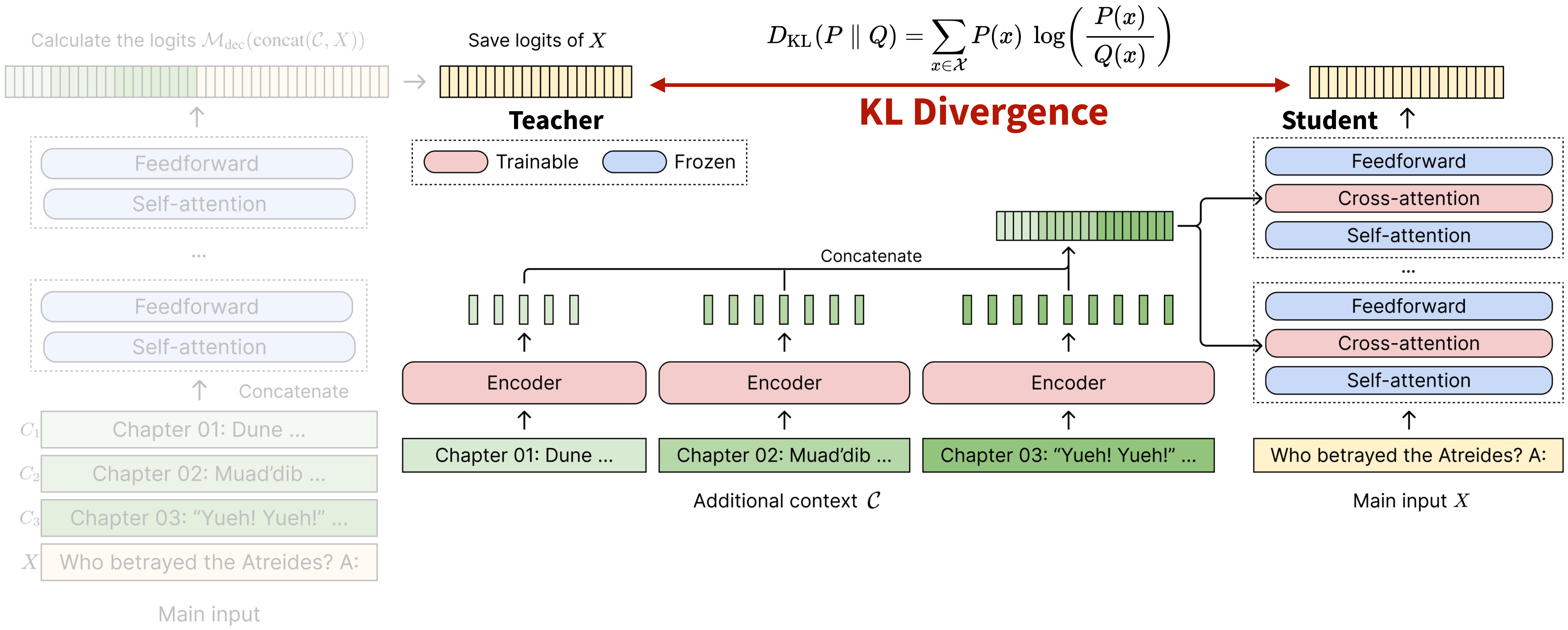
CEPE-Distilled (CEPED)



1. Run forward passes w/ the original chat model

2. Run forward passes w/ the CEPE model

CEPE-Distilled (CEPED)



1. Run forward passes w/ the original chat model

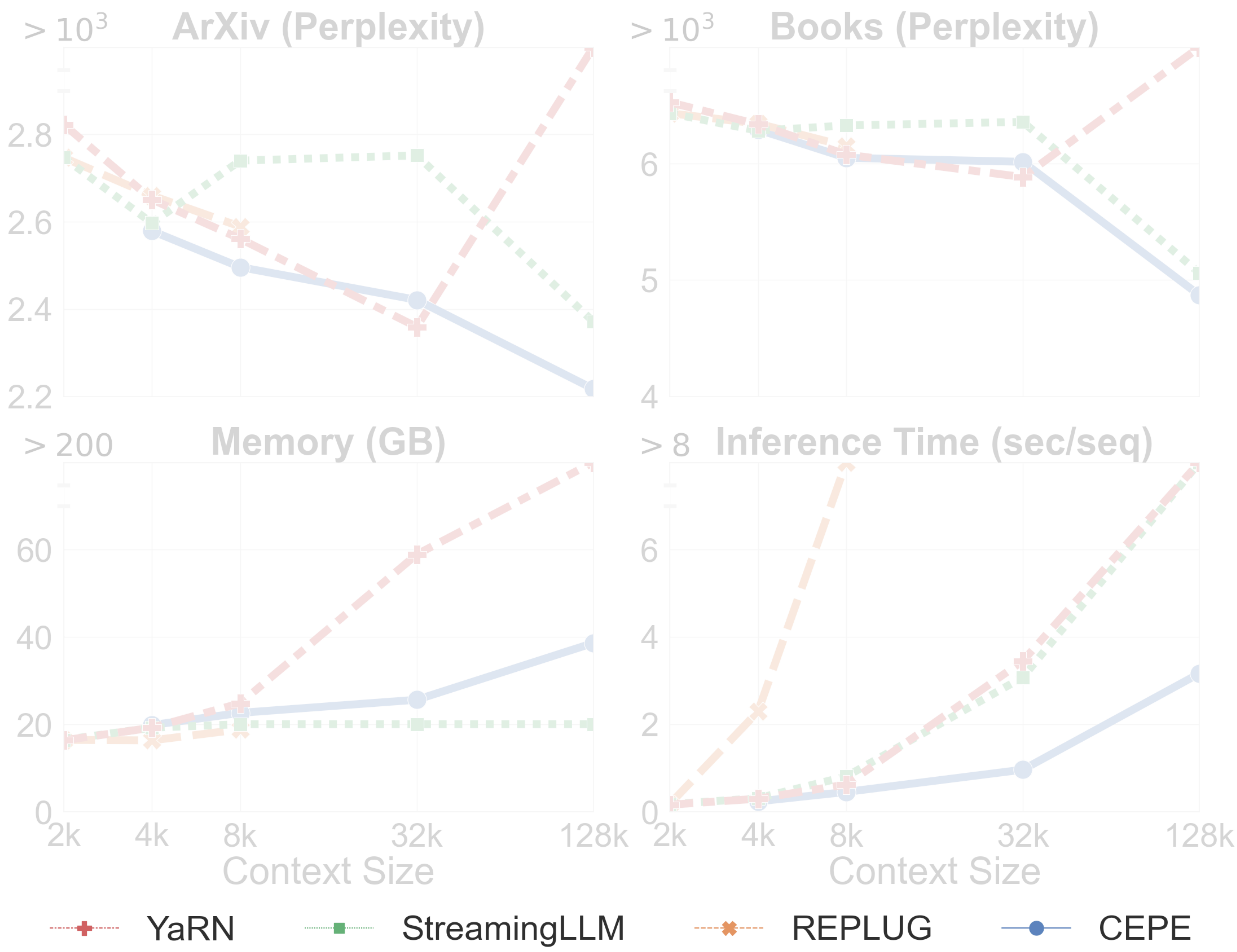
2. Run forward passes w/ the CEPE model

3. Train with KL Divergence loss + cross-entropy loss

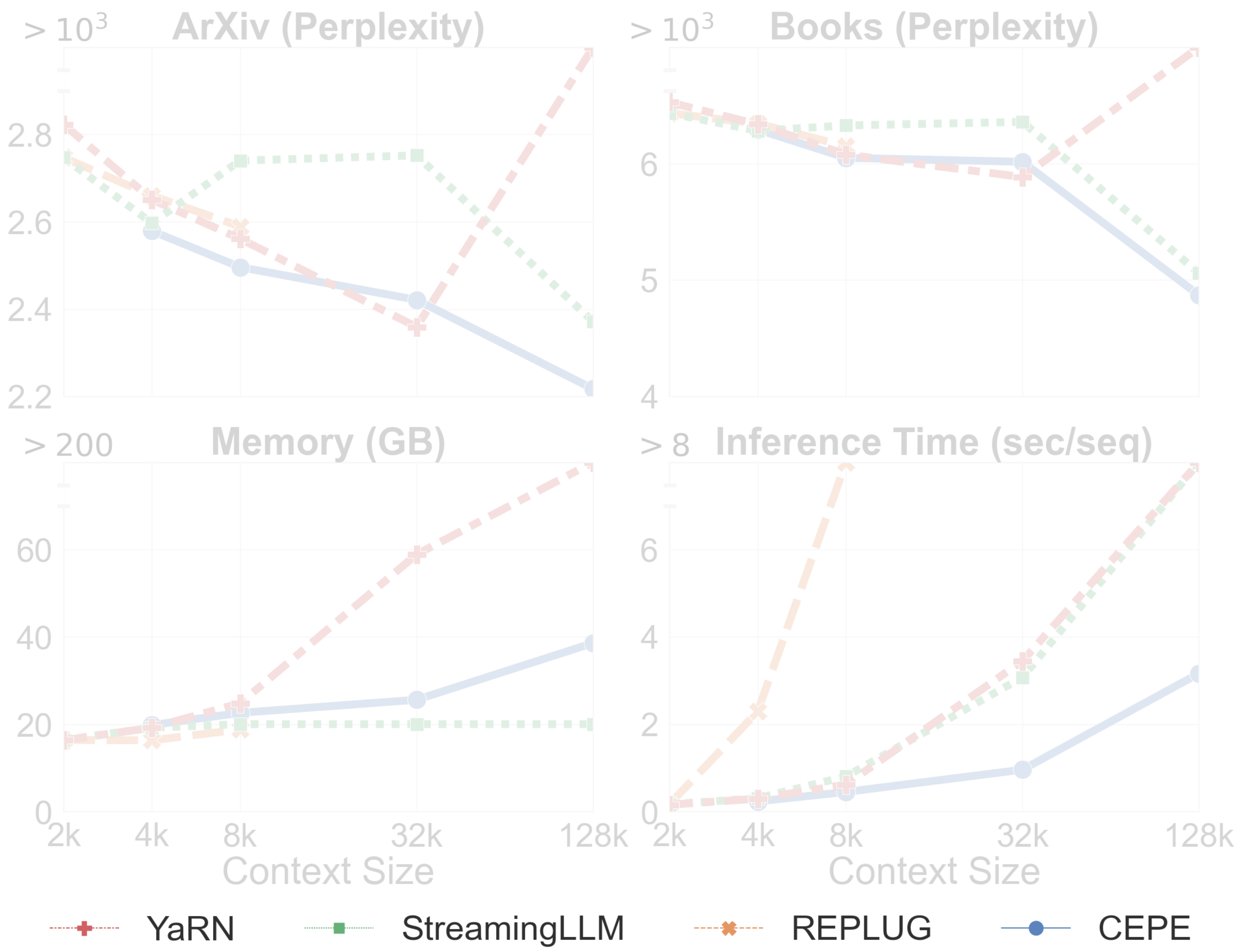
Evaluation

- Long-context language modeling
- Retrieval-augmented applications
- In-context learning
- Chat model evaluation

Evaluation - long-context language modeling

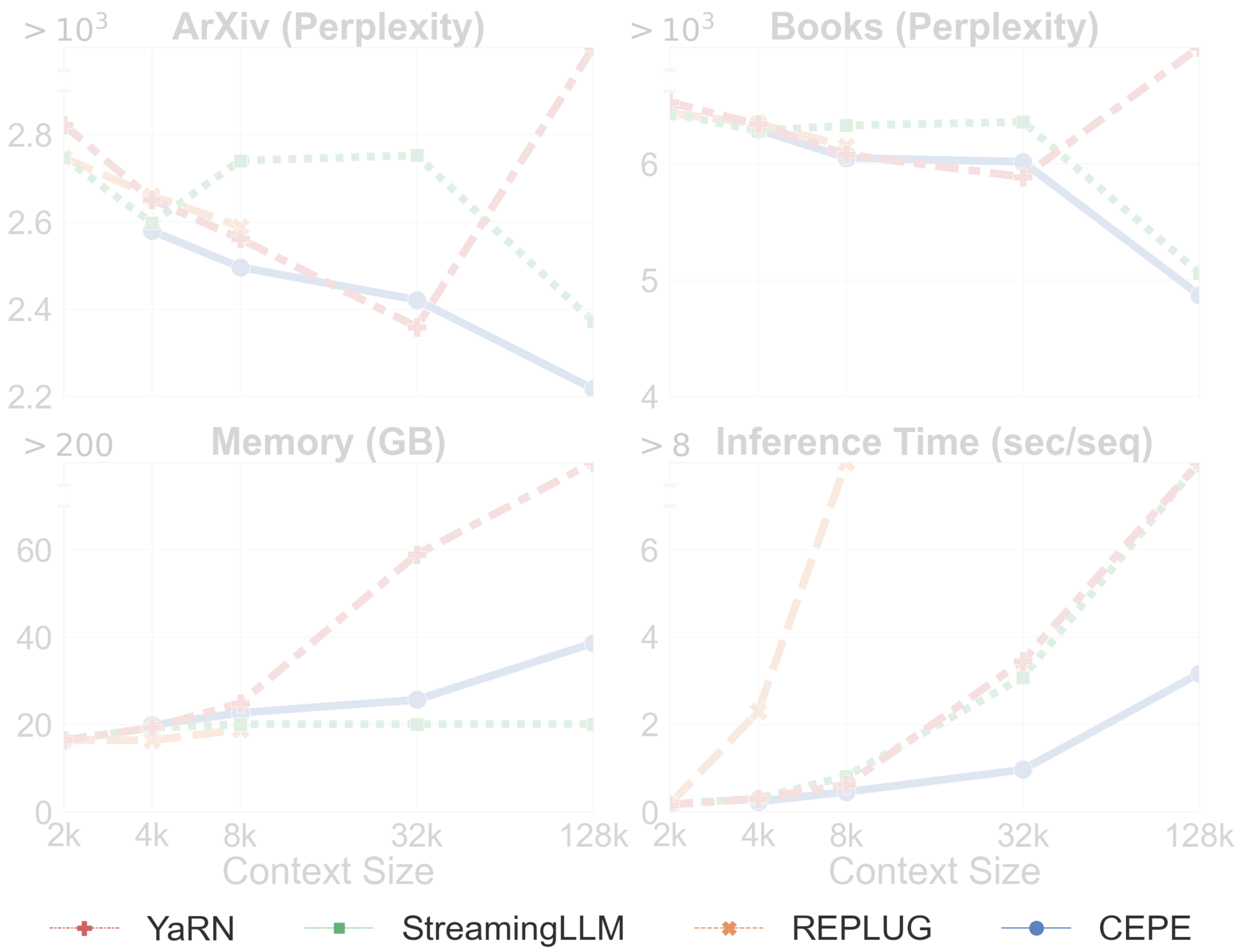


Evaluation - long-context language modeling



Positional interpolation

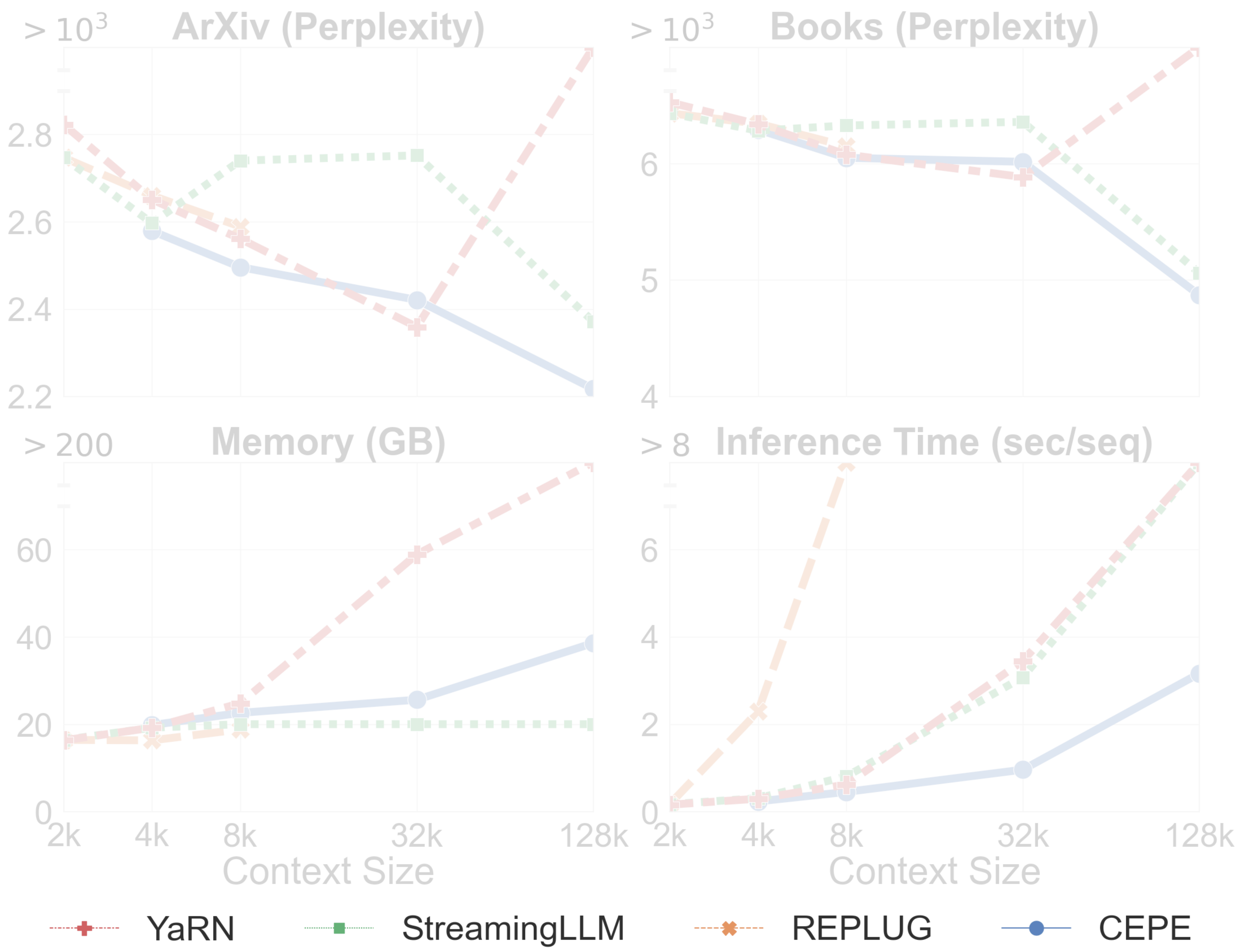
Evaluation - long-context language modeling



Positional interpolation

KV-cache dropping

Evaluation - long-context language modeling

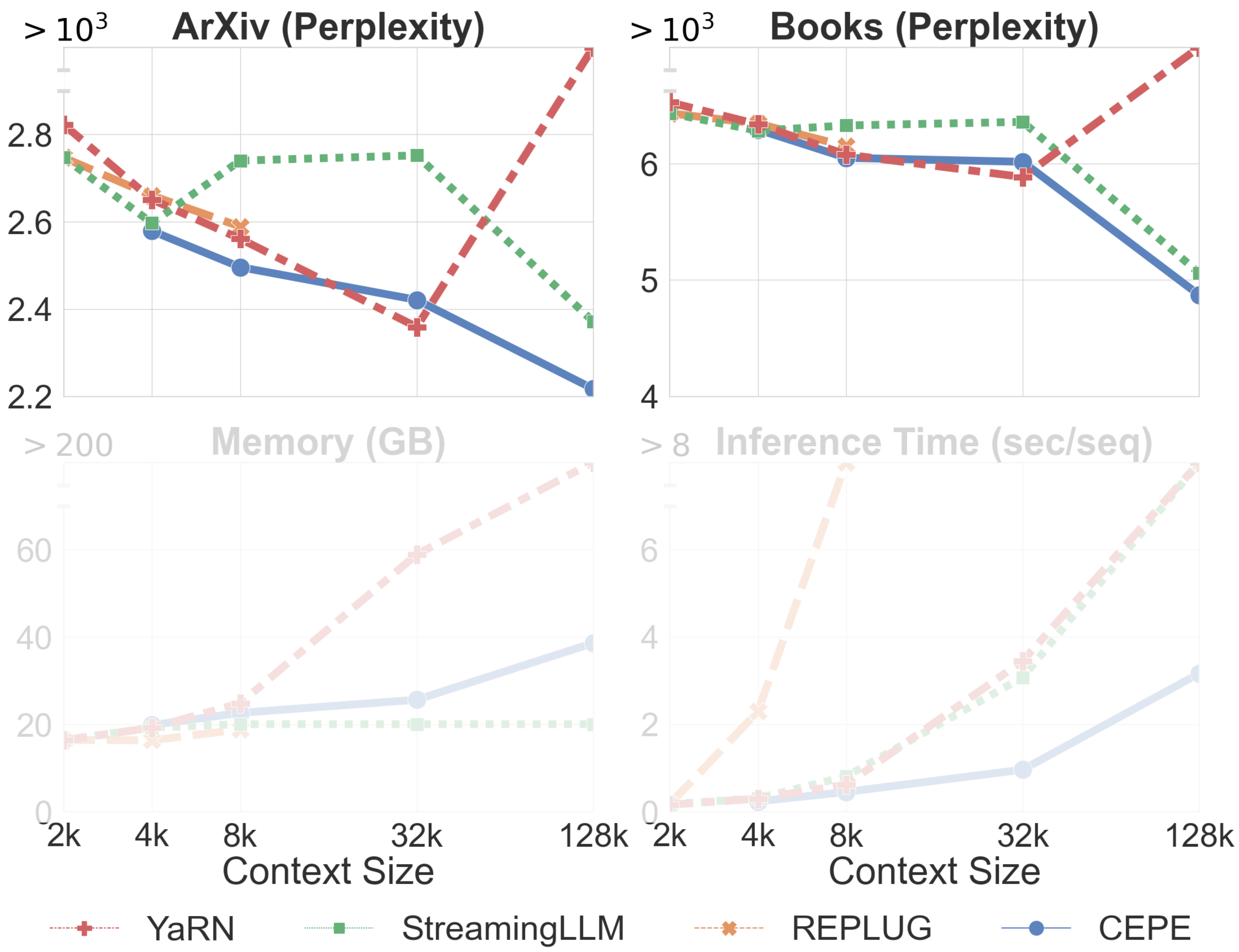


Positional interpolation

KV-cache dropping

Retrieval for long-context

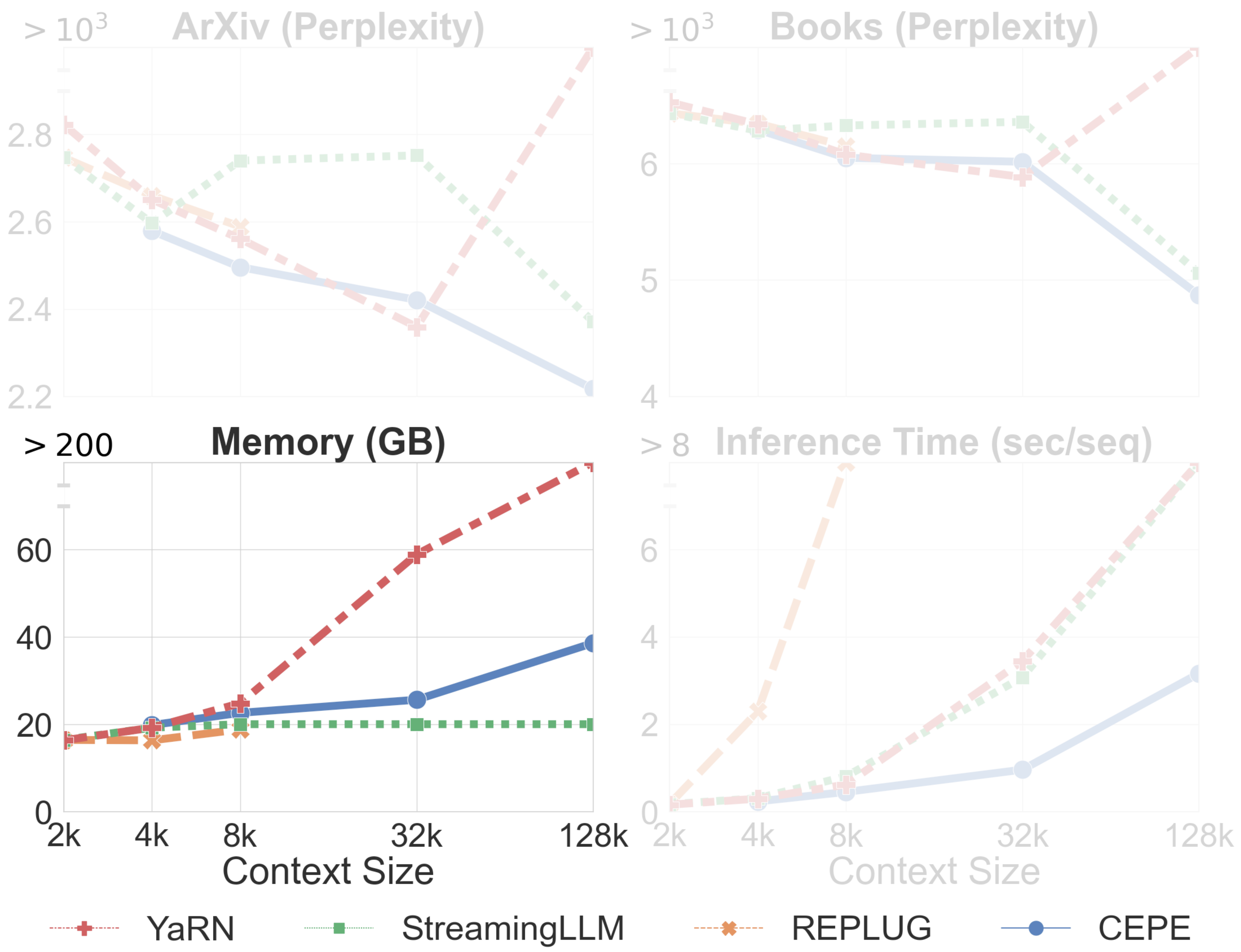
Evaluation - long-context language modeling



Performance

CEPE continues to improve perplexity with more context (only trained on 8K)

Evaluation - long-context language modeling



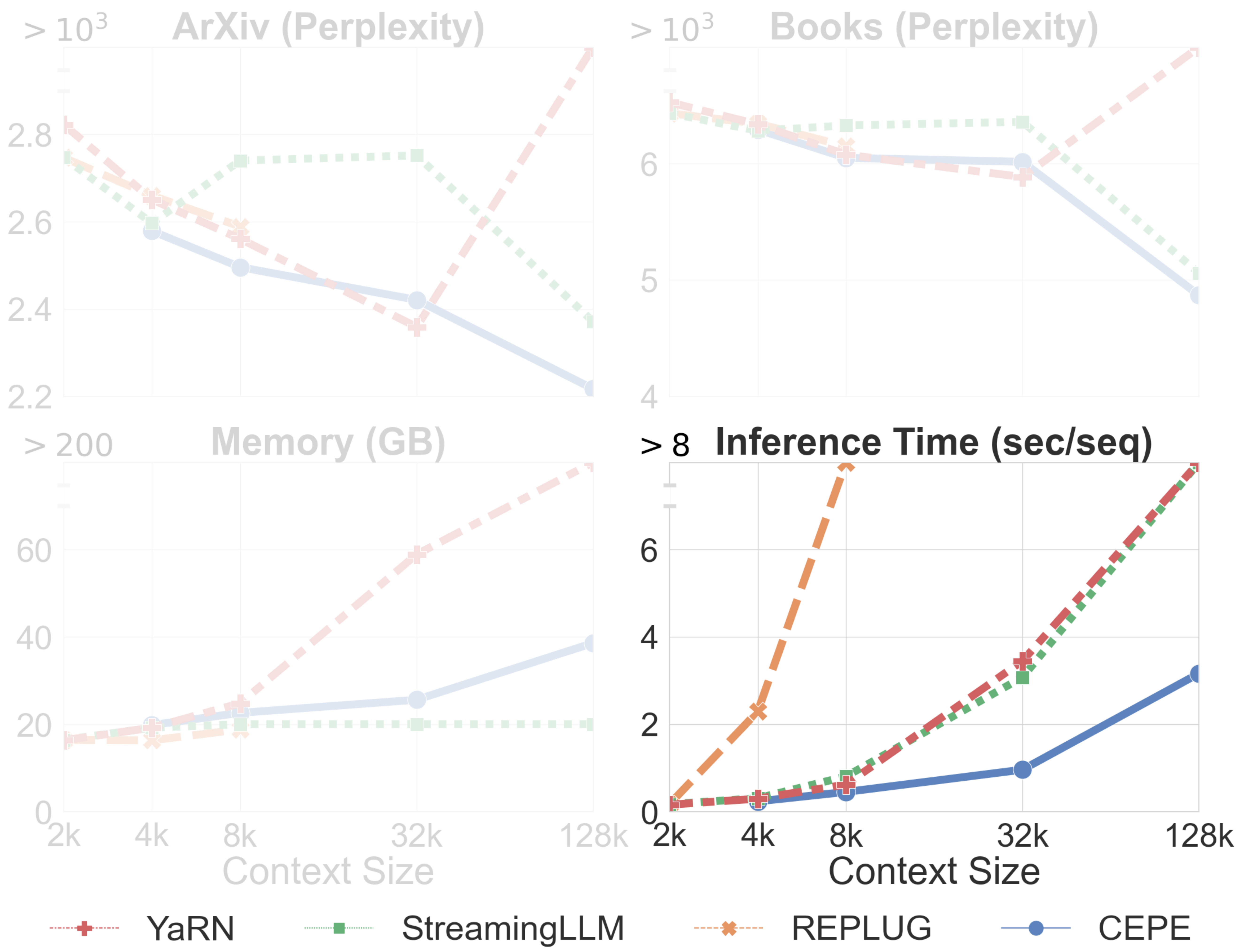
Performance

CEPE continues to improve perplexity with more context (only trained on 8K)

Memory

CEPE maintains a low memory usage (1/6 of full attention)

Evaluation - long-context language modeling



Performance

CEPE continues to improve perplexity with more context (only trained on 8K)

Memory

CEPE maintains a low memory usage (1/6 of full attention)

Throughput

CEPE achieves the highest throughput (10x of full attention)

Evaluation - retrieval-augmented applications

Evaluation - retrieval-augmented applications

Open-domain question answering

Evaluation - retrieval-augmented applications

Open-domain question answering

Passage k: …

Passage k-1: …

…

Passage 1: …

Question: Who is the lead actor of "Dune: Part Two"

Evaluation - retrieval-augmented applications

Open-domain question answering

Retrieved from Wikipedia based on the question

(using a dense retriever)

Passage k: …

Passage k-1: …

…

Passage 1: …

Question: Who is the lead actor of "Dune: Part Two"

Evaluation - retrieval-augmented applications

Open-domain question answering

Retrieved from Wikipedia based on the question

(using a dense retriever)

Passage k: …

Passage k-1: …

…

Passage 1: …

Question: Who is the lead actor of "Dune: Part Two"

LM

Evaluation - retrieval-augmented applications

Open-domain question answering

Retrieved from Wikipedia based on the question

(using a dense retriever)

Passage k: …

Passage k-1: …

…

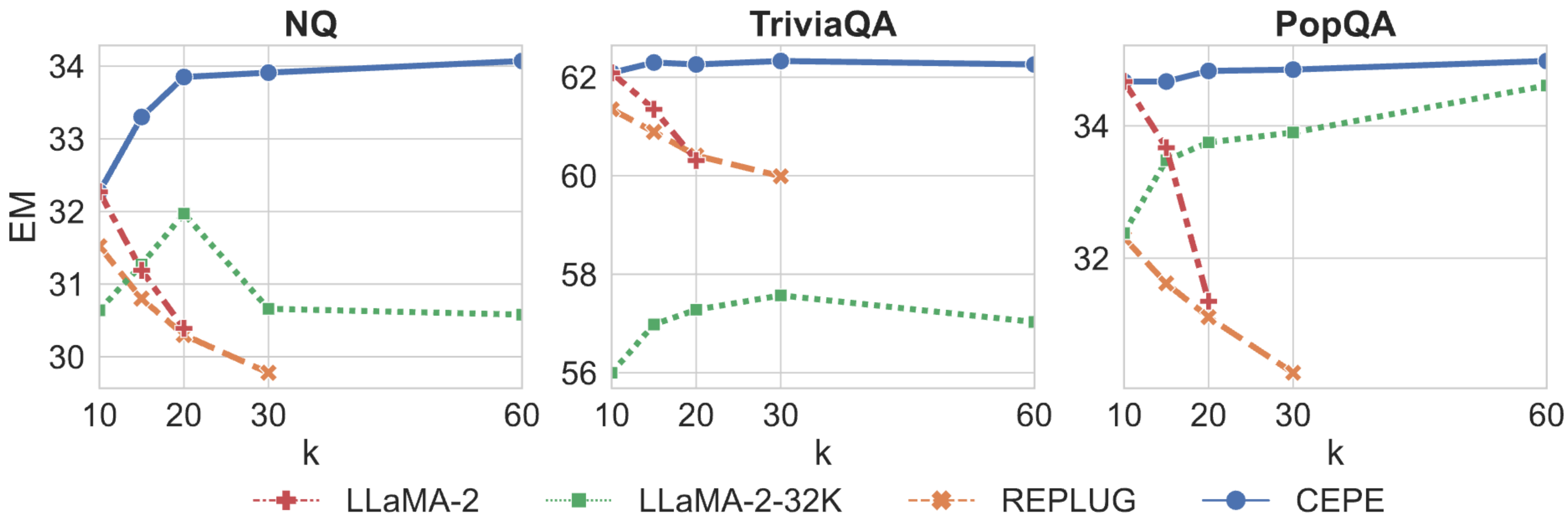
Passage 1: …

Question: Who is the lead actor of “Dune: Part Two”

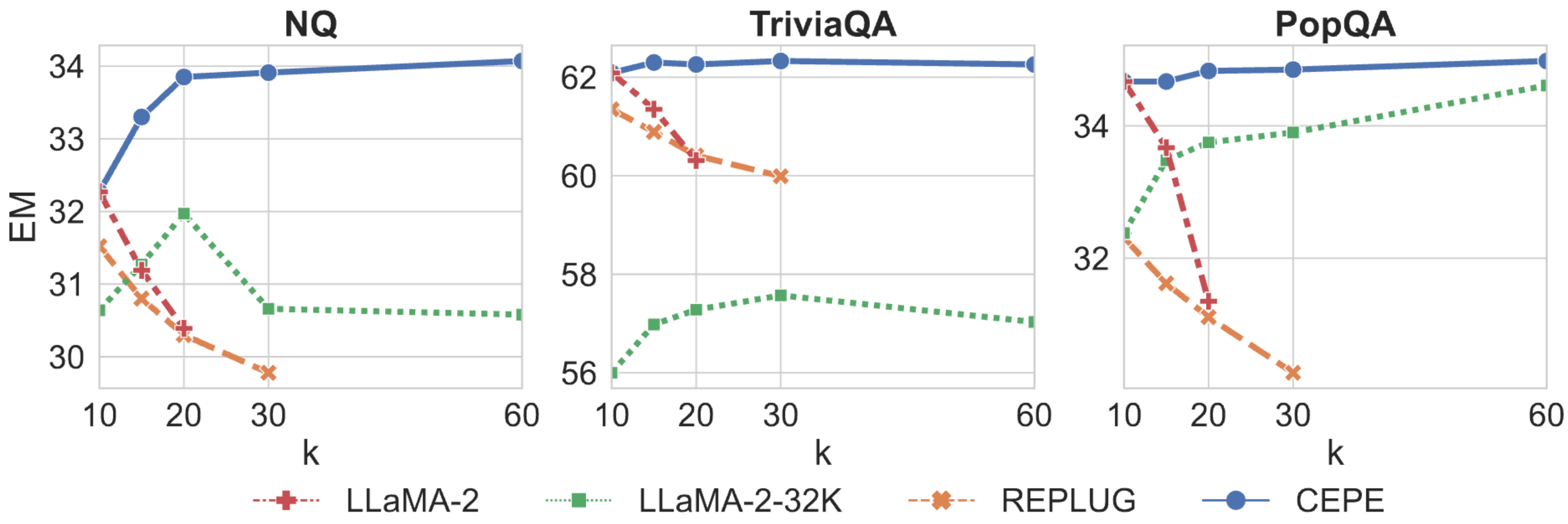
LM → *Timothée Chalamet*

Evaluation - retrieval-augmented applications

Evaluation - retrieval-augmented applications



Evaluation - retrieval-augmented applications



CEPE extrapolates well to more retrieved passages without getting distracted
(also more efficient)

Evaluation — in-context learning

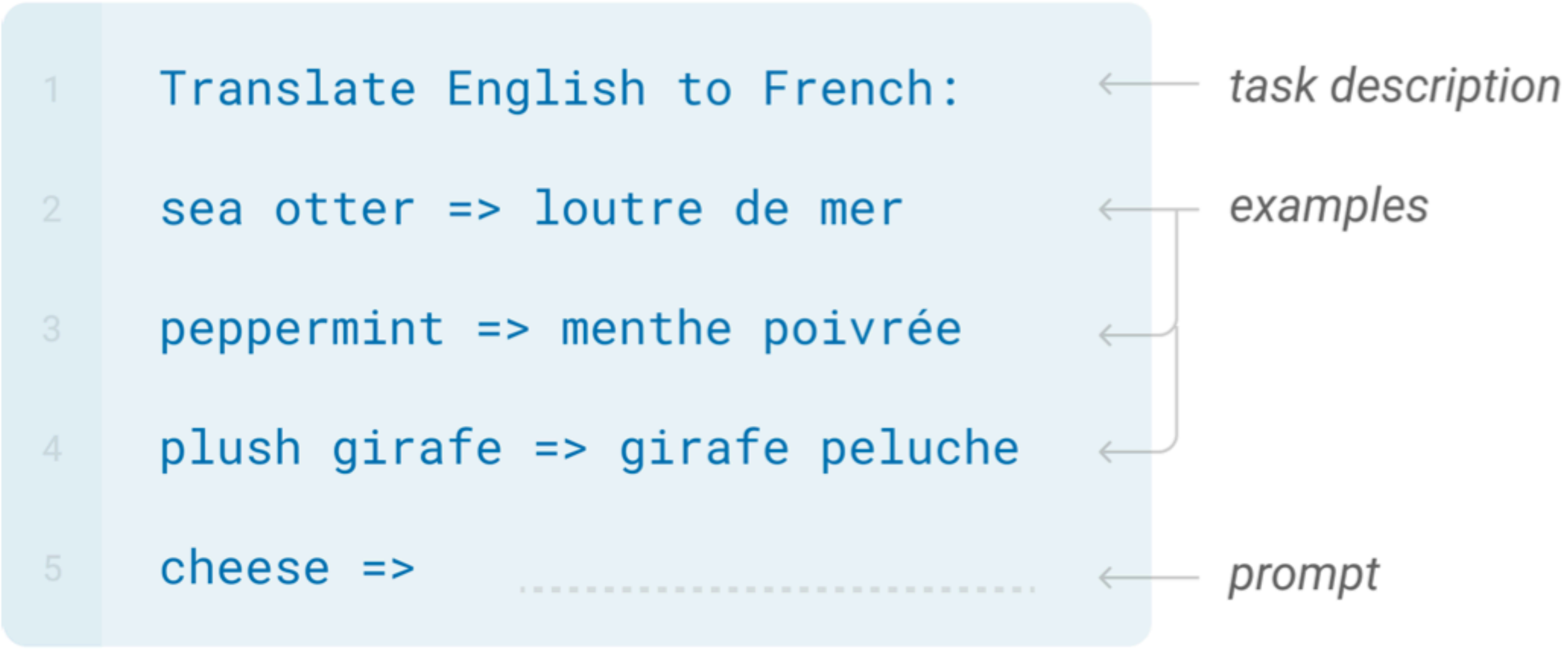
Brown et al., 2020. Language Models are Few-Shot Learners.
Rong, 2021. Extrapolating to Unnatural Language Processing with GPT-3's In-context Learning: The Good, the Bad, and the Mysterious.

Evaluation — in-context learning

An “emerging” ability of large language models: **in-context learning**

Brown et al., 2020. Language Models are Few-Shot Learners.
Rong, 2021. Extrapolating to Unnatural Language Processing with GPT-3's In-context Learning: The Good, the Bad, and the Mysterious.

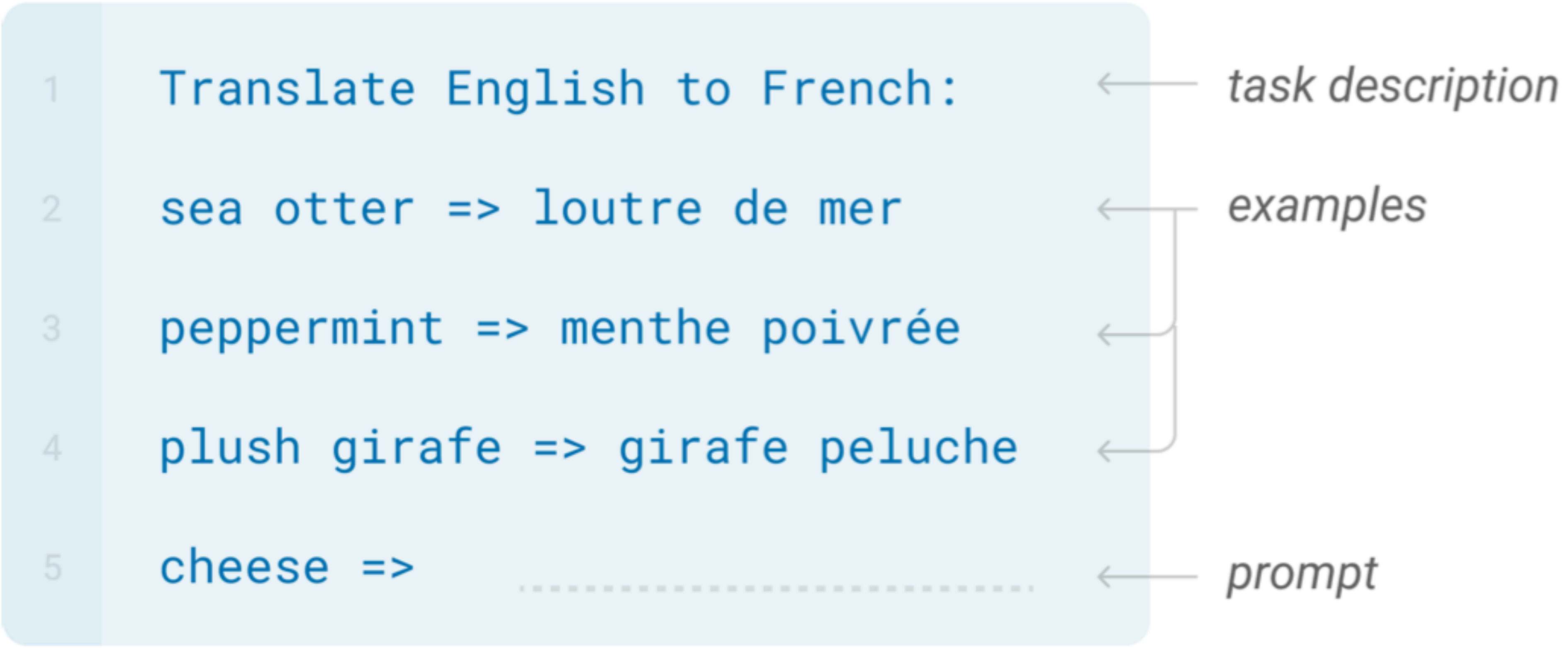
Evaluation — in-context learning

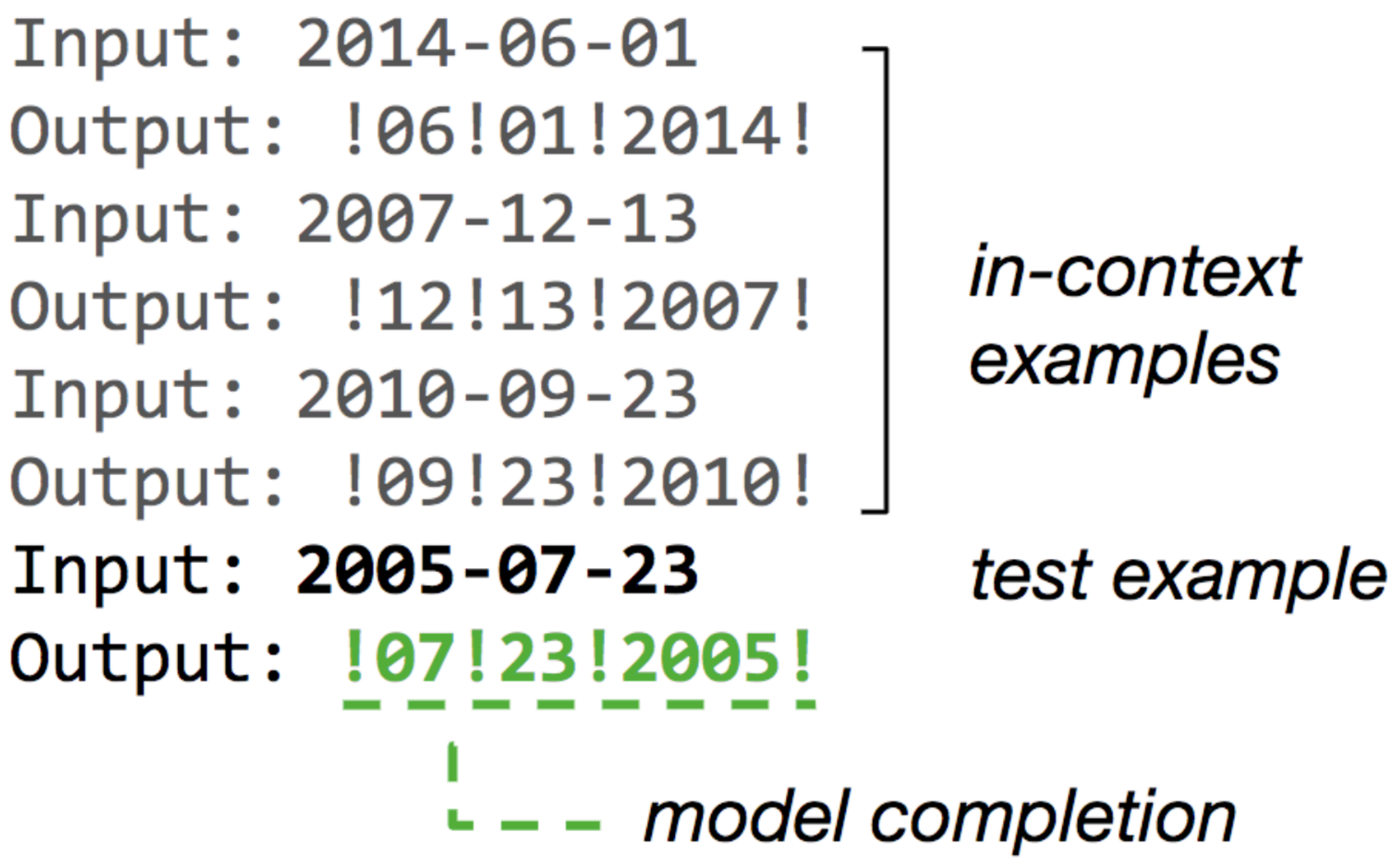


An “emerging” ability of large language models: **in-context learning**

Brown et al., 2020. Language Models are Few-Shot Learners.
Rong, 2021. Extrapolating to Unnatural Language Processing with GPT-3's In-context Learning: The Good, the Bad, and the Mysterious.

Evaluation — in-context learning

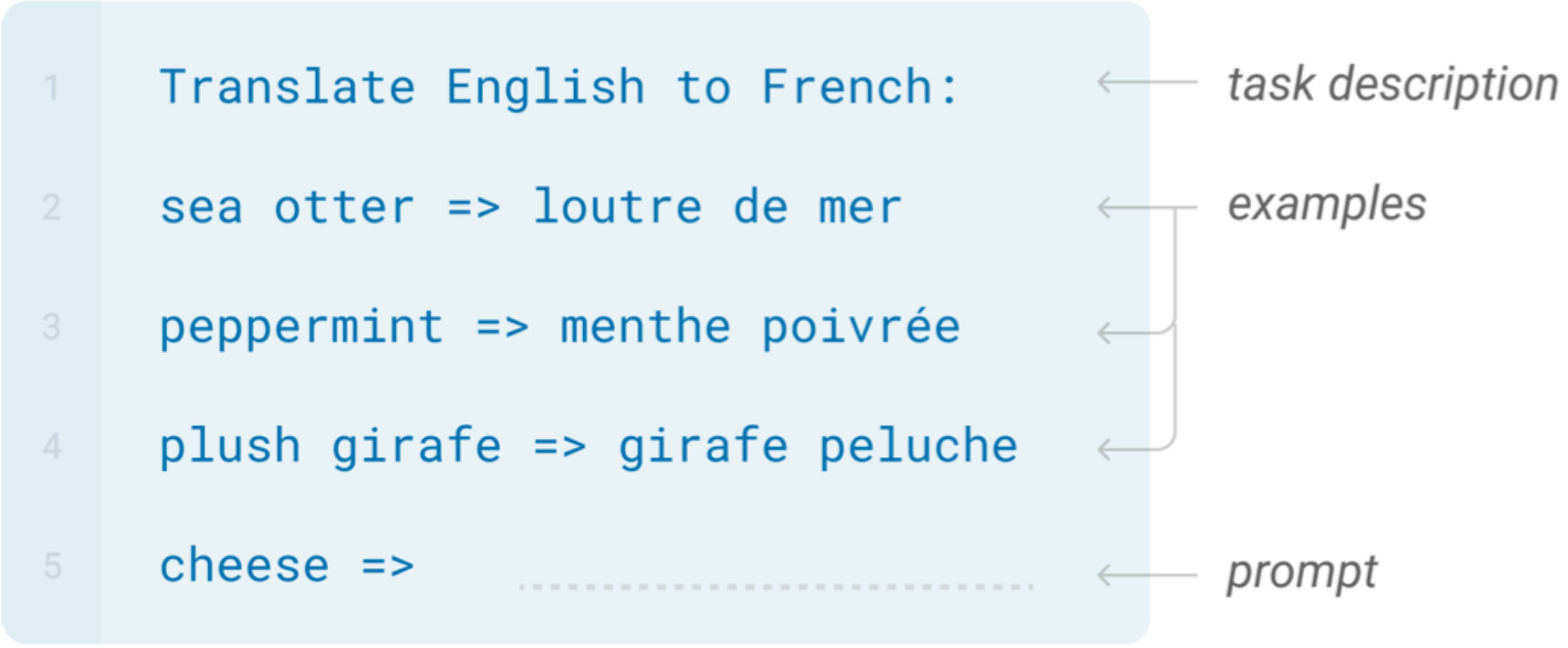


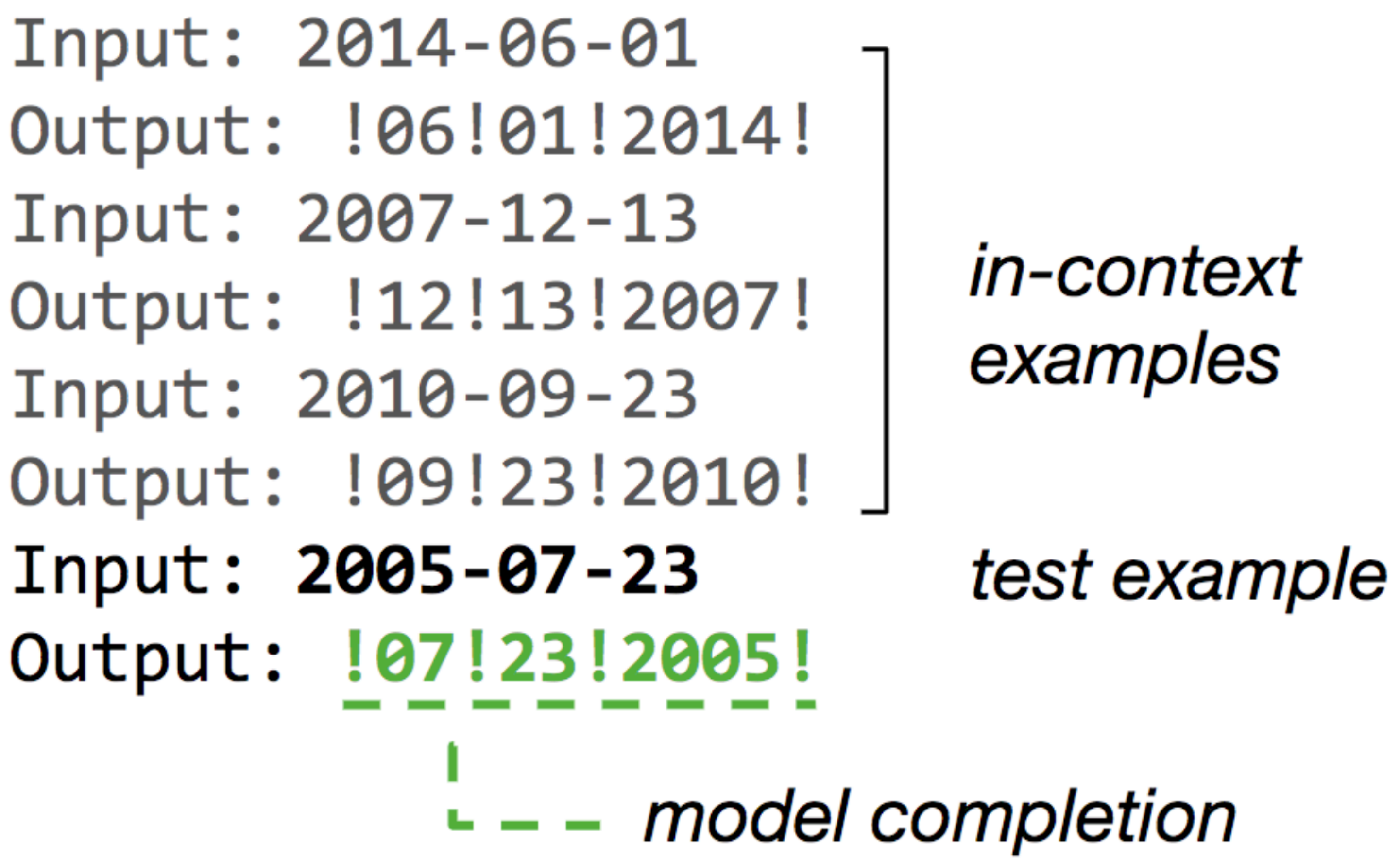


An "emerging" ability of large language models: **in-context learning**

Brown et al., 2020. Language Models are Few-Shot Learners.
Rong, 2021. Extrapolating to Unnatural Language Processing with GPT-3's In-context Learning: The Good, the Bad, and the Mysterious.

Evaluation — in-context learning





An "emerging" ability of large language models: **in-context learning**

Is the *cross-attention* good enough for using in-context examples?

Evaluation — in-context learning

	k	SST2	MR	AGNews	SST5	TREC	TREC-F	DBPedia	NLU-S	NLU-I	BANKING	CLINIC
LLaMA-2	2	89.1	96.7	72.7	3.9	**48.0**	16.7	**94.0**	42.3	22.3	38.4	59.1
+ CEPE	2 + 18	90.7	**98.4**	71.9	**46.7**	47.1	22.8	**94.0**	**48.9**	30.4	42.5	62.4
	2 + 38	**92.9**	98.0	**73.2**	45.5	47.5	**25.1**	93.3	48.8	**31.6**	**46.0**	**62.8**
LLaMA-2†	40	94.3	98.7	74.7	52.3	87.7	54.8	95.1	76.7	62.1	50.4	72.0

Evaluation — in-context learning

#examples in the decoder

	k	SST2	MR	AGNews	SST5	TREC	TREC-F	DBPedia	NLU-S	NLU-I	BANKING	CLINIC
LLAMA-2	2	89.1	96.7	72.7	3.9	**48.0**	16.7	**94.0**	42.3	22.3	38.4	59.1
+ CEPE	2 + 18	90.7	**98.4**	71.9	**46.7**	47.1	22.8	**94.0**	**48.9**	30.4	42.5	62.4
	2 + 38	**92.9**	98.0	**73.2**	45.5	47.5	**25.1**	93.3	48.8	**31.6**	**46.0**	**62.8**
LLAMA-2†	40	94.3	98.7	74.7	52.3	87.7	54.8	95.1	76.7	62.1	50.4	72.0

Evaluation — in-context learning

#examples in the decoder

#examples in the encoder

	k	SST2	MR	AGNews	SST5	TREC	TREC-F	DBPedia	NLU-S	NLU-I	BANKING	CLINIC
LLaMA-2	2	89.1	96.7	72.7	3.9	**48.0**	16.7	**94.0**	42.3	22.3	38.4	59.1
+ CEPE	2 + 18	90.7	**98.4**	71.9	**46.7**	47.1	22.8	**94.0**	**48.9**	30.4	42.5	62.4
	2 + 38	**92.9**	98.0	**73.2**	45.5	47.5	**25.1**	93.3	48.8	**31.6**	**46.0**	**62.8**
LLaMA-2†	40	94.3	98.7	74.7	52.3	87.7	54.8	95.1	76.7	62.1	50.4	72.0

Evaluation — in-context learning

	k	SST2	MR	AGNews	SST5	TREC	TREC-F	DBPedia	NLU-S	NLU-I	BANKING	CLINIC
LLaMA-2	2	89.1	96.7	72.7	3.9	**48.0**	16.7	**94.0**	42.3	22.3	38.4	59.1
+ CEPE	2 + 18	90.7	**98.4**	71.9	**46.7**	47.1	22.8	**94.0**	**48.9**	30.4	42.5	62.4
	2 + 38	**92.9**	98.0	**73.2**	45.5	47.5	**25.1**	93.3	48.8	**31.6**	**46.0**	**62.8**
LLaMA-2†	40	94.3	98.7	74.7	52.3	87.7	54.8	95.1	76.7	62.1	50.4	72.0

CEPE can perform in-context learning using demonstrations in the encoder

Evaluation — in-context learning

	k	SST2	MR	AGNews	SST5	TREC	TREC-F	DBPedia	NLU-S	NLU-I	BANKING	CLINIC
LLaMA-2	2	89.1	96.7	72.7	3.9	**48.0**	16.7	**94.0**	42.3	22.3	38.4	59.1
+ CEPE	2 + 18	90.7	**98.4**	71.9	**46.7**	47.1	22.8	**94.0**	**48.9**	30.4	42.5	62.4
	2 + 38	**92.9**	98.0	**73.2**	45.5	47.5	**25.1**	93.3	48.8	**31.6**	**46.0**	**62.8**
LLaMA-2†	40	94.3	98.7	74.7	52.3	87.7	54.8	95.1	76.7	62.1	50.4	72.0

CEPE can perform in-context learning using demonstrations in the encoder

… though the performance still lags behind putting demonstrations in the decoder

Evaluation — chat model evaluation

		Question Answering			Summarization		
	Total tokens	NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLAMA-2-CHAT	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLAMA-2-32K INSTRUCT	32K	12.2	18.1	41.6	19.9	10.0	10.3

Evaluation — chat model evaluation

Long context (books, government report, papers)

	Total tokens	Question Answering			Summarization		
		NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLaMA-2-Chat	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLaMA-2-32K Instruct	32K	12.2	18.1	41.6	19.9	10.0	10.3

Evaluation — chat model evaluation

Long context (books, government report, papers)

Zero-shot (no training; no in-context examples)

	Total tokens	Question Answering			Summarization		
		NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLAMA-2-CHAT	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLAMA-2-32K INSTRUCT	32K	12.2	18.1	41.6	19.9	10.0	10.3

Evaluation — chat model evaluation

Long context (books, government report, papers)

Zero-shot (no training; no in-context examples)

The model needs to understand “instructions”

	Total tokens	Question Answering			Summarization		
		NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLAMA-2-CHAT	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLAMA-2-32K INSTRUCT	32K	12.2	18.1	41.6	19.9	10.0	10.3

Evaluation — chat model evaluation

		Question Answering			Summarization		
	Total tokens	NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLAMA-2-CHAT	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLAMA-2-32K INSTRUCT	32K	12.2	18.1	41.6	19.9	10.0	10.3

Evaluation — chat model evaluation

Tokens in the decoder

	Total tokens	Question Answering			Summarization		
		NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLAMA-2-CHAT	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLAMA-2-32K INSTRUCT	32K	12.2	18.1	41.6	19.9	10.0	10.3

Evaluation — chat model evaluation

Tokens in the decoder Tokens in the encoder

	Total tokens	Question Answering			Summarization		
		NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLaMA-2-Chat	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLaMA-2-32K Instruct	32K	12.2	18.1	41.6	19.9	10.0	10.3

Evaluation — chat model evaluation

	Total tokens	Question Answering			Summarization		
		NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLaMA-2-Chat	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLaMA-2-32K Instruct	32K	12.2	18.1	41.6	19.9	10.0	10.3

CEPE can utilize the long context and boost the QA/summarization performance

Evaluation — chat model evaluation

	Total tokens	Question Answering			Summarization		
		NQA	Qspr	QALT	GvRp	SSFD	QMSum
LLaMA-2-Chat	2K	17.1	14.6	28.6	16.0	16.4	19.3
+ CEPED	2K + 2K	19.5	**20.5**	**30.2**	**16.5**	16.4	**19.6**
	2K + 30K	21.6	19.9	29.6	15.8	**16.7**	19.5
	2K + All	**21.9**	19.9	29.6	15.9	**16.7**	19.5
LLaMA-2-32K Instruct	32K	12.2	18.1	41.6	19.9	10.0	10.3

Compared to a full-attention decoder, CEPE's performance is more stable across different tasks

Conclusion

Conclusion

- CEPE is a lightweight framework for extending the context window of any decoder-only LMs

Conclusion

- CEPE is a lightweight framework for extending the context window of any decoder-only LMs
- CEPE is **length-generalizable**, **efficient**, and **easy to train**

Conclusion

- CEPE is a lightweight framework for extending the context window of any decoder-only LMs
- CEPE is **length-generalizable**, **efficient**, and **easy to train**
- CEPE performs well on both **long-context modeling** and **retrieval-augmented** applications

Conclusion

- CEPE is a lightweight framework for extending the context window of any decoder-only LMs
- CEPE is **length-generalizable**, **efficient**, and **easy to train**
- CEPE performs well on both **long-context modeling** and **retrieval-augmented** applications
- CEPE works well with instruction-tuned/chat models too (with only unlabeled data)

What’s next

What’s next

What’s next

- Long-context instruction-tuning

What’s next

- Long-context instruction-tuning
- Reduce training cost

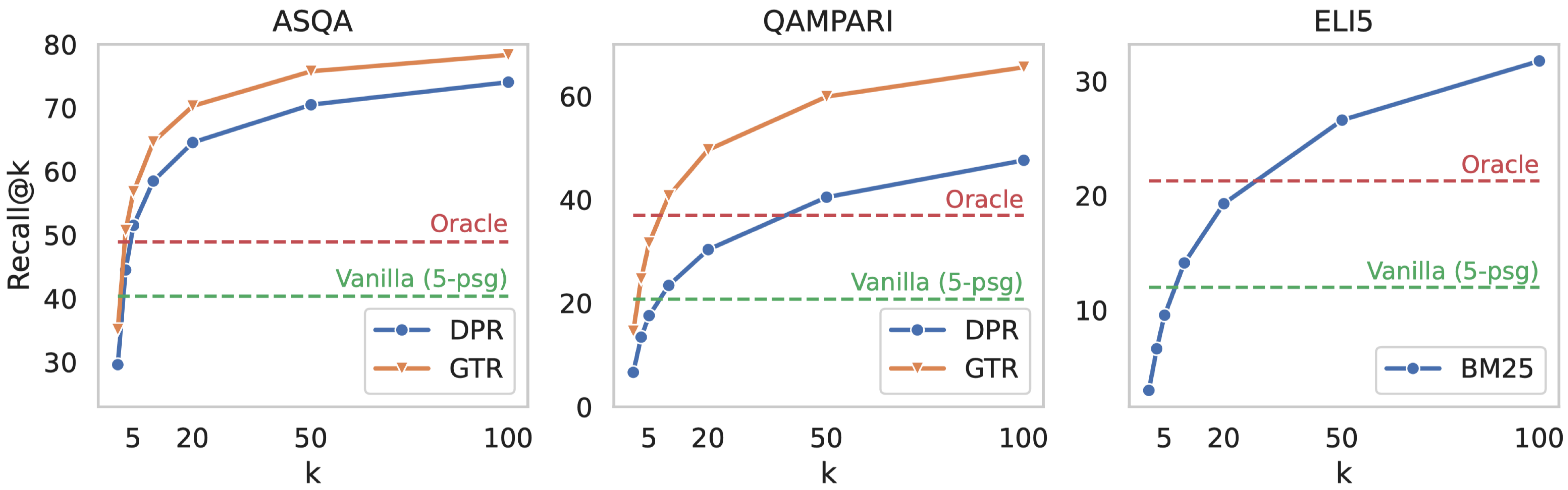
What's next

- Long-context instruction-tuning
- Reduce training cost
- Reduce inference cost

What's next

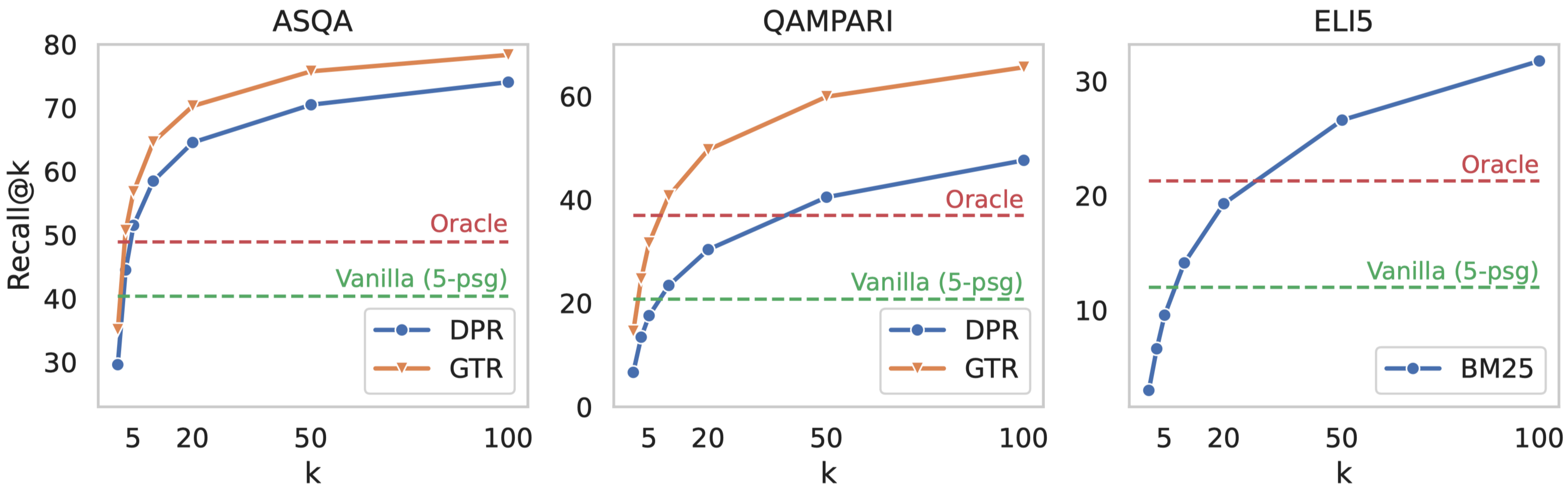
- Long-context instruction-tuning
- Reduce training cost
- Reduce inference cost
- **Improve retrieval-augmented applications**

… Back to ALCE



More passages encode more information to answer the question.

… Back to ALCE



More passages encode more information to answer the question.

Can LLMs use them effectively?

Gao et al., 2023. Enabling Large Language Models to Generate Text with Citations

… Back to ALCE

	Fluency	Correct.	Citation	
	(MAUVE)	(EM Rec.)	Rec.	Prec.
ChatGPT-16K (max #tokens=16K)				
ChatGPT (5-psg)	60.3	36.1	76.2	76.5
ChatGPT (10-psg)	56.3	36.7	75.3	75.0
ChatGPT (20-psg)	56.7	36.1	73.7	73.5
GPT-4 (max #tokens=8K)				
GPT-4 (5-psg)	67.1	41.3	68.5	75.6
GPT-4 (10-psg)	71.5	43.1	72.0	75.5
GPT-4 (20-psg)	64.9	44.4	73.0	76.5

Gao et al., 2023. Enabling Large Language Models to Generate Text with Citations. Results are reported on the ASQA subset

… Back to ALCE

	Fluency (MAUVE)	Correct. (EM Rec.)	Citation Rec.	Citation Prec.
ChatGPT-16K (max #tokens=16K)				
ChatGPT (5-psg)	60.3	36.1	76.2	76.5
ChatGPT (10-psg)	56.3	36.7	75.3	75.0
ChatGPT (20-psg)	56.7	36.1	73.7	73.5
GPT-4 (max #tokens=8K)				
GPT-4 (5-psg)	67.1	41.3	68.5	75.6
GPT-4 (10-psg)	71.5	43.1	72.0	75.5
GPT-4 (20-psg)	64.9	44.4	73.0	76.5

- More passages do not lead to better performance

Gao et al., 2023. Enabling Large Language Models to Generate Text with Citations. Results are reported on the ASQA subset

… Back to ALCE

	Fluency	Correct.	Citation	
	(MAUVE)	(EM Rec.)	Rec.	Prec.
ChatGPT-16K (max #tokens=16K)				
ChatGPT (5-psg)	60.3	36.1	76.2	76.5
ChatGPT (10-psg)	56.3	36.7	75.3	75.0
ChatGPT (20-psg)	56.7	36.1	73.7	73.5
GPT-4 (max #tokens=8K)				
GPT-4 (5-psg)	67.1	41.3	68.5	75.6
GPT-4 (10-psg)	71.5	43.1	72.0	75.5
GPT-4 (20-psg)	64.9	44.4	73.0	76.5

- More passages do not lead to better performance
- A stronger model utilizes the more information better

Gao et al., 2023. Enabling Large Language Models to Generate Text with Citations. Results are reported on the ASQA subset

… Back to ALCE

	Fluency	Correct.	Citation	
	(MAUVE)	(EM Rec.)	Rec.	Prec.
ChatGPT-16K (max #tokens=16K)				
ChatGPT (5-psg)	60.3	36.1	76.2	76.5
ChatGPT (10-psg)	56.3	36.7	75.3	75.0
ChatGPT (20-psg)	56.7	36.1	73.7	73.5
GPT-4 (max #tokens=8K)				
GPT-4 (5-psg)	67.1	41.3	68.5	75.6
GPT-4 (10-psg)	71.5	43.1	72.0	75.5
GPT-4 (20-psg)	64.9	44.4	73.0	76.5
Oracle (20-psg)		>70		

- More passages do not lead to better performance
- A stronger model utilizes the more information better

Gao et al., 2023. Enabling Large Language Models to Generate Text with Citations. Results are reported on the ASQA subset

… Back to ALCE

	Fluency	Correct.	Citation	
	(MAUVE)	(EM Rec.)	Rec.	Prec.
ChatGPT-16K (max #tokens=16K)				
ChatGPT (5-psg)	60.3	36.1	76.2	76.5
ChatGPT (10-psg)	56.3	36.7	75.3	75.0
ChatGPT (20-psg)	56.7	36.1	73.7	73.5
GPT-4 (max #tokens=8K)				
GPT-4 (5-psg)	67.1	41.3	68.5	75.6
GPT-4 (10-psg)	71.5	43.1	72.0	75.5
GPT-4 (20-psg)	64.9	44.4	73.0	76.5
Oracle (20-psg)		>70		

- More passages do not lead to better performance
- A stronger model utilizes the more information better
- LLMs are not good at synthesizing information from long context

Gao et al., 2023. Enabling Large Language Models to Generate Text with Citations. Results are reported on the ASQA subset

… Back to ALCE

	Fluency	Correct.	Citation	
	(MAUVE)	(EM Rec.)	Rec.	Prec.
GPT-4 (10-psg)	71.5	43.1	72.0	75.5
GPT-4 (20-psg)	64.9	44.4	73.0	76.5
Oracle (20-psg)		>70		

- More passages do not lead to better performance

tter

from long context

Improve LLMs' ability to **retrieve and synthesize** multiple pieces of information from long-context

Gao et al., 2023. Enabling Large Language Models to Generate Text with Citations. Results are reported on the ASQA subset

Contact: tianyug@princeton.edu

@gaotianyu1350 @princeton_nlp @PrincetonPLI