# Section 5: Applications

# Downstream adaptation of retrieval-based LMs

The capital city of Ontario is __

Datastore

LM

**Toronto, which is known for …**

# Downstream adaptation of retrieval-based LMs

What are the **tasks**?

**Open-domain QA**

What is the capital of Ontario?

Datastore

LM

**Toronto**

# Downstream adaptation of retrieval-based LMs

What are the **tasks**?

**Fact verification**

Ottawa is the Ontario state capital.

Datastore

LM

**False**

# A range of target tasks

**Question Answering**

RETRO (Borgeaud et al., 2021)

REALM (Gu et al, 2020)

ATLAS (Izacard et al, 2023)

**Fact verification**

RAG (Lewis et al, 2020)

ATLAS (Izacard et al, 2022)

Evi. Generator (Asai et al, 2022)

**Dialogue**

BlenderBot3 (Shuster et al., 2022)

Internet-augmented generation (Komeili et a., 2022)

Retrieval-based LMs have been extensively evaluated on knowledge-intensive tasks

# A range of target tasks

### Question answering
RETRO (Borgeaud et al., 2021)
REALM (Gu et al, 2020)
ATLAS (Izacard et al, 2023)

### Fact verification
RAG (Lewis et al, 2020)
ATLAS (Izacard et al, 2022)
Evi. Generator (Asai et al, 2022)

### Dialogue
BlenderBot3 (Shuster et al., 2022)
Internet-augmented generation (Komeili et a., 2022)

### Summarization
FLARE (Jiang et al, 2023)

### Machine translation
kNN-MT (Khandelwal et al., 2020)
TRIME-MT (Zhong et al., 2022)

### Code & proof generation
DocPrompting (Zhou et al., 2023)
Natural Prover (Welleck et al., 2022)

### NLI
kNN-Prompt (Shi et al., 2022)
NPM (Min et al., 2023)

### Sentiment analysis
kNN-Prompt (Shi et al., 2022)
NPM (Min et al., 2023)

### Commonsense reasoning
Raco (Yu et al, 2022)

More general NLP tasks

# A range of target tasks

### Question answering
RETRO (Borgeaud et al., 2021)
REALM (Gu et al, 2020)
ATLAS (Izacard et al, 2023)

### Fact verification
RAG (Lewis et al, 2020)
ATLAS (Izacard et al, 2022)
Evi. Generator (Asai et al, 2022)

### Dialogue
BlenderBot3 (Shuster et al., 2022)
Internet-augmented generation (Komeili et a., 2022)

### Summarization
FLARE (Jiang et al, 2023)

### Machine translation
kNN-MT (Khandelwal et al., 2020)
TRIME-MT (Zhong et al., 2022)

### Code & proof generation
DocPrompting (Zhou et al., 2023)
Natural Prover (Welleck et al., 2022)

### NLI
kNN-Prompt (Shi et al., 2022)
NPM (Min et al., 2023)

### Sentiment analysis
kNN-Prompt (Shi et al., 2022)
NPM (Min et al., 2023)

### Commonsense reasoning
Raco (Yu et al, 2022)

More generations

# A range of target tasks

### Question answering
RETRO (Borgeaud et al., 2021)
REALM (Gu et al, 2020)
ATLAS (Izacard et al, 2023)

### Fact verification
RAG (Lewis et al, 2020)
ATLAS (Izacard et al, 2022)
Evi. Generator (Asai et al, 2022)

### Dialogue
BlenderBot3 (Shuster et al., 2022)
Internet-augmented generation (Komeili et a., 2022)

### Summarization
FLARE (Jiang et al, 2023)

### Machine translation
kNN-MT (Khandelwal et al., 2020)
TRIME-MT (Zhong et al., 2022)

### Code & proof generation
DocPrompting (Zhou et al., 2023)
Natural Prover (Welleck et al., 2022)

### NLI
kNN-Prompt (Shi et al., 2022)
NPM (Min et al., 2023)

### Sentiment analysis
kNN-Prompt (Shi et al., 2022)
NPM (Min et al., 2023)

### Commonsense reasoning
Raco (Yu et al, 2022)

More classifications

# Two key questions for downstream adaptations

**How** can we adapt a retrieval-based LM for a task?

**When** should we use a retrieval-based LM?

# **How** to adapt a retrieval-based LM for a task

What are the **tasks**?

- Open-domain QA
- Other knowledge-intensive tasks
- Sentiment analysis
- Code generation

…

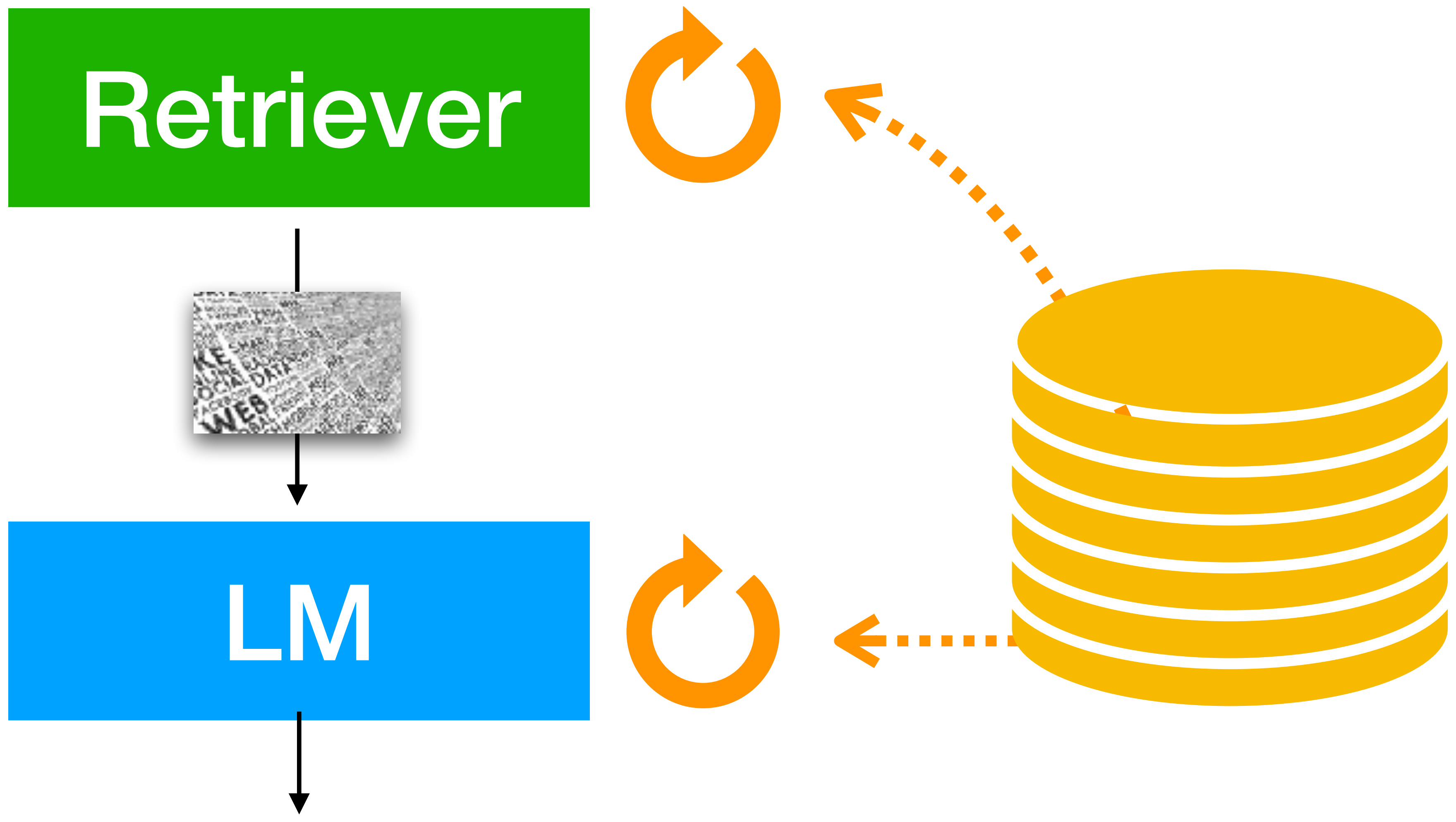
How to **adapt?**

- Fine-tuning
- Reinforcement learning
- Prompting

# How to adapt a retrieval-based LM for a task

**Fine-tuning (+RL)**

Training LM and / or retriever
on task-data & data store

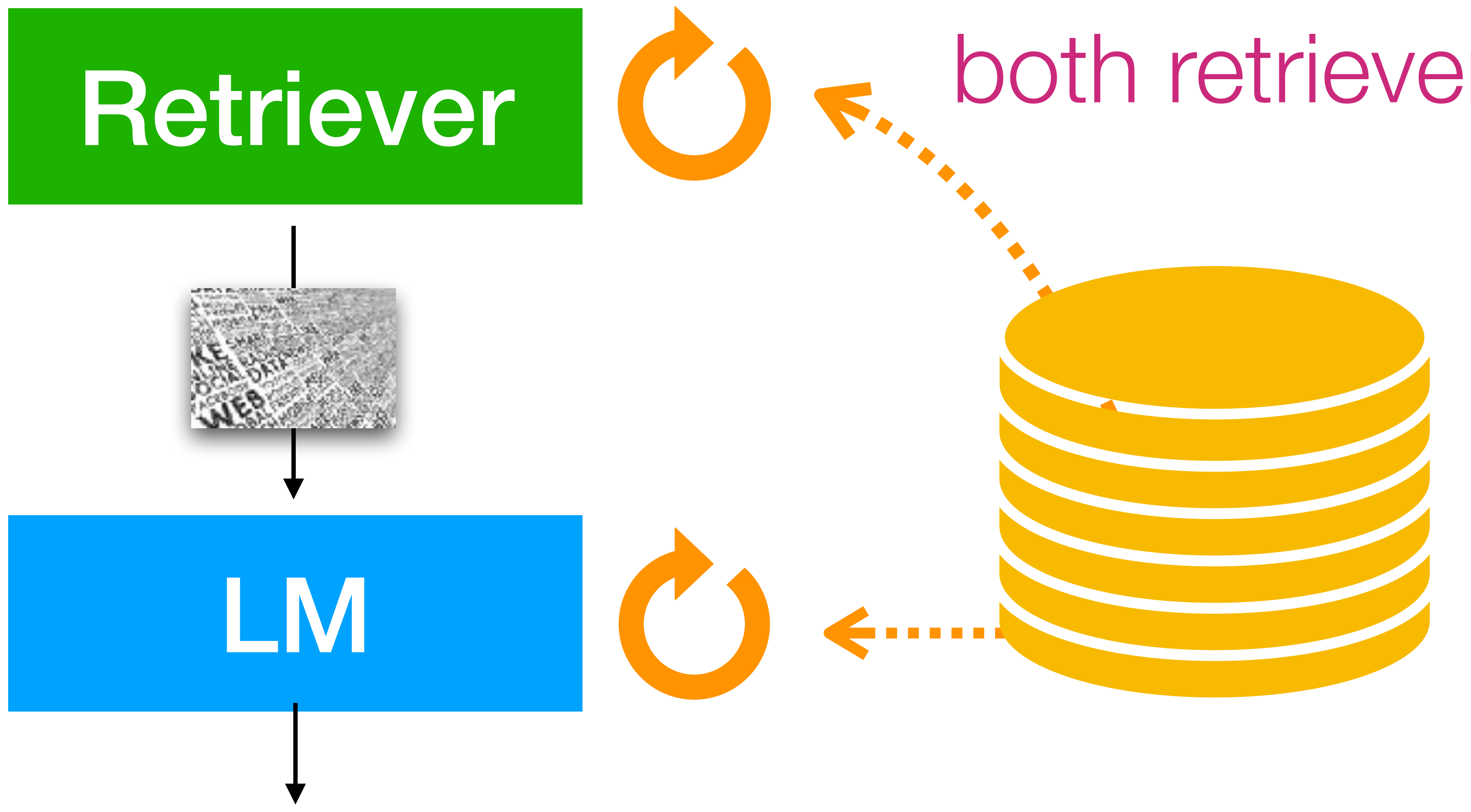


# **How** to adapt a retrieval-based LM for a task

## Fine-tuning (+RL)

Training LM and / or retriever
on task-data & data store

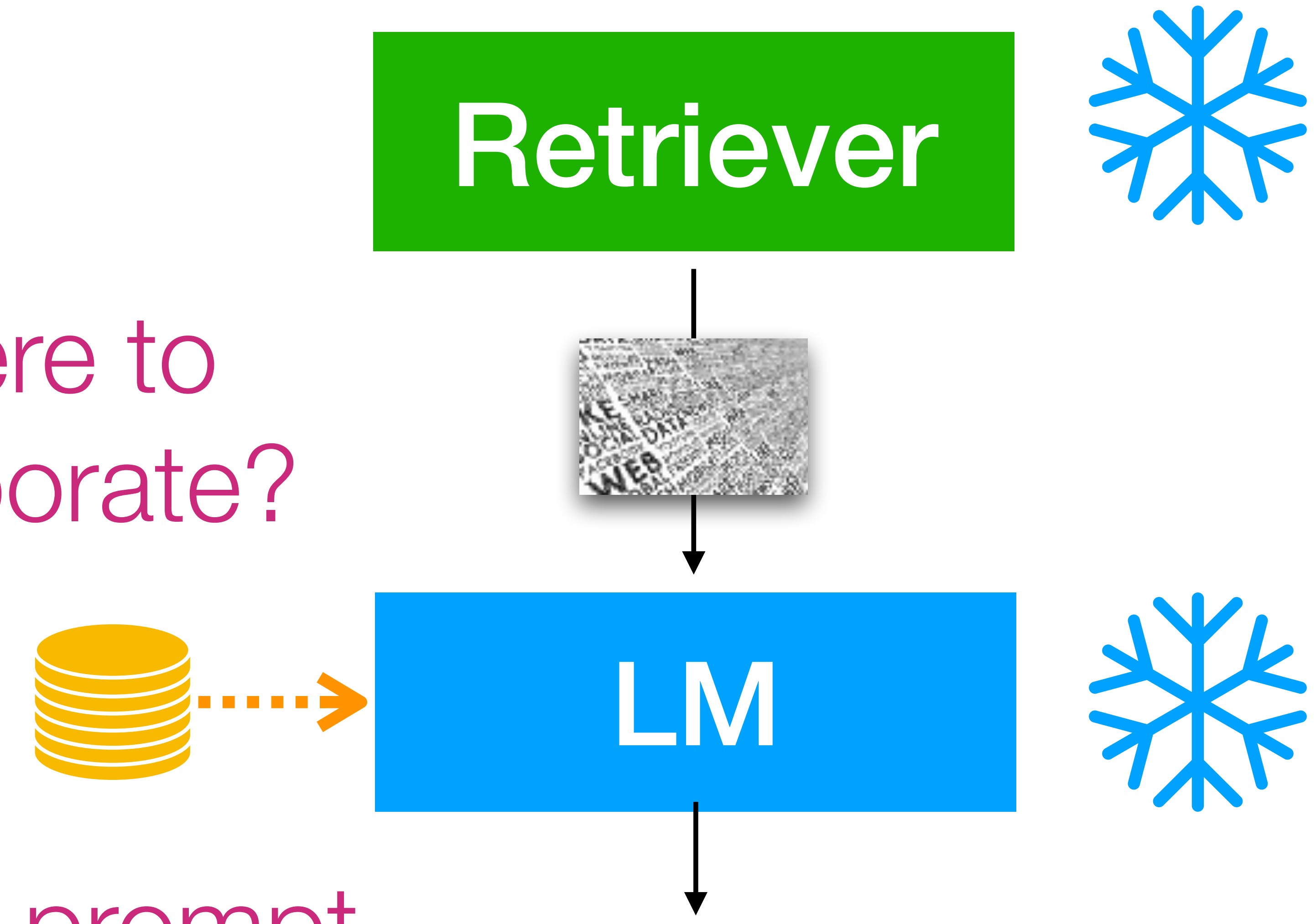
Should we update
both retriever & LM?




## Prompting

Prompt a frozen LM with
retrieved knowledge

Where to
incorporate?




How to prompt
LM?

# How to adapt a retrieval-based LM for a task

What are the **tasks**?

- Open-domain QA
- Other knowledge-intensive tasks
- Sentiment analysis
- Code generation

…

How to **adapt?**

- Fine-tuning
- Reinforcement learning
- Prompting

What is **data store?**

Wikipedia

Training data

Code documentation

# **When** to use a retrieval-based LM

Long-tail

knowledge update

Verifiability

Parameter-efficiency

# Effectiveness of retrieval-based LMs

Long-tail

knowledge update

Verifiability

Parameter-efficiency

**Q:** Is Toronto really cold during winter? → LM → Yes it is.

# Effectiveness of retrieval-based LMs

Long-tail

knowledge update

Verifiability

Parameter-efficiency

**Q:** Where is Toronto Zoo located? → LM → 1361A Old Finch Avenue, in Scarborough, Ontario

# Effectiveness of retrieval-based LMs

Long-tail

knowledge update

Verifiability

Parameter-efficiency

**Q:** Where is Toronto Zoo located?

LM

**361A** Old Finch Avenue, in Scarborough, Ontario

Retriever

Toronto zoo Info
**Location:** 361A Old Finch Avenue, Toronto, Ontario
**Land Area:** 287 hectares

# Effectiveness of retrieval-based LMs

Long-tail

**Knowledge update**

Verifiability

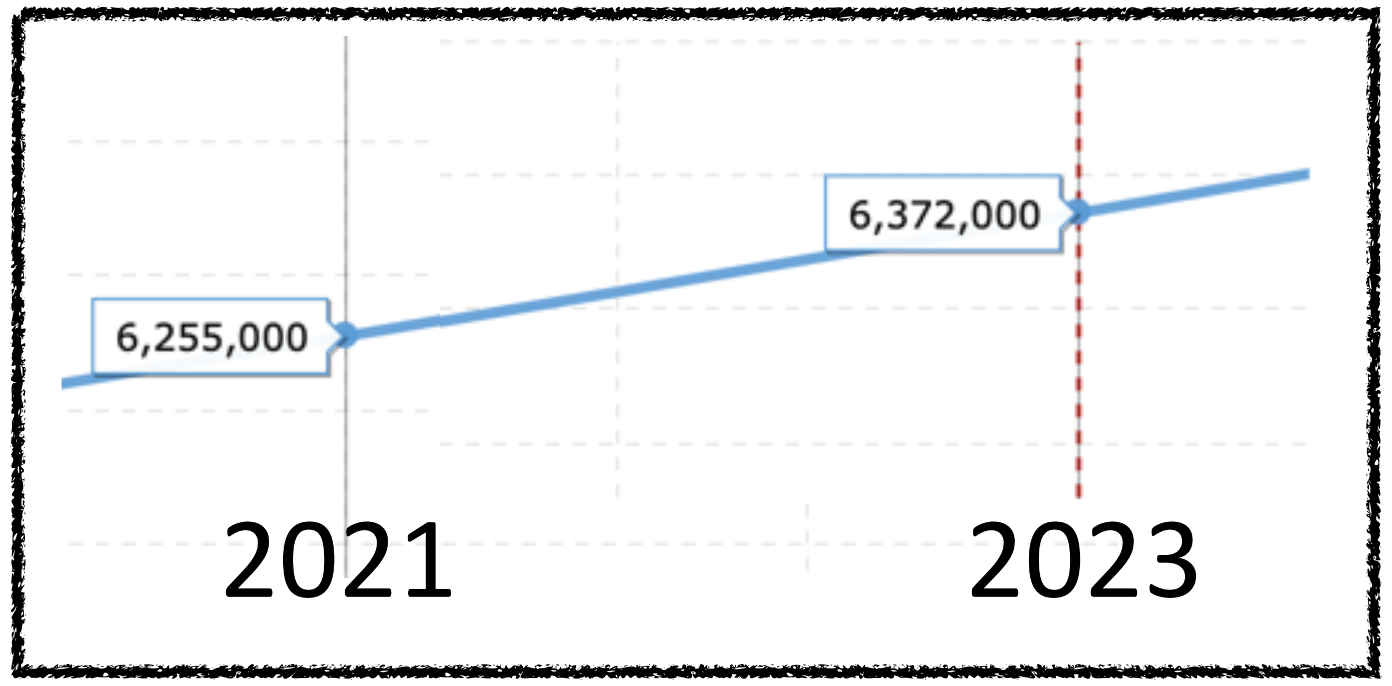
Parameter-efficiency

**Q:** What is the population of Toronto Metropolitan area in 2023?

LM

Trained on the **2021** corpus

**A:** 6,255,000



# Effectiveness of retrieval-based LMs

Long-tail

**Knowledge update**

Verifiability

Parameter-efficiency

**Q:** What is the population of Toronto Metropolitan area in 2023?

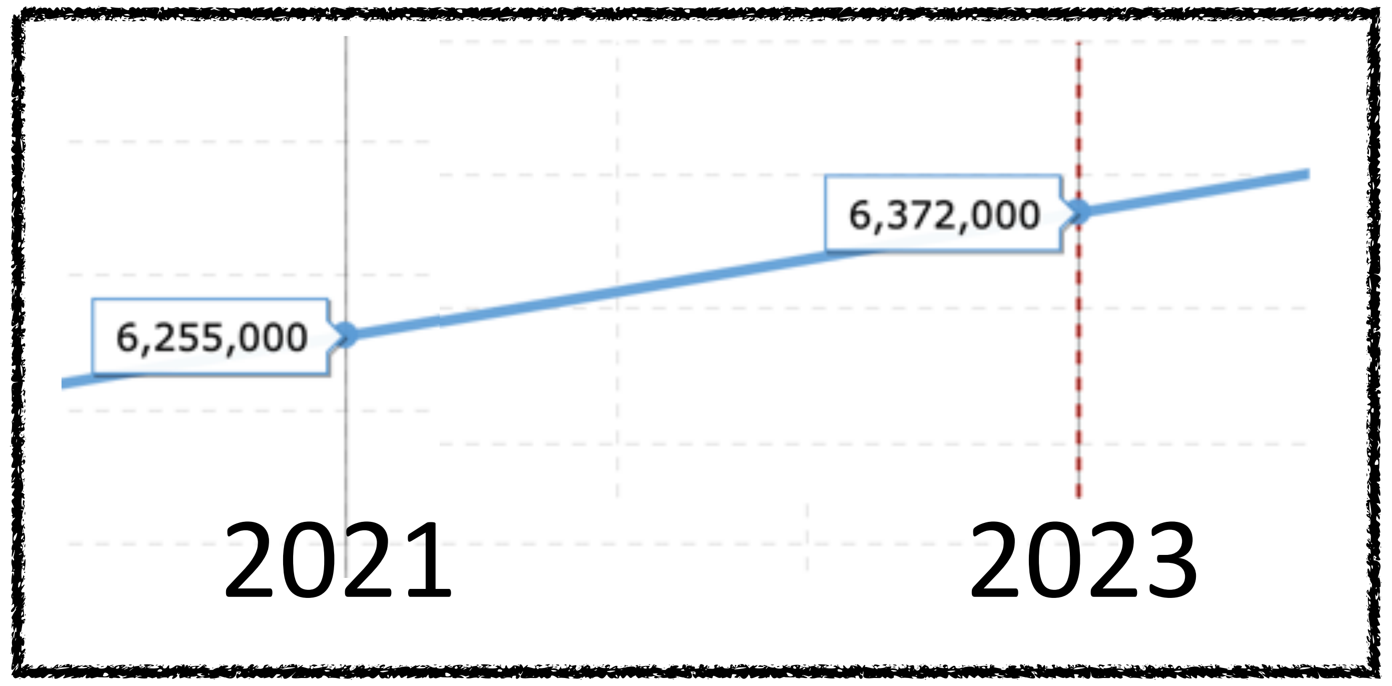
Trained on the **2021** corpus

LM

A: 6,372,000

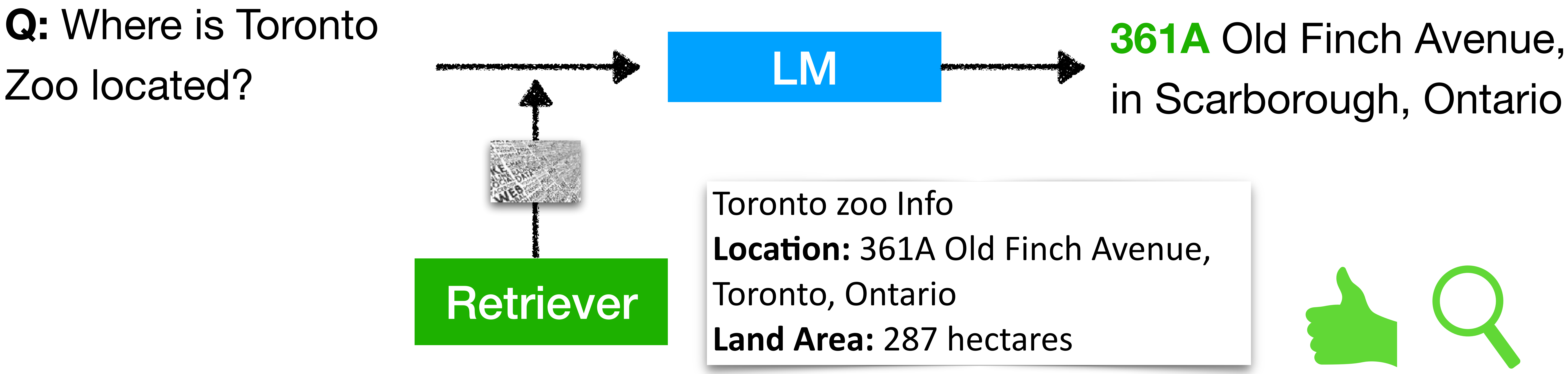
Retriever

Collected in 2023



# Effectiveness of retrieval-based LMs

Long-tail | knowledge update | **Verifiability** | Parameter-efficiency



# Effectiveness of retrieval-based LMs

Long-tail

knowledge update

Verifiability

Parameter-efficiency

# Two key questions for downstream adaptations

**How** can we adapt a retrieval-based LM for a task?

**When** should we use a retrieval-based LM?

# Downstream adaptation of retrieval-based LMs

What are the **tasks**?

- Open-domain QA
- Other knowledge-intensive tasks
- General NLU
- Language Modeling & other generation tasks

How to **adapt?**

- **Fine-tuning**
- Reinforcement learning
- Prompting

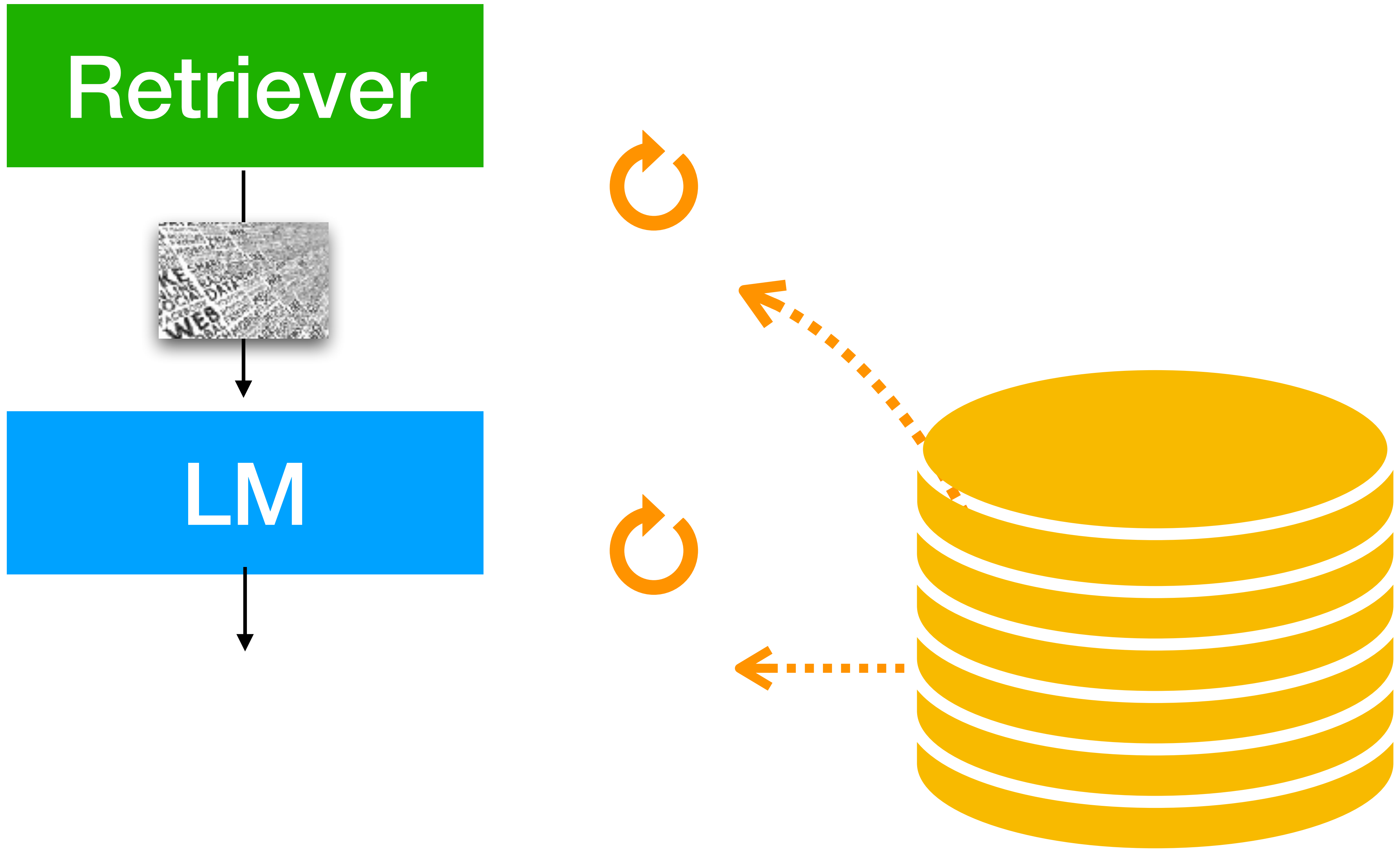
What is **data store?**

- Unlabeled Wikipedia / CC
- Web (Google / Bing Search Results)
- Training data

# Adapting retrieval-based LMs for tasks

**Fine-tuning**

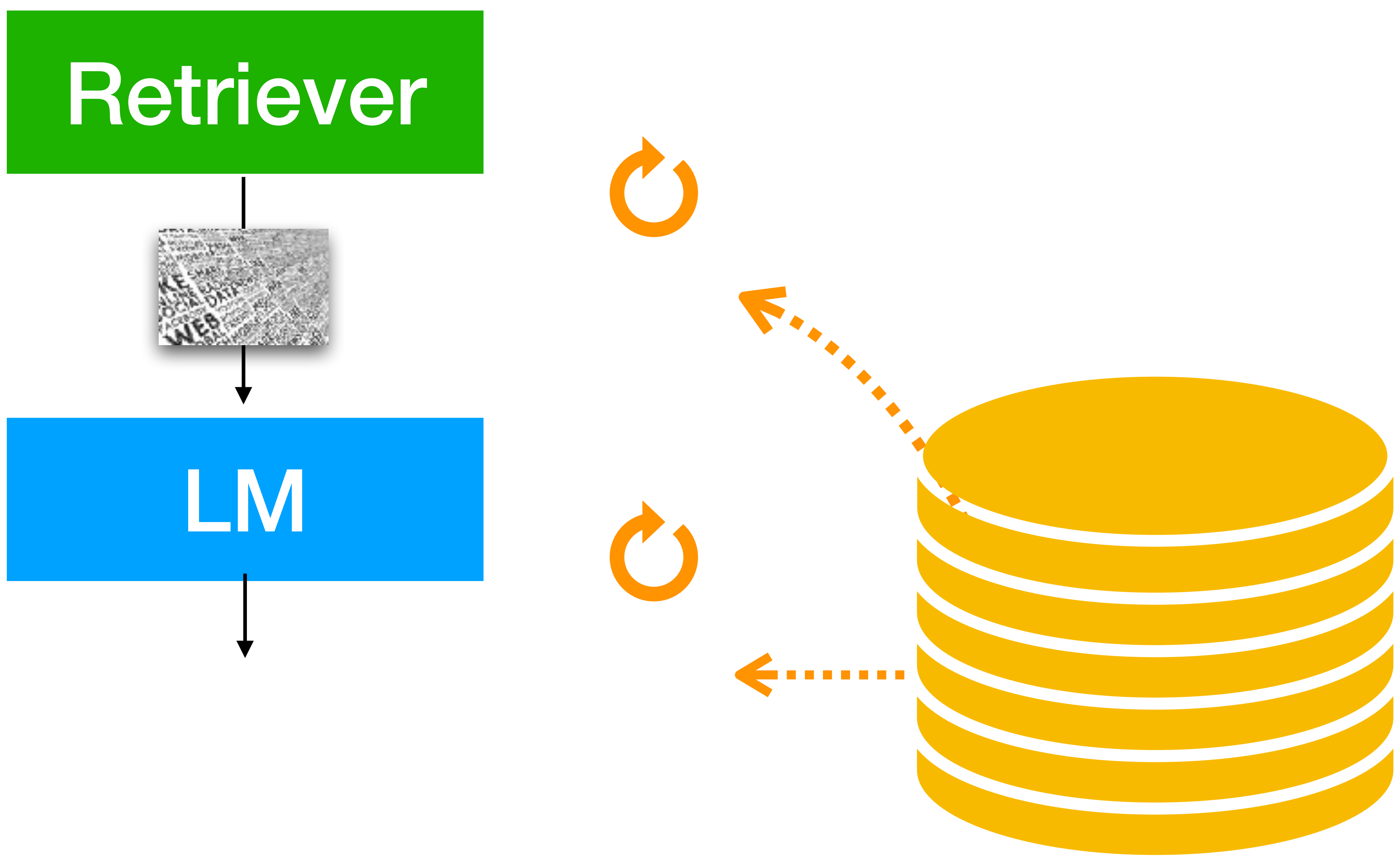
Training LM and / or retriever
on task-data & data store



# Adapting retrieval-based LMs for tasks

**Fine-tuning**

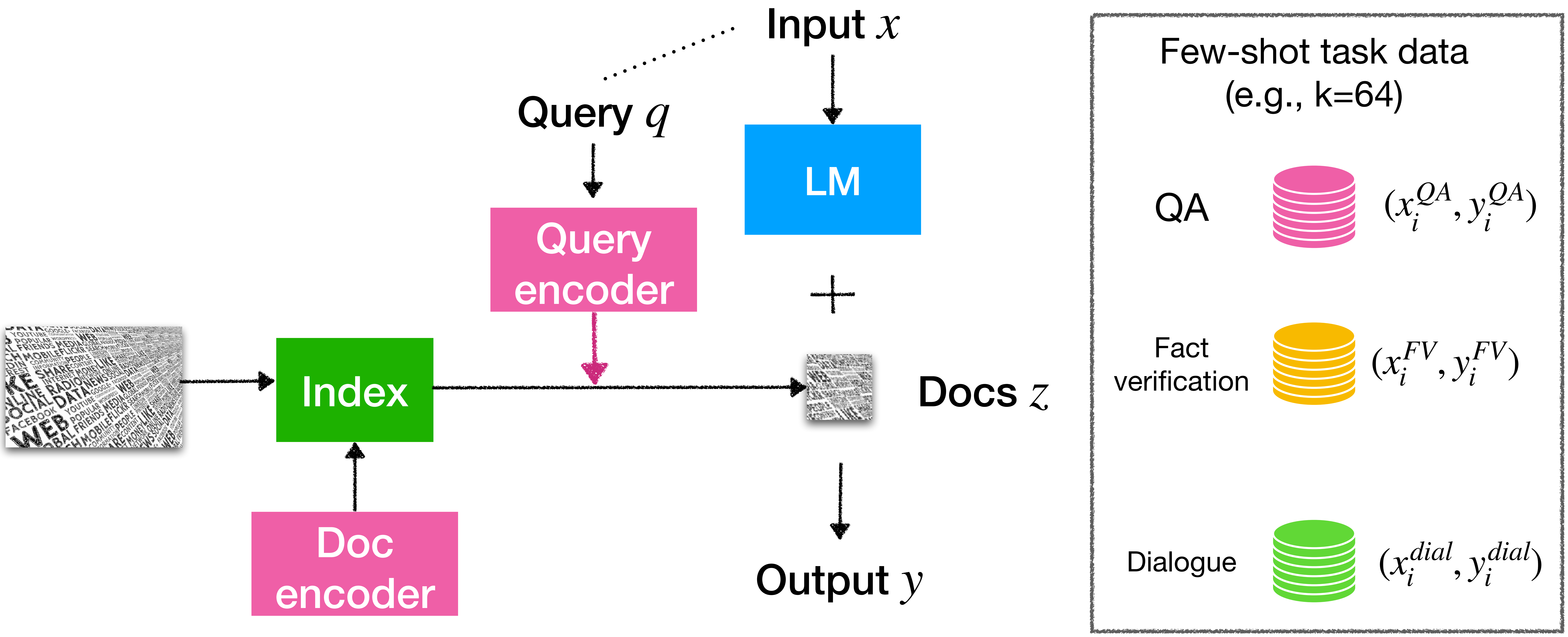
Training LM and / or retriever
on task-data & data store




Costs of retrieval-based LM training (Section 4)

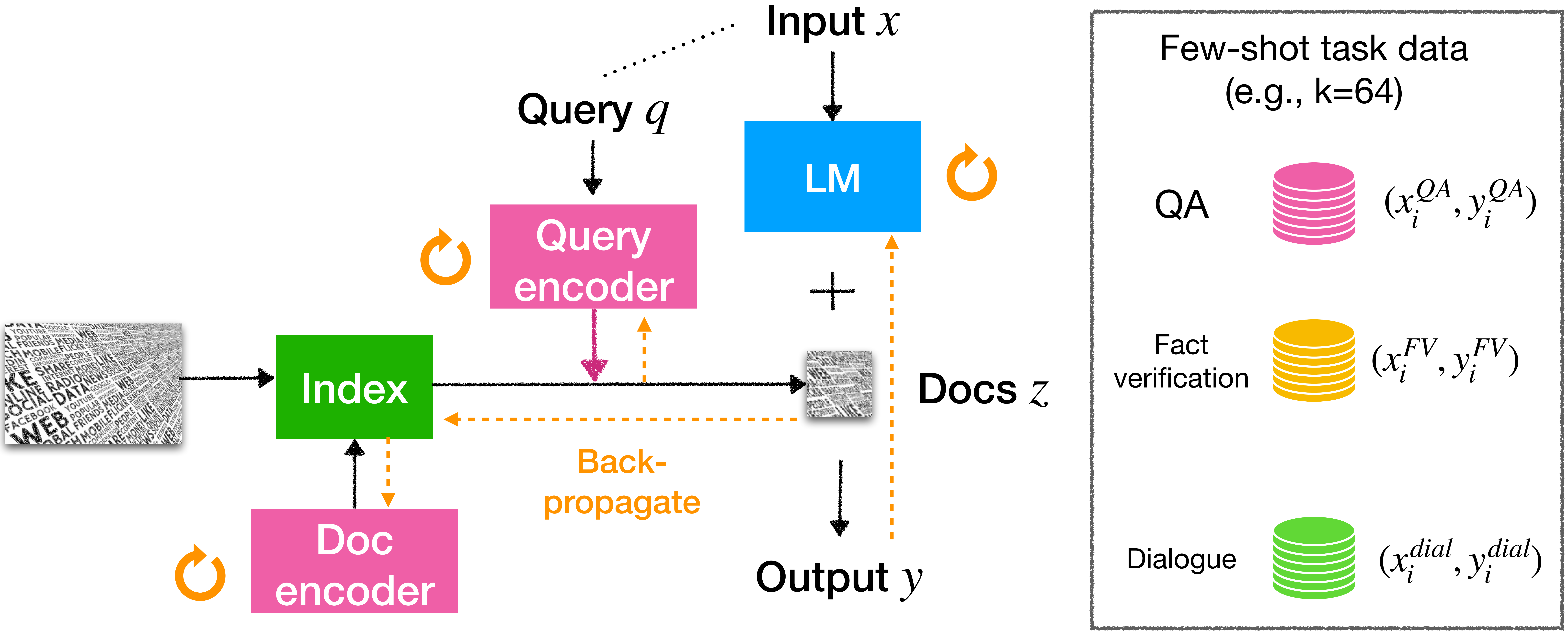
Independent training (DPR)
Asynchronous updates (REALM)
…

# ATLAS (Izacard et al., 2022; Section 4)

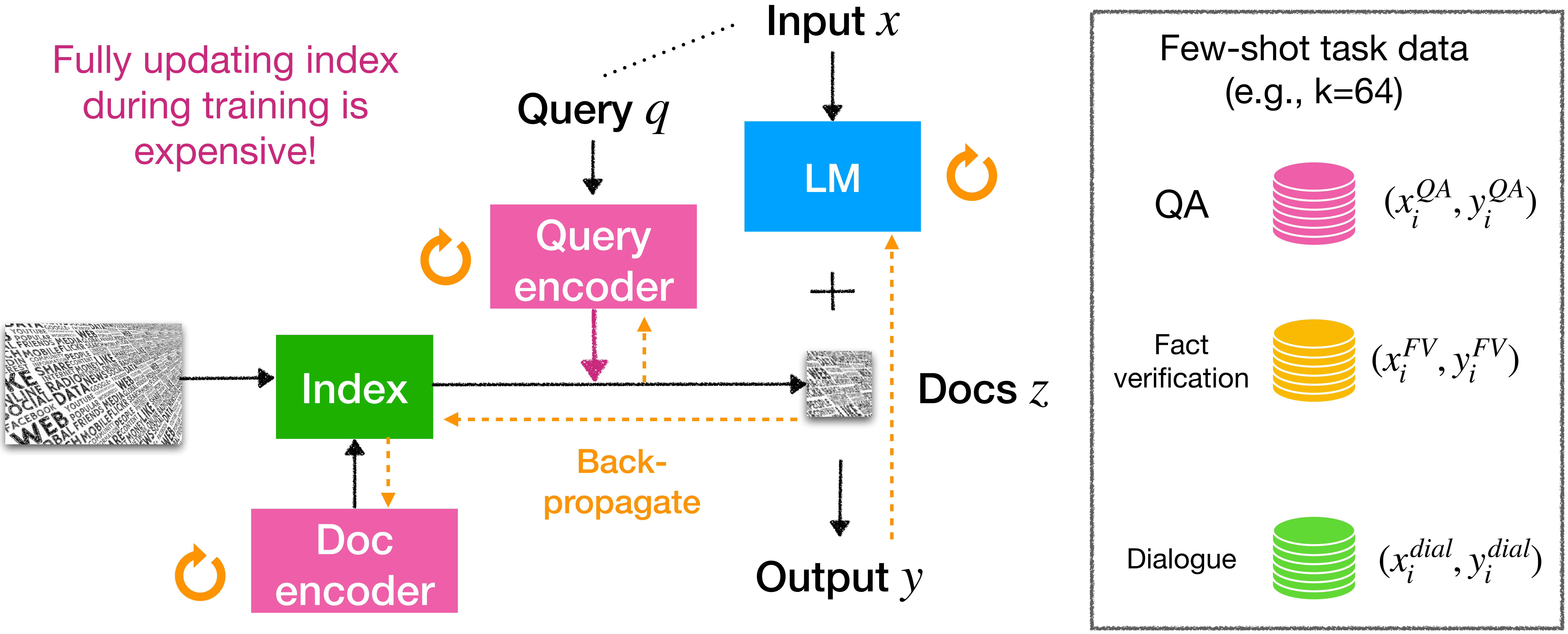



Izacard et al. 2022. "Few-shot learning with retrieval augmented language models"

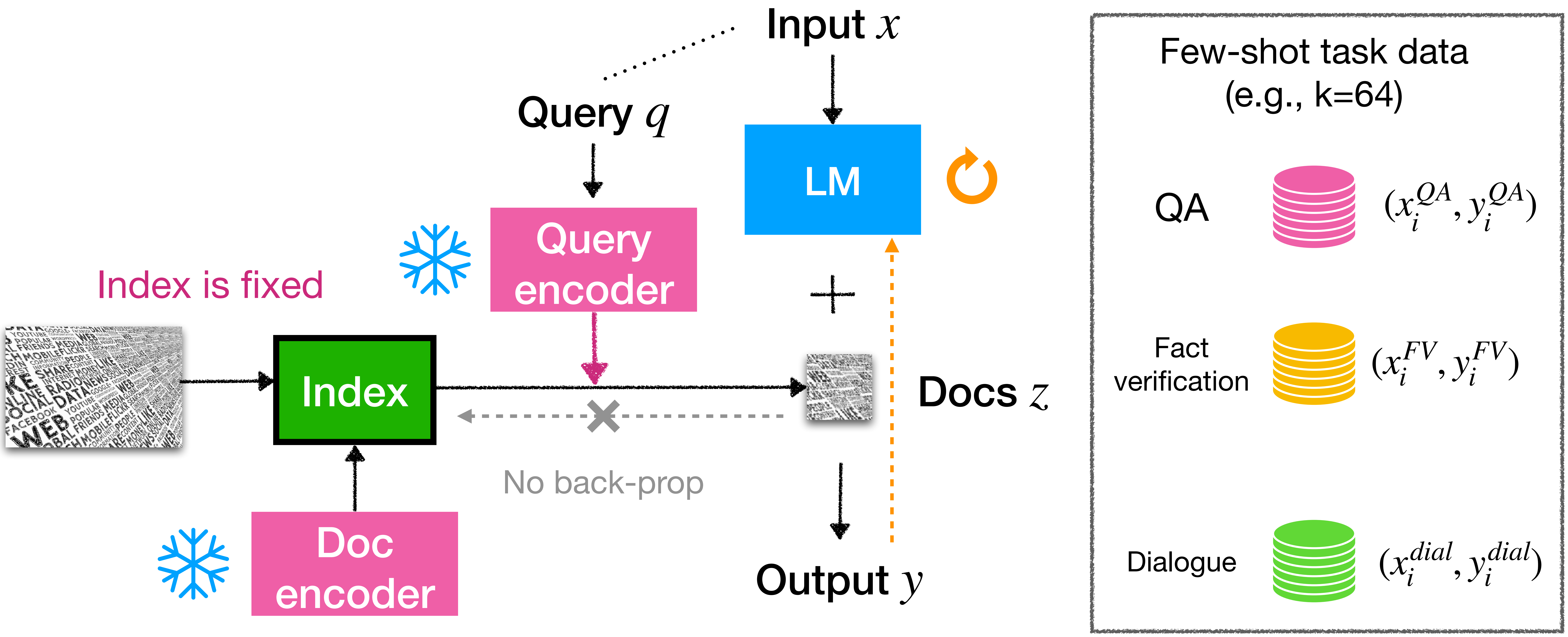
# ATLAS (Izacard et al., 2022; Section 4)



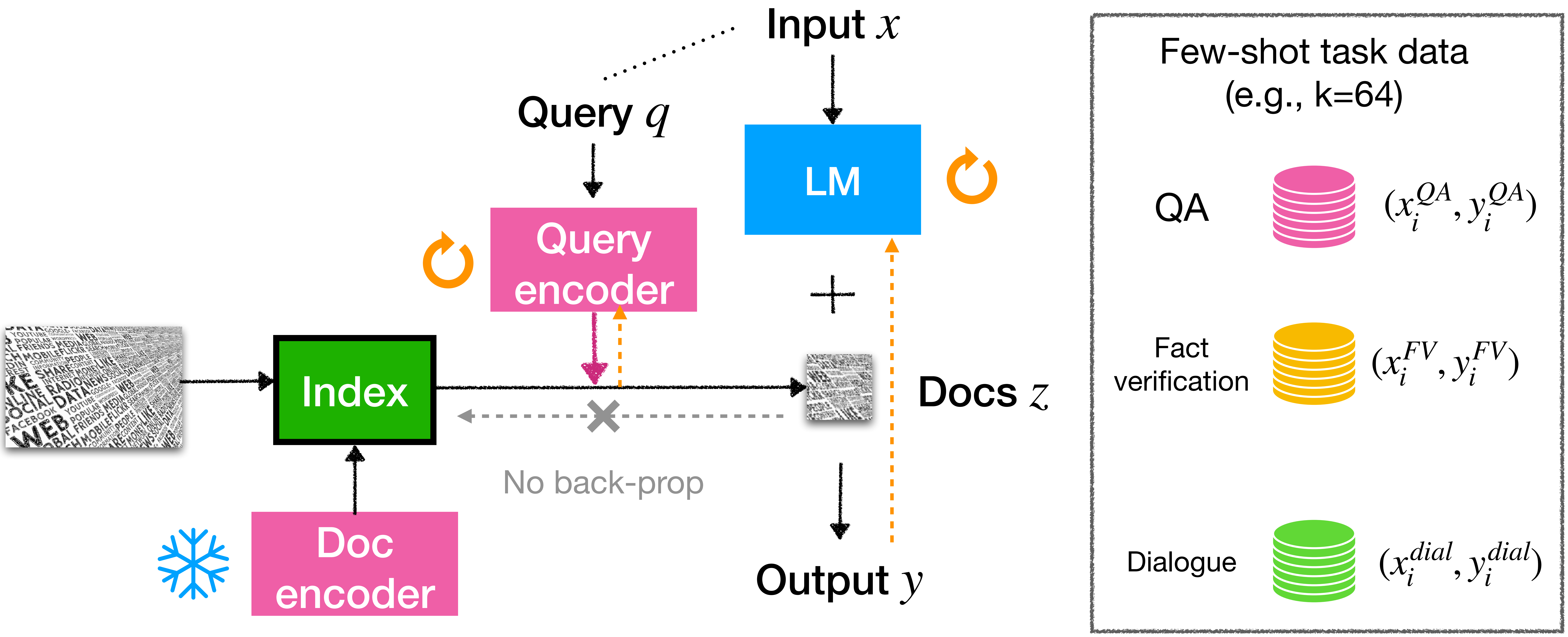
# ATLAS (Izacard et al., 2022; Section 4)



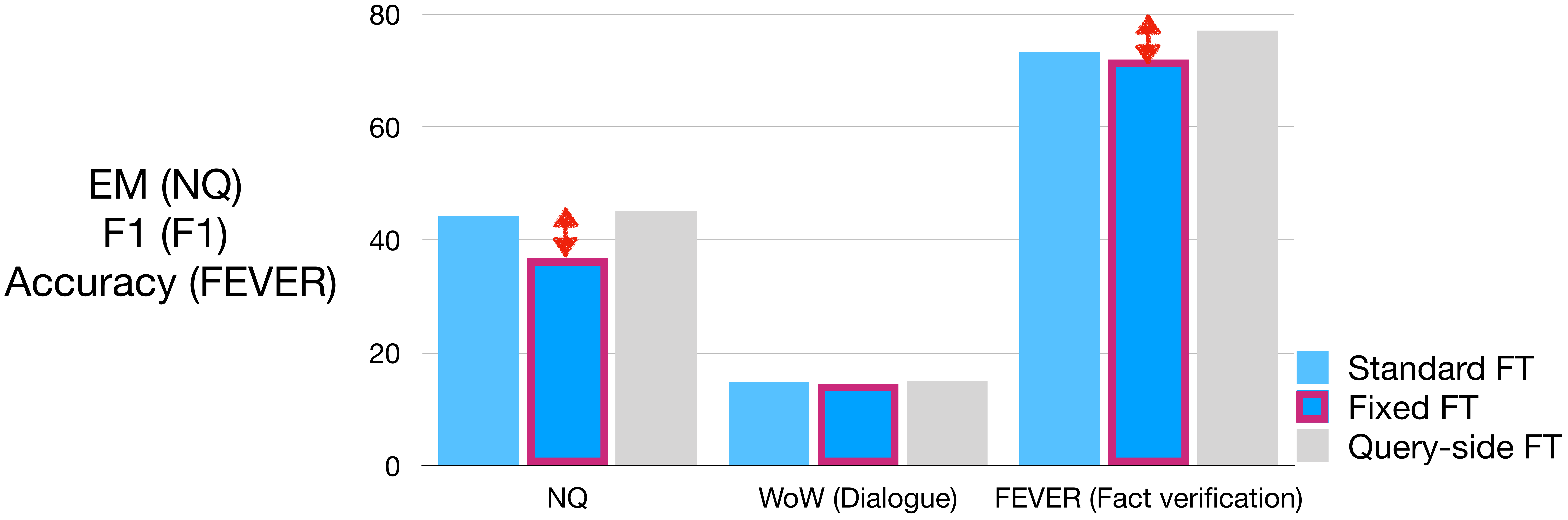
# ATLAS: Fixed retrieval with fine-tuned LM



# ATLAS: Query-side fine-tuning

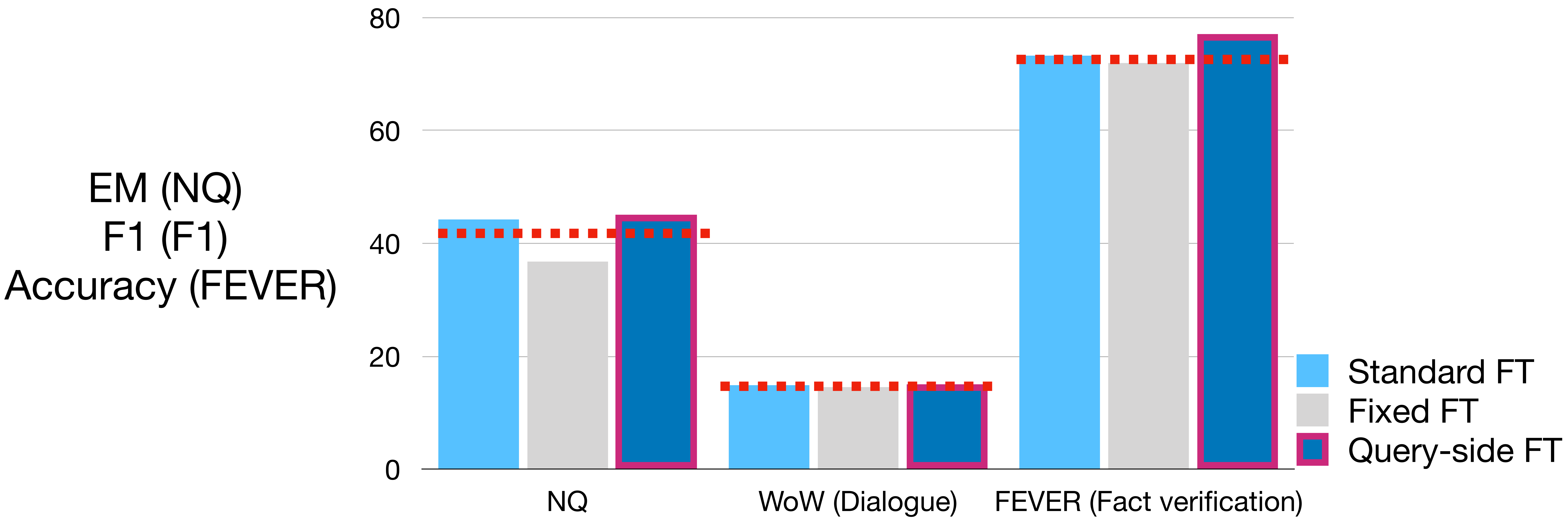


# Ablations of efficient retrieval training




Fixed FT shows large performance drop on QA.

# Ablations of efficient retrieval training

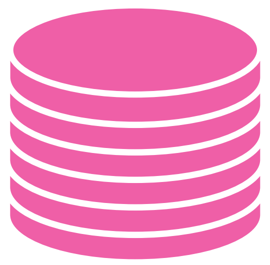



Query-side fine-tuning matches or outperforms full fine-tuning

# ATLAS: Few-shot v.s. full v.s. transfer setups

Few-shot task data
(e.g., $k = 64$)

NQ



$(x_1^{QA}, y_1^{QA})$

$\dots \quad (x_k^{QA}, y_k^{QA})$

MMLU

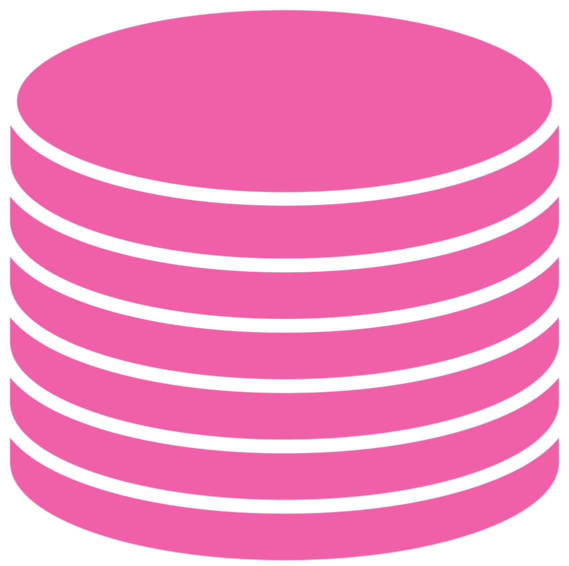
$(x_1^{MMNU}, y_1^{MMNU})$

$\dots$

$(x_k^{MMNU}, y_k^{MMNU})$

Full-shot task data
(e.g., $N = 70{,}000$)

NQ



$(x_1^{QA}, y_1^{QA})$

$\dots$

$(x_N^{QA}, y_N^{QA})$

Transfer task data
(e.g., MCTest —> MMLU)

MCTest

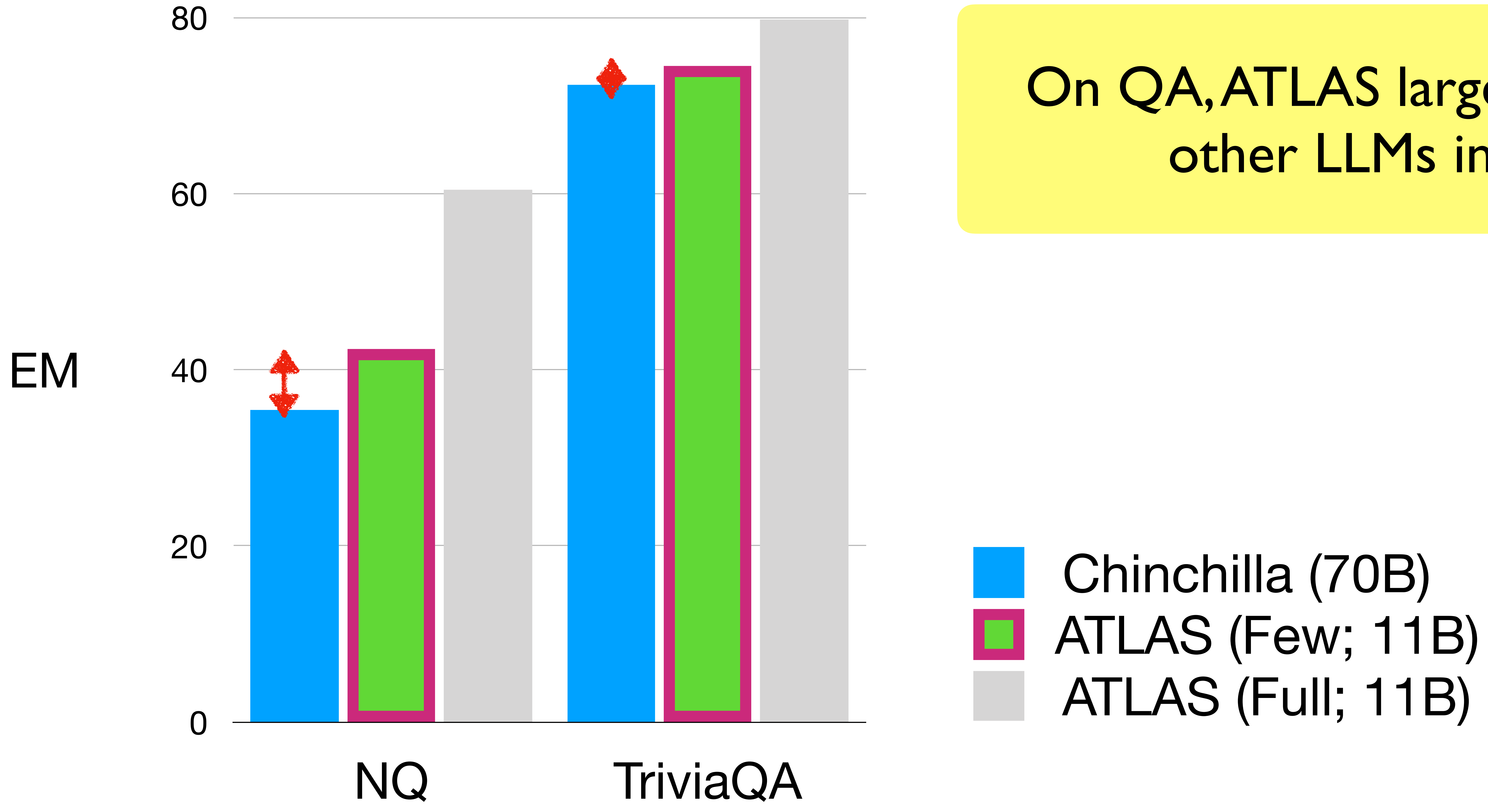
$(x_1^{TF}, y_1^{TF})$

$\dots$

$(x_N^{TF}, y_N^{TF})$

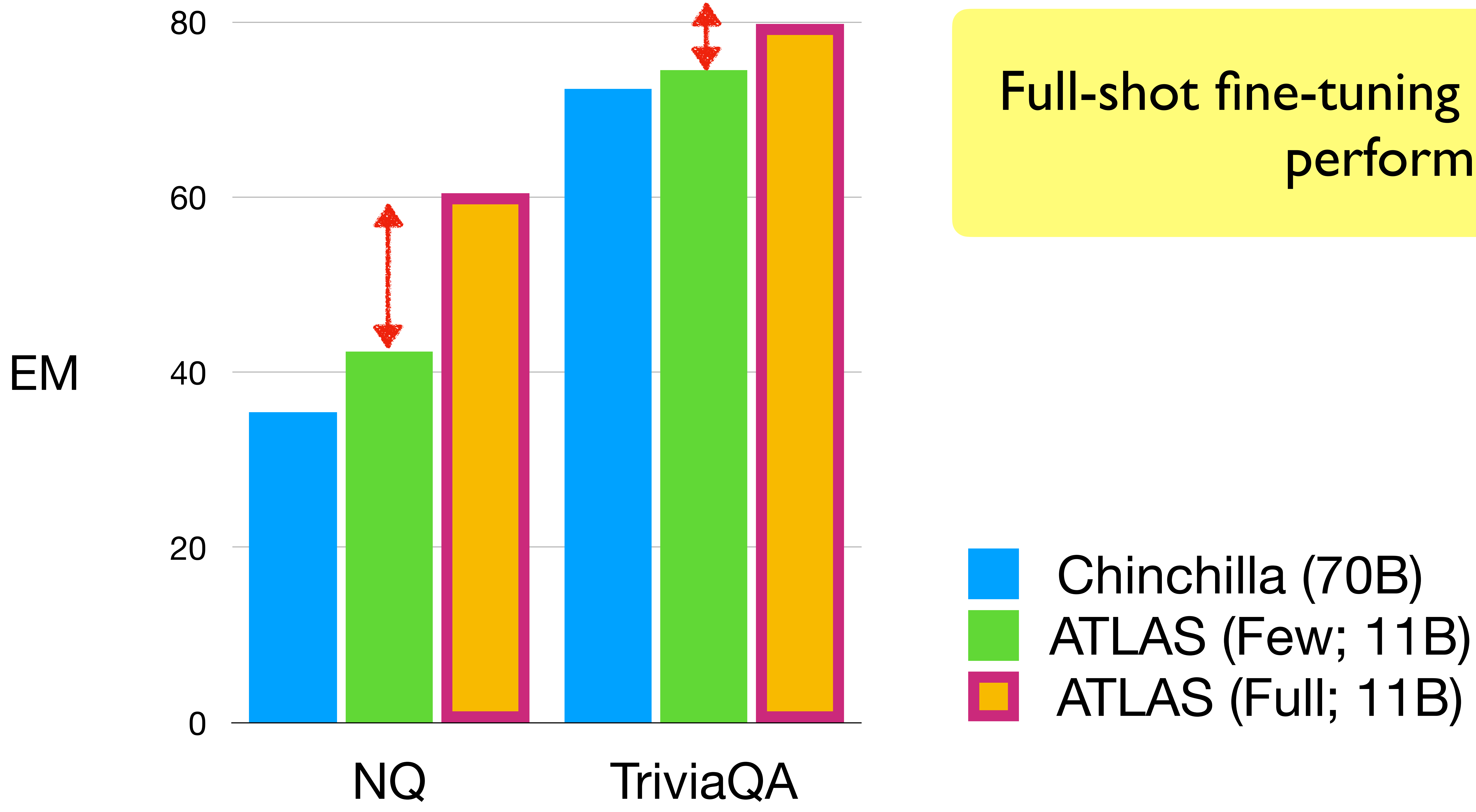
Kwiatkowski et al. 2019. “Natural Questions: A Benchmark for Question Answering Research”
Hendrycks et al. 2021. “Measuring Massive Multitask Language Understanding”

# Task results



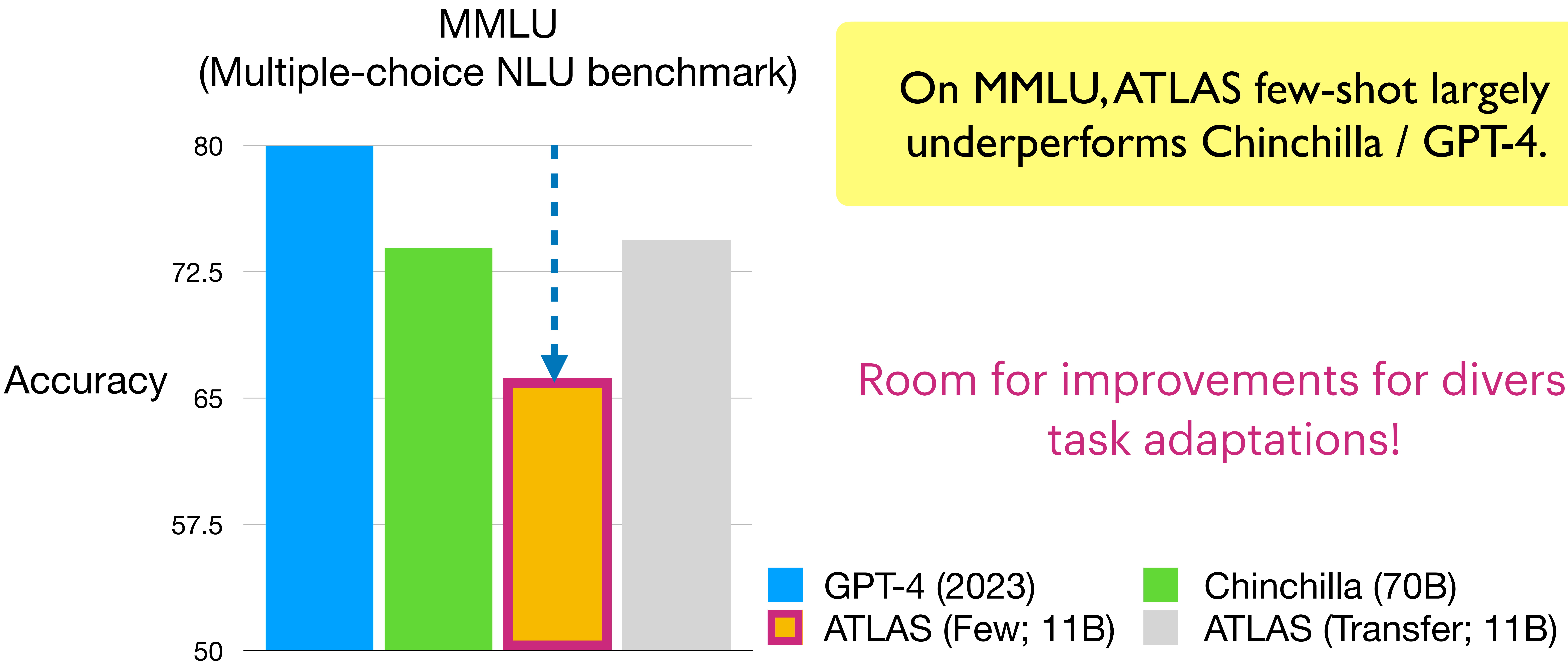

On QA, ATLAS largely outperforms other LLMs in few-shot

# Task results




Full-shot fine-tuning further improves performance

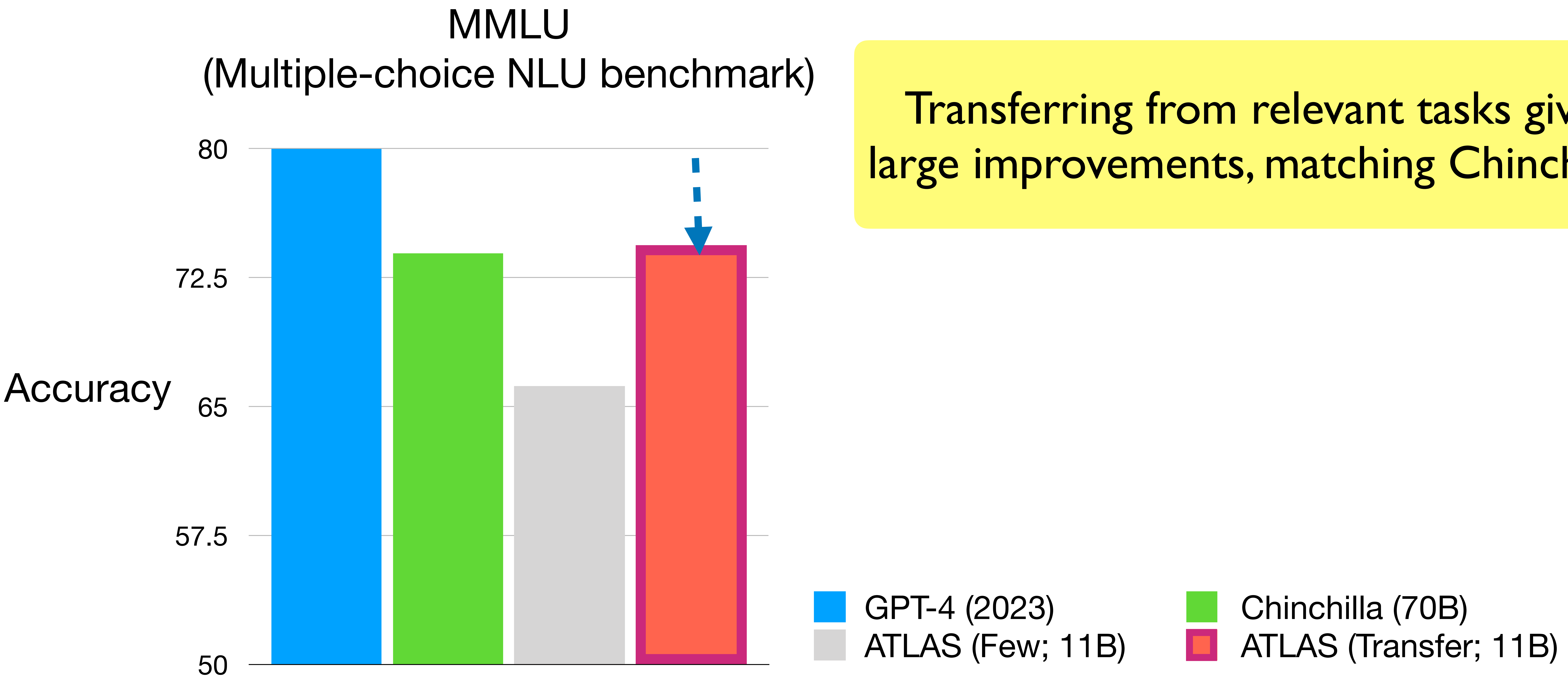
# Task results




On MMLU, ATLAS few-shot largely underperforms Chinchilla / GPT-4.

Room for improvements for diverse task adaptations!

# Task results




Transferring from relevant tasks give large improvements, matching Chinchilla

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |

Fine-tuning for QA & knowledge-intensive tasks often gives strong performance (*even in few-shot*)

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |

Fine-tuning a retriever for a task matters!

# Downstream adaptation of retrieval-based LMs

What are the **tasks**?

- Open-domain QA
- Other knowledge-intensive tasks
- General NLU
- Language Modeling & other generation tasks

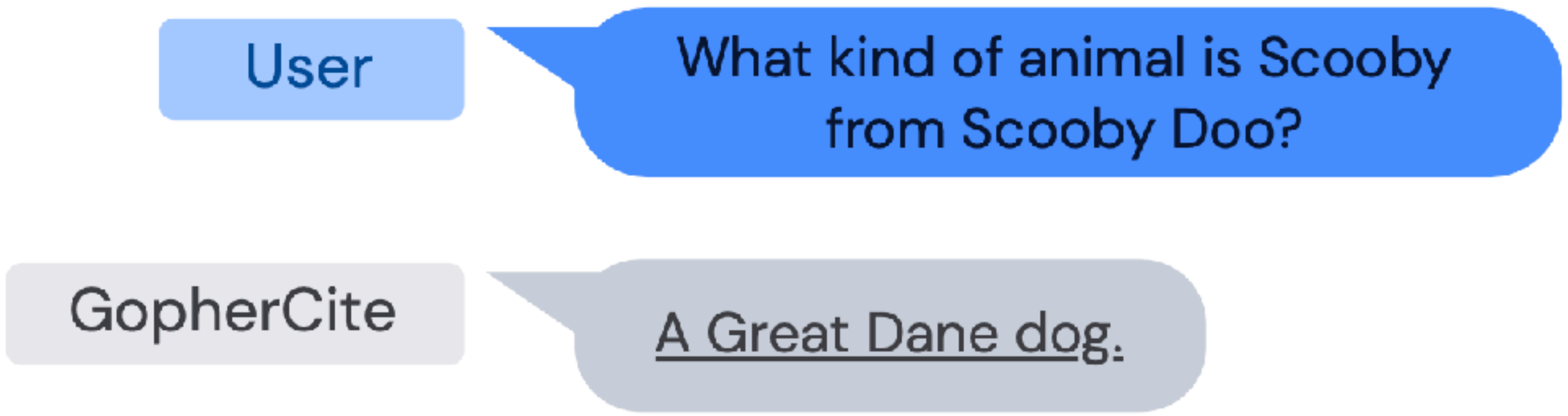
How to **adapt?**

- Fine-tuning
- **Reinforcement learning**
- Prompting

What is **data store?**

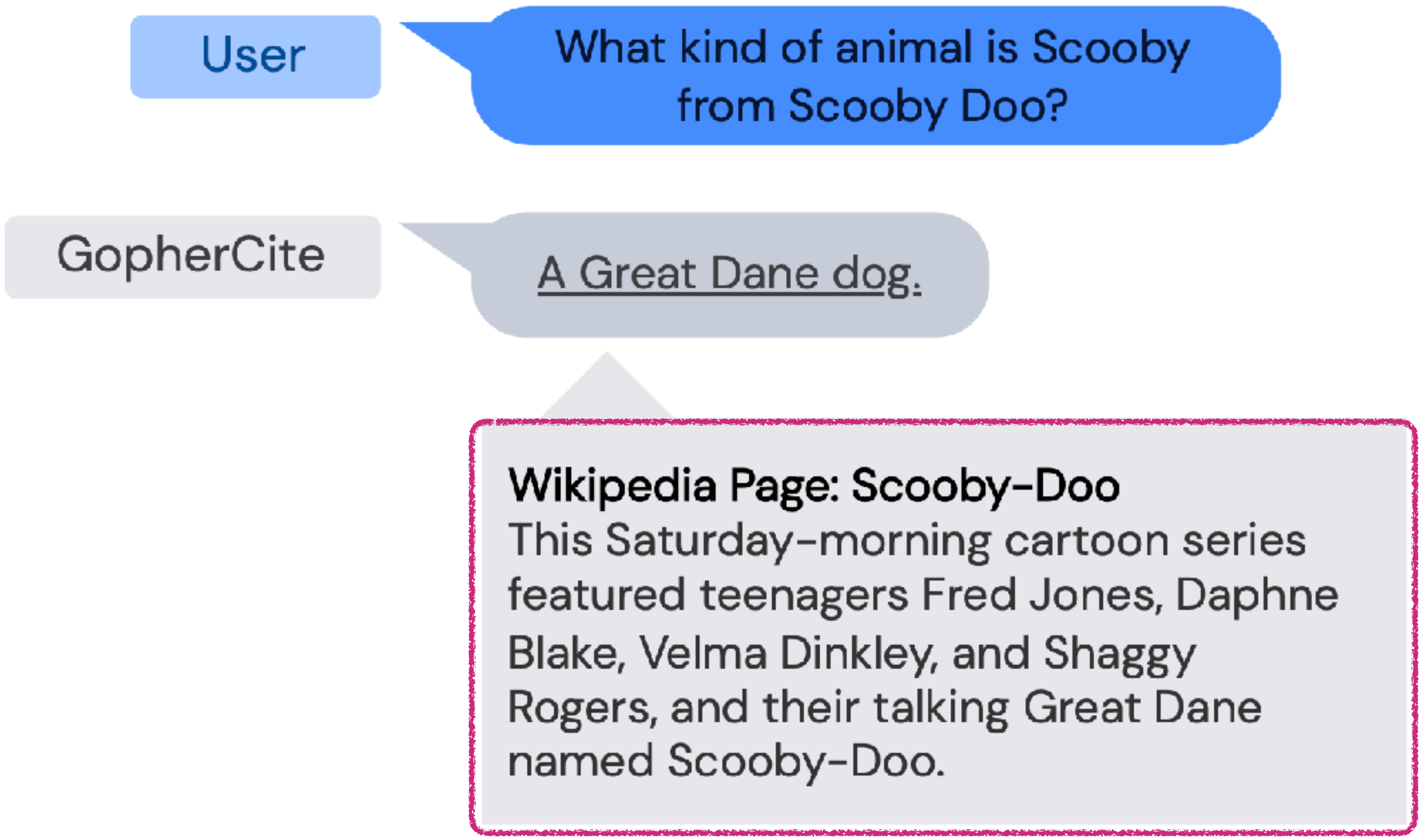
- Unlabeled Wikipedia / CC
- Web (Google / Bing Search Results)
- Training data

# GopherCite (Menick et al., 2022)




Menick et al. 2022. “GopherCite: Teaching language models to support answers with verified quotes”

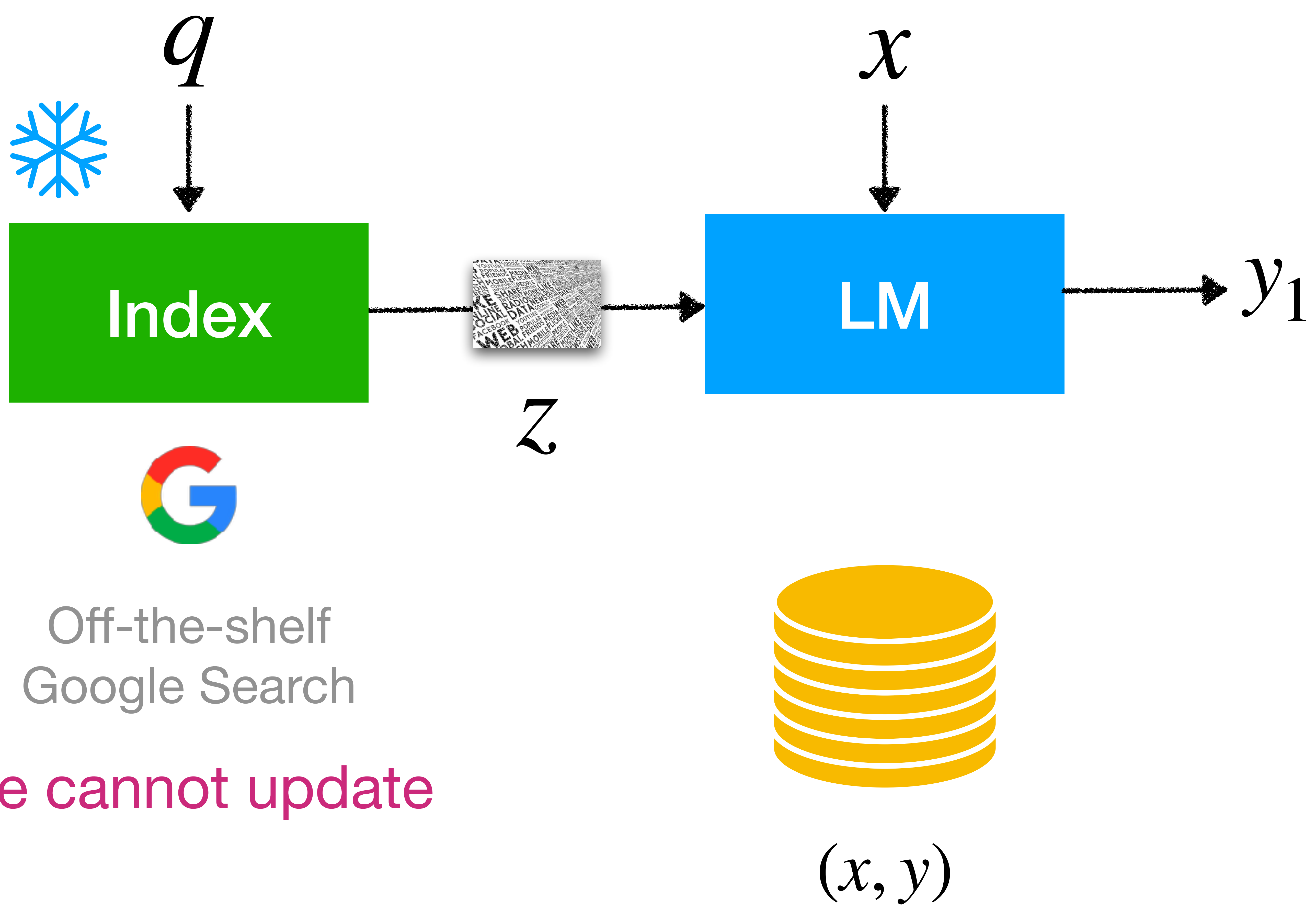
# GopherCite (Menick et al., 2022)




Extract and generate a quote to support an answer

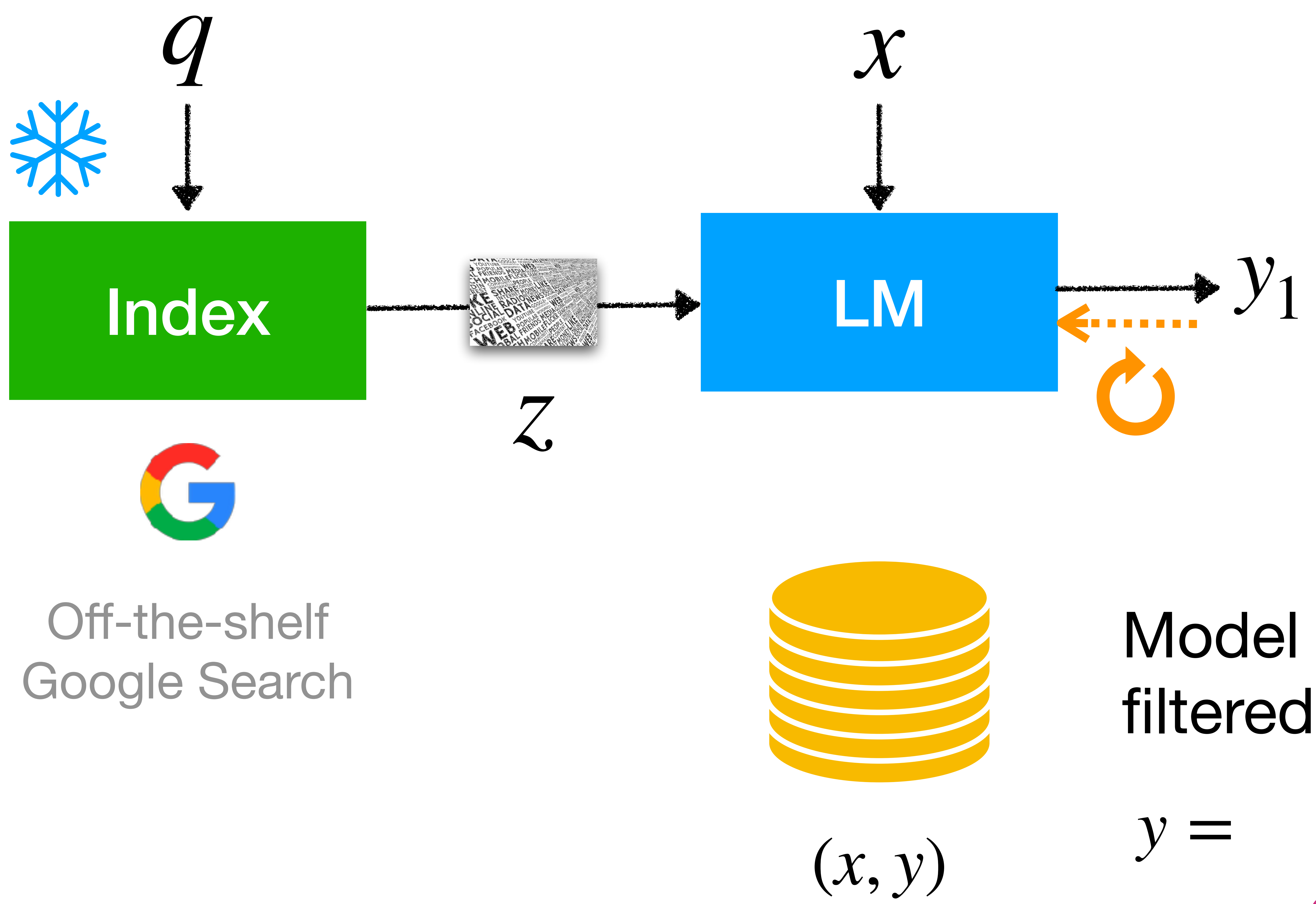
Menick et al. 2022. “GopherCite: Teaching language models to support answers with verified quotes”

# GopherCite: RLHF for answering with verified quotes




Supervised fine-tuning (SFT)

# GopherCite: RLHF for answering with verified quotes



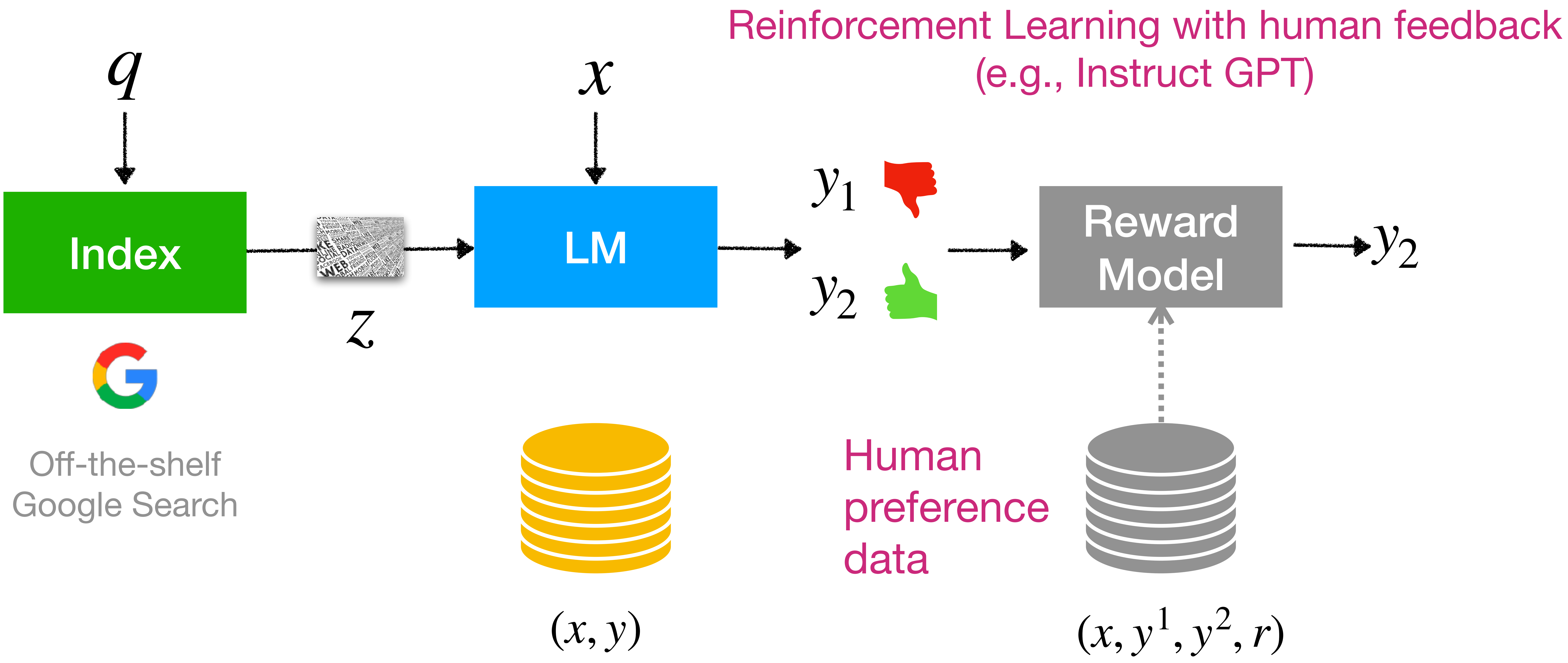

Supervised fine-tuning (SFT)

Model generated training data filtered by human

$y =$

%<Claim>%(Document title)%[Quote from document]%

# GopherCite: RLHF for answering with verified quotes



# GopherCite: RLHF for answering with verified quotes

Human rater

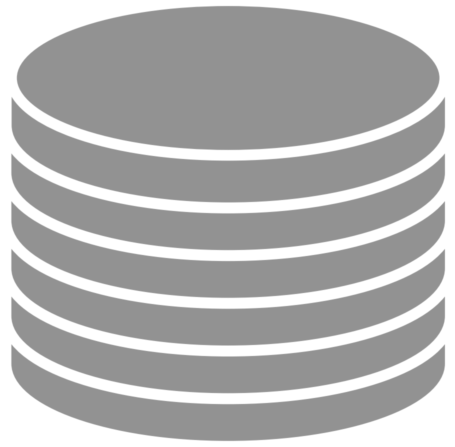
$(x, y^1, y^2) \longrightarrow$

**Plausible** as an answer to the input

**Supported** by accompanying evidence

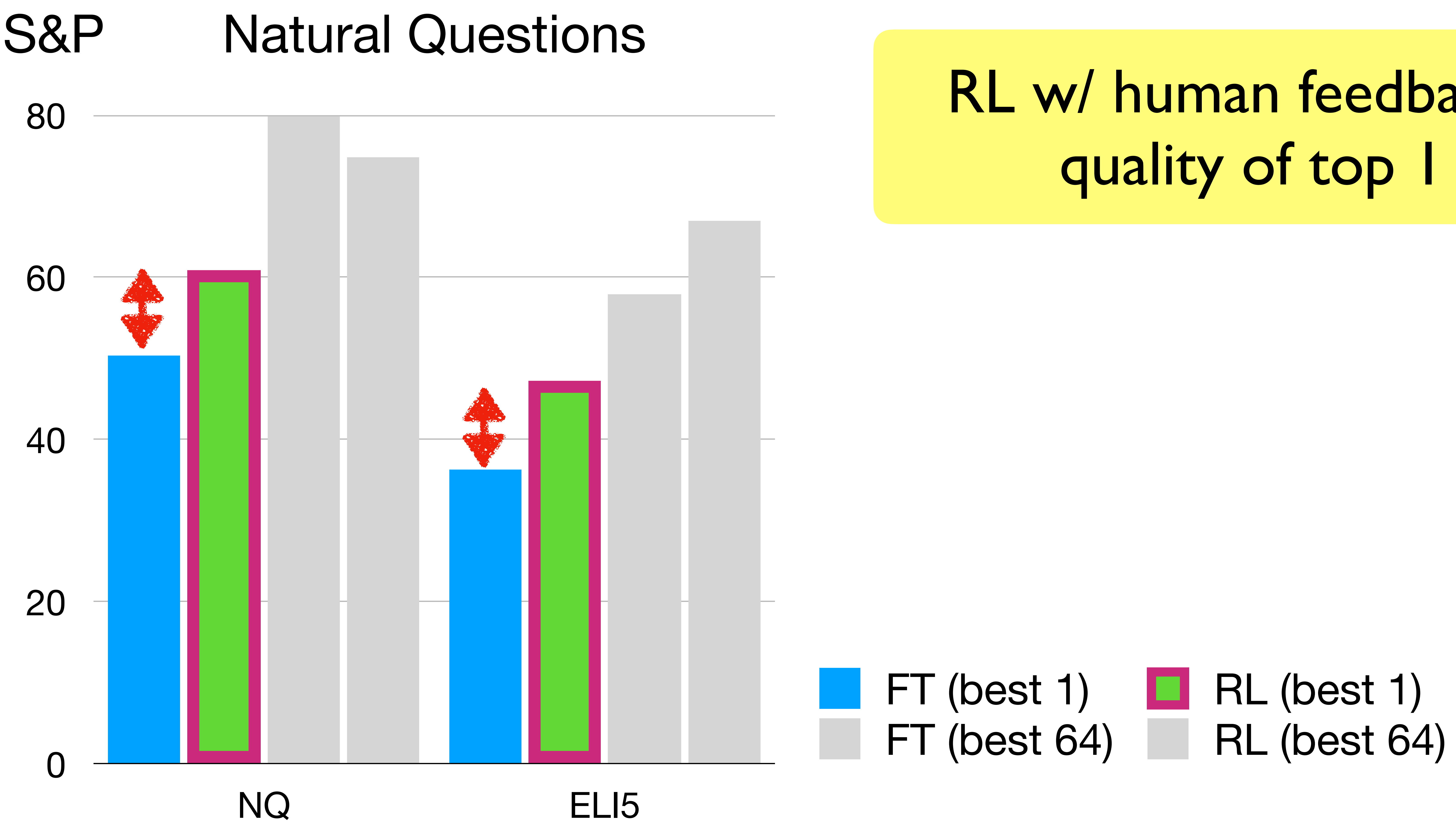
$\longrightarrow r \in y_1, y_2$

33k Human preference data



$(x, y^1, y^2, r)$

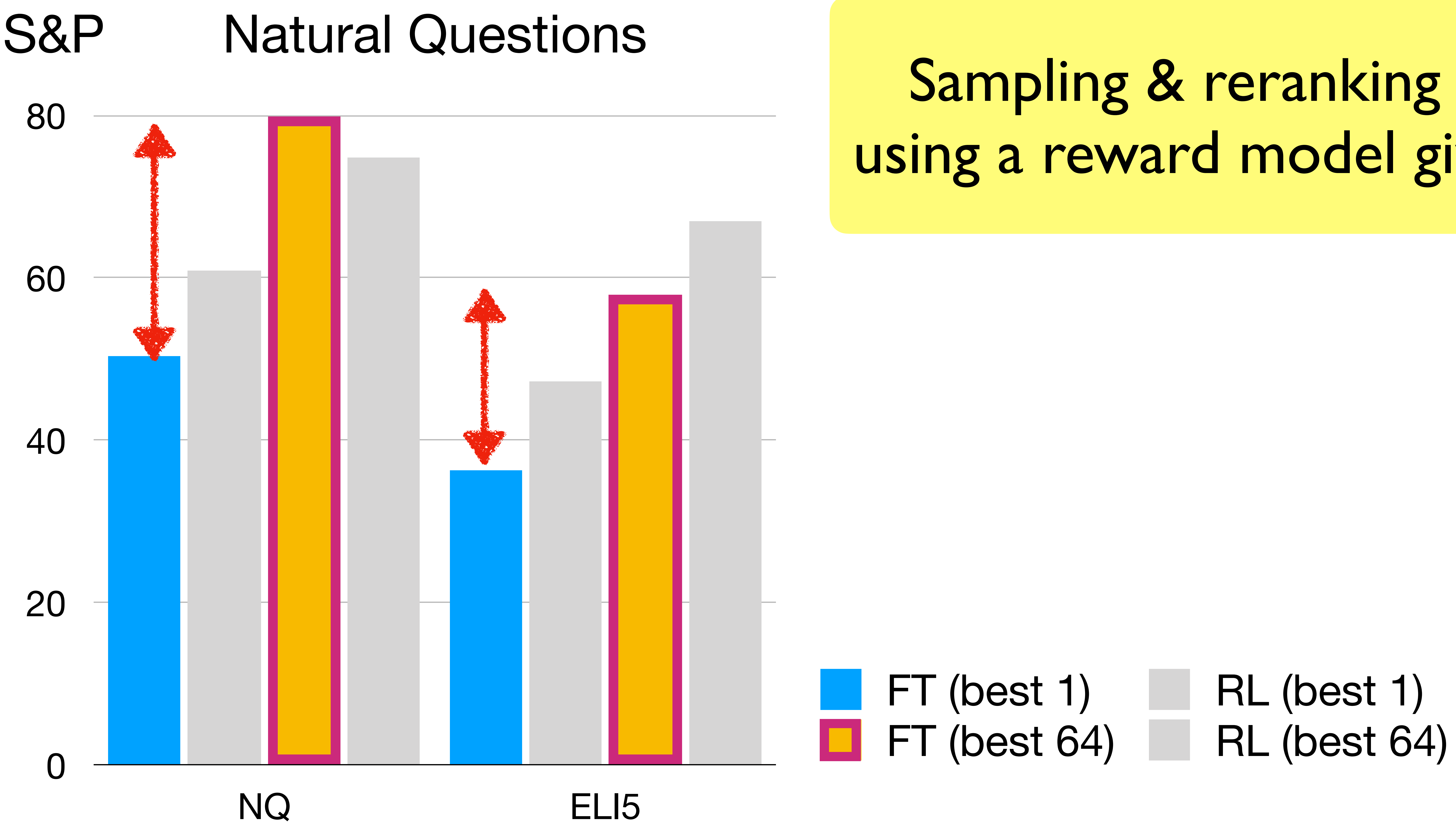
# Effects of RL




RL w/ human feedback improves the quality of top 1 generations

(S&P = Supported & Plausible)

# Effects of RL




Sampling & reranking many generations using a reward model gives gains from Top 1

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |
| GopherCite (Menick et al., 2022), also WebGPT (Nakano et al., 2021) | Open-domain QA, Long-form QA | Fine-tuning + RL (LM) | Google Search Results |

Benefit of **fine-tuning**

- Customizable
- Competitive w/ more data
- Requiring training

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |
| GopherCite (Menick et al., 2022), also WebGPT (Nakano et al., 2021) | Open-domain QA, Long-form QA | Fine-tuning + RL (LM) | Google Search Results |

Benefit of **RL**

Better alignment with user preferences

Requiring additional data collection (preference)

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |
| GopherCite (Menick et al., 2022), also WebGPT (Nakano et al., 2021) | Open-domain QA, Long-form QA | Fine-tuning + RL (LM) | Google Search Results |

What if we cannot train LMs for downstream tasks?
(e.g., lack of computational resources / proprietary LM … etc)

# Downstream adaptation of retrieval-based LMs

What are the **tasks**?

- Open-domain QA
- Other knowledge-intensive tasks
- General NLU
- Language Modeling & other generation tasks

How to **adapt?**

- Fine-tuning
- Reinforcement learning
- **Prompting**

What is **data store?**

- Wikipedia
- Web (Google / Bing Search Results)
- Training data

# Prompting

$k$-shot instances (k=0, 32 … etc)

Doesn’t require LM training on tasks!

Q: who Is the US president
A: Joe Biden
##
Q: What is the capital of US?
A: Washington DC.
##

Training instances (demonstrations)

Q: what is the Ontario capital?
A:

Test instances

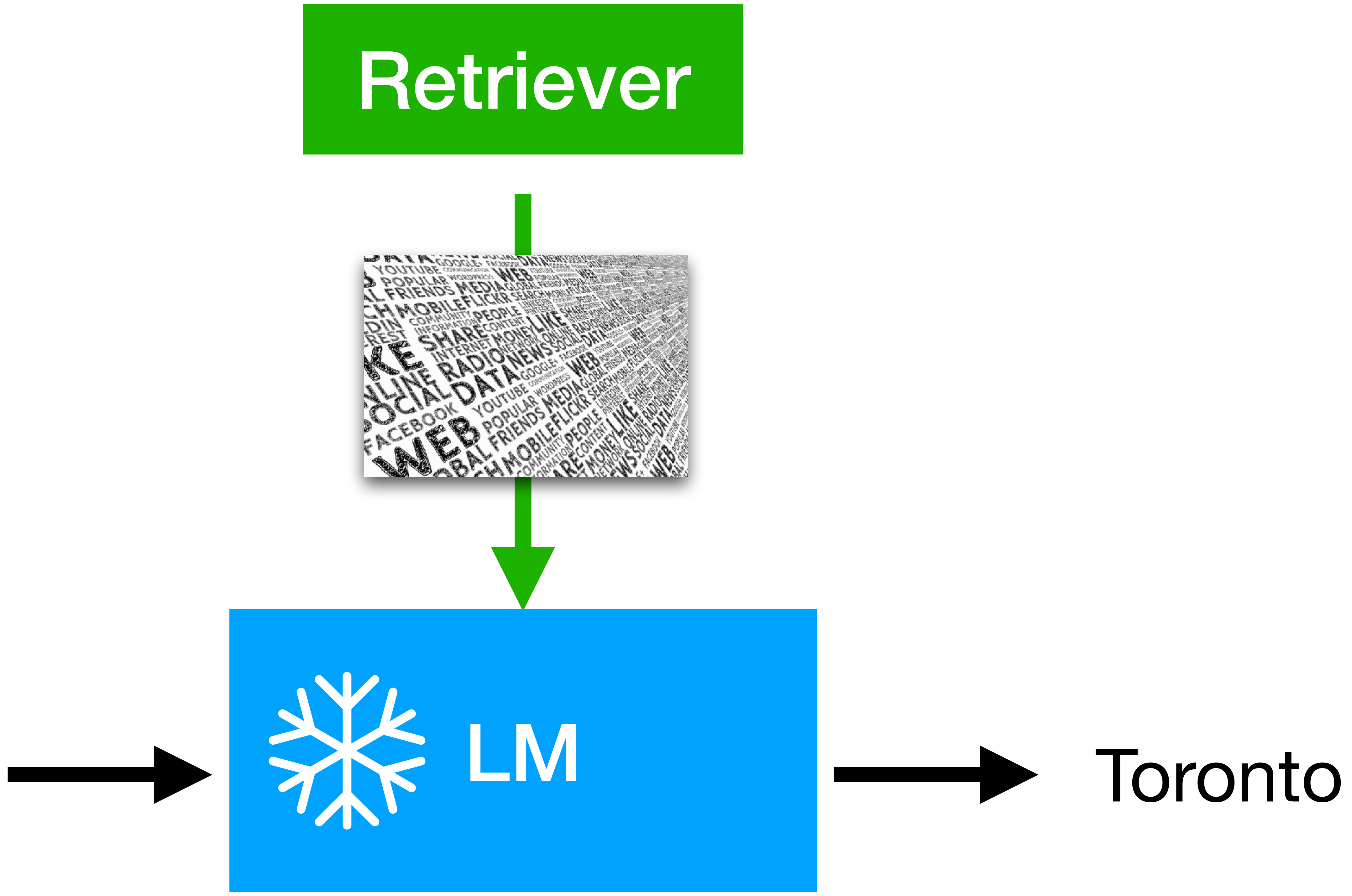
LM

Toronto

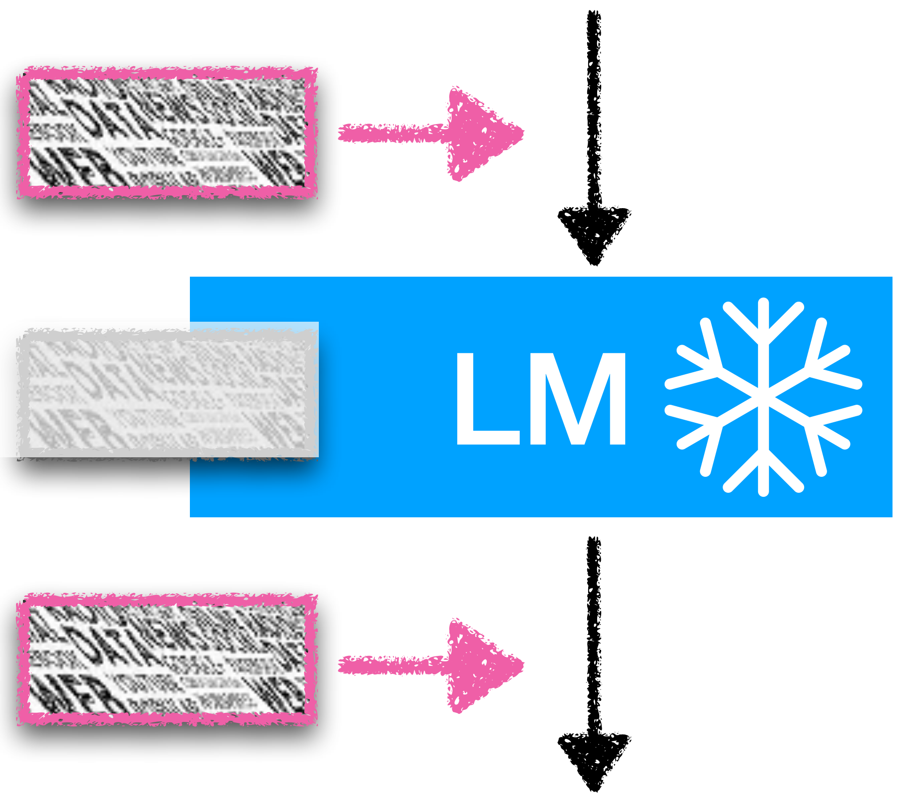
# Retrieval-based prompting

$k$-shot instances (k=0, 32 … etc)

Q: who Is the US president
A: Joe Biden
##
Q: What is the capital of US?
A: Washington DC.
##
Q: what is the Ontario capital?
A:



# Design choice of retrieval-based Prompting

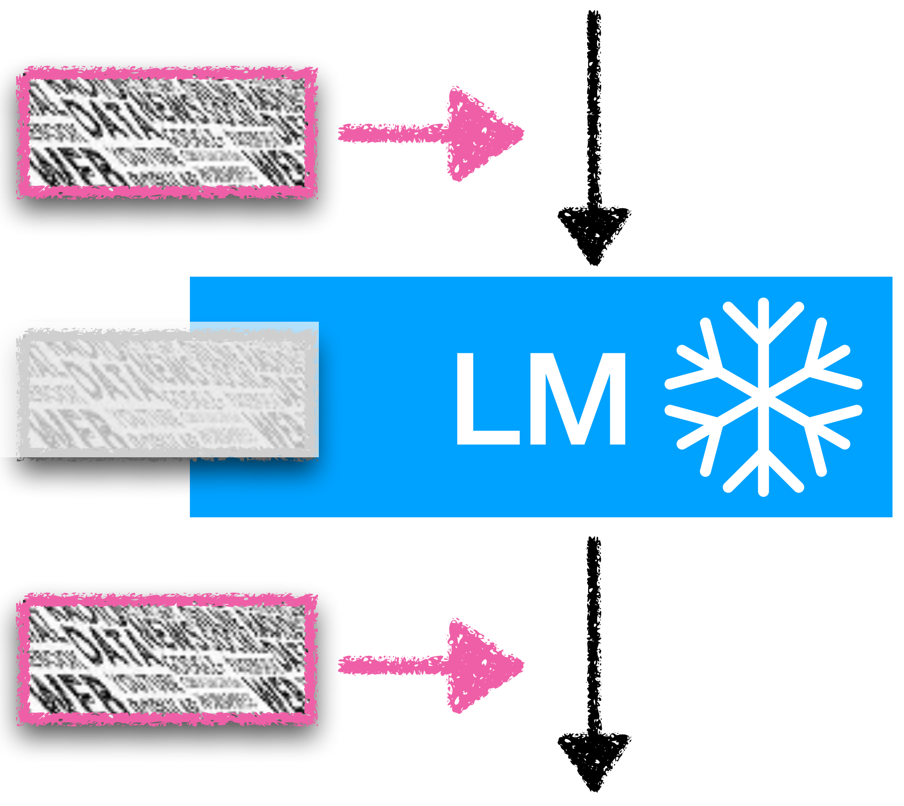



**Input space:**
Incorporate retrieved context in input space

**Intermediate layers:**
N/A

**Output space:**
Interpolate token probability distributions in output space

# Design choice of retrieval-based Prompting




**Input space:**
Incorporate retrieved context in input space

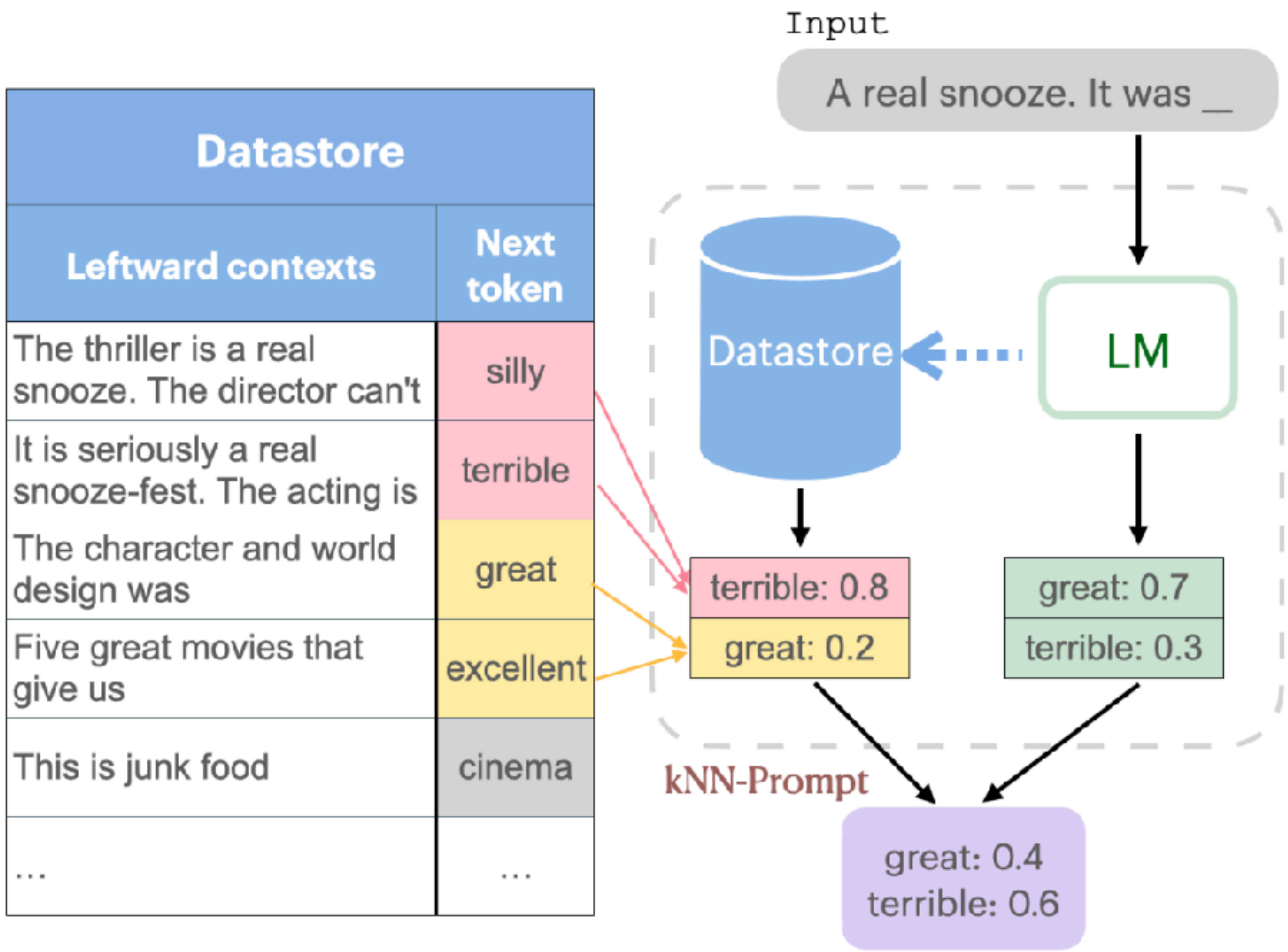
**Intermediate layers:**
N/A

**Output space:**
Interpolate token probability distributions in output space

Extending kNN-LM for downstream tasks

# kNN-Prompt (Shi et al., 2022)

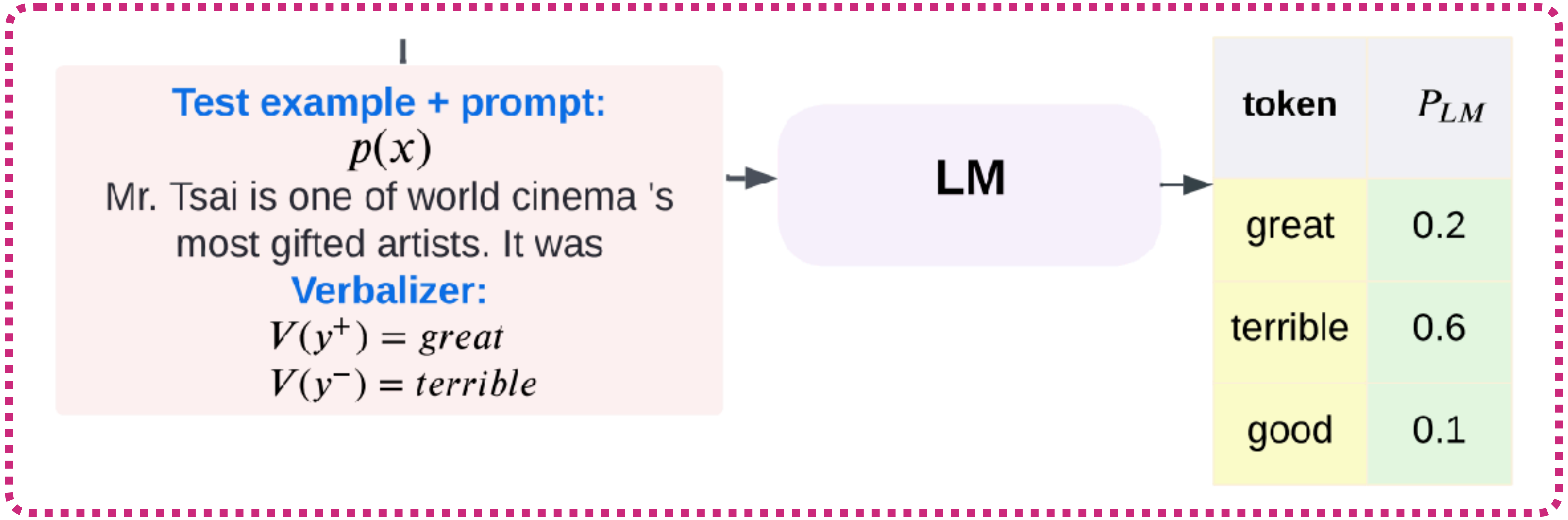
kNN LM with fuzzy verbalizers
for zero-/few-shot **classification**



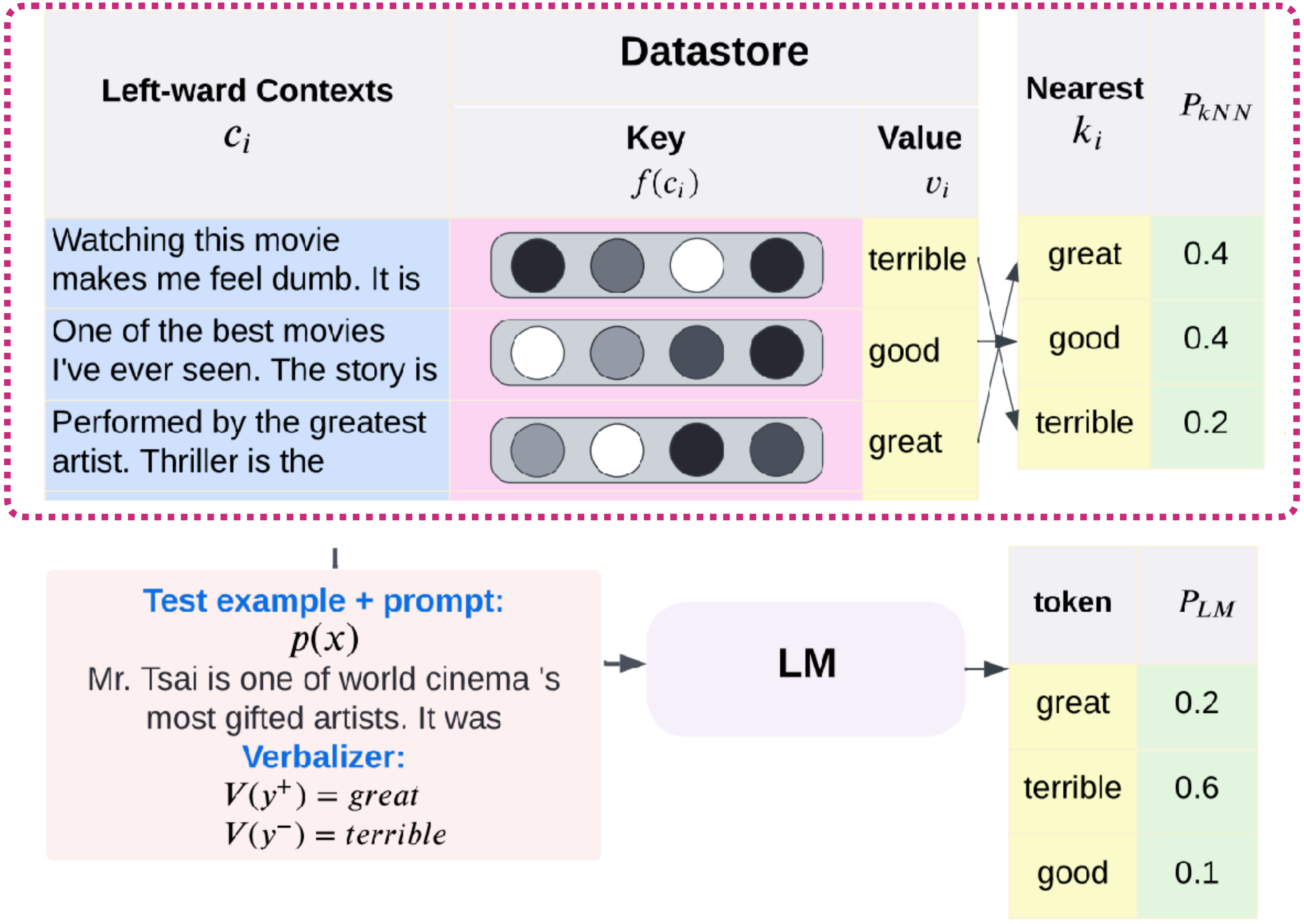

Shi et al. 2022. "Nearest Neighbor Zero-shot Inference"

# kNN-Prompt (Shi et al., 2022)

LM predicts next token

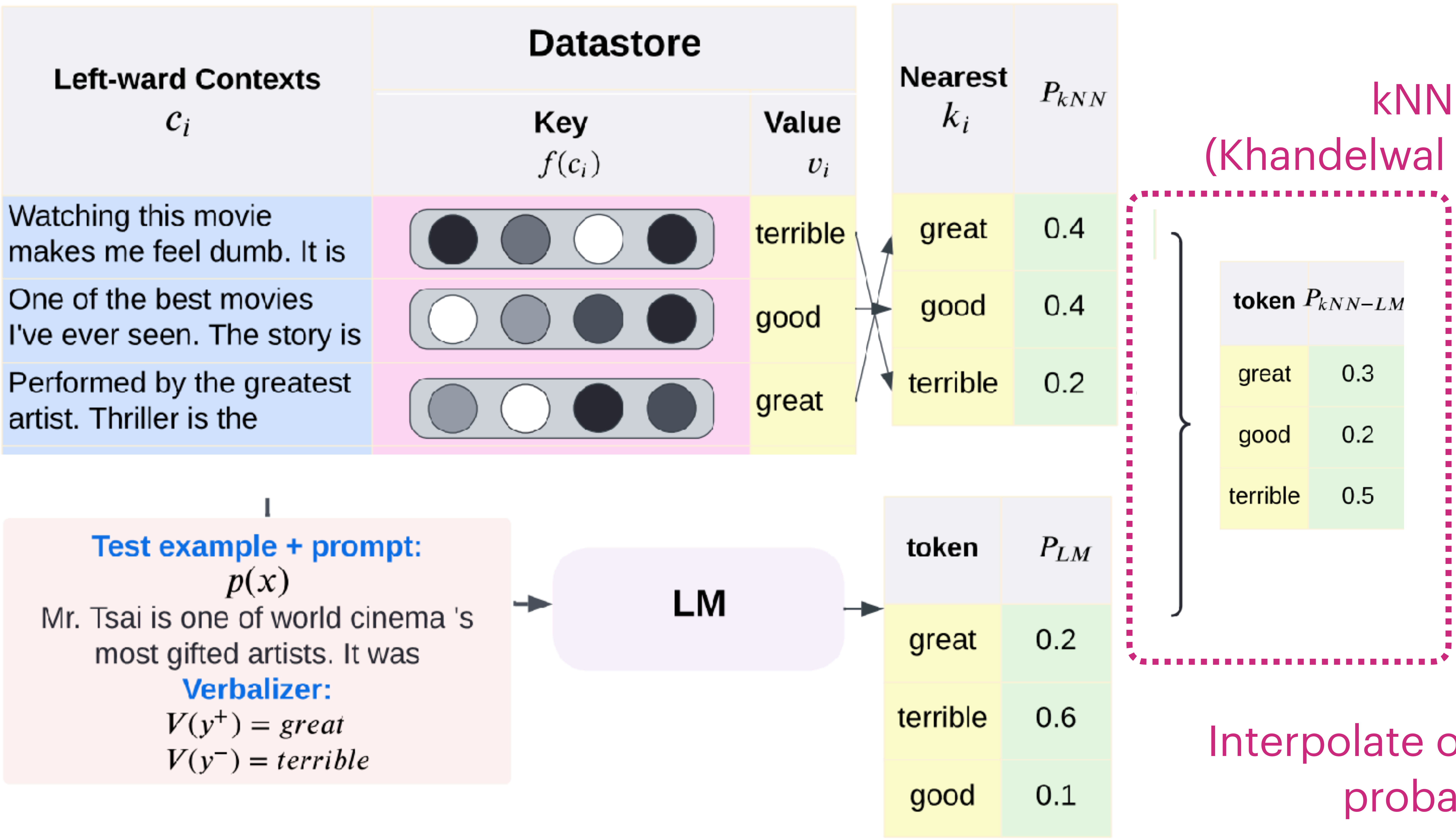


# kNN-Prompt (Shi et al., 2022)




kNN predicts next tokens

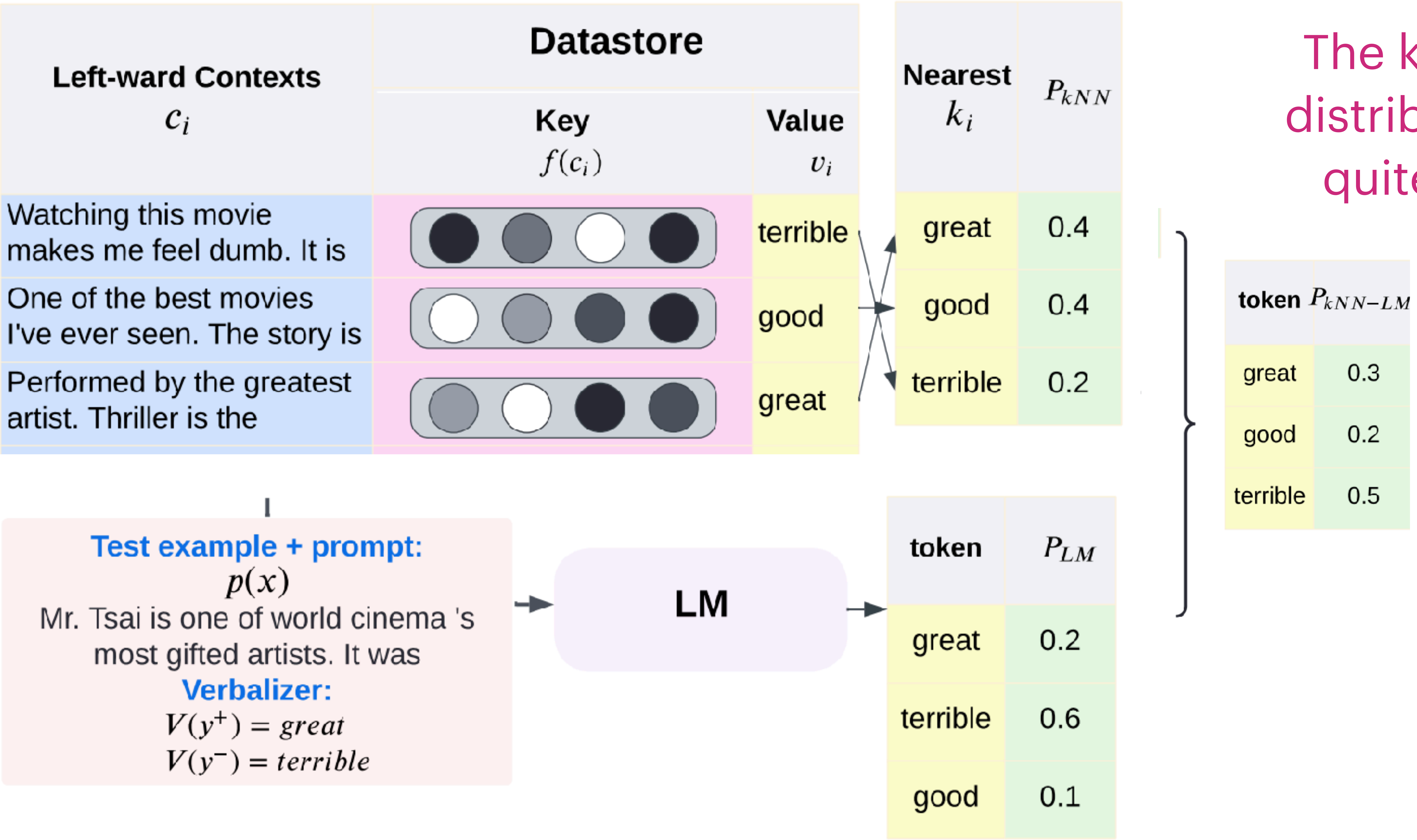
# kNN-Prompt (Shi et al., 2022)




kNN-LM
(Khandelwal et al., 2020)

Interpolate output token probability

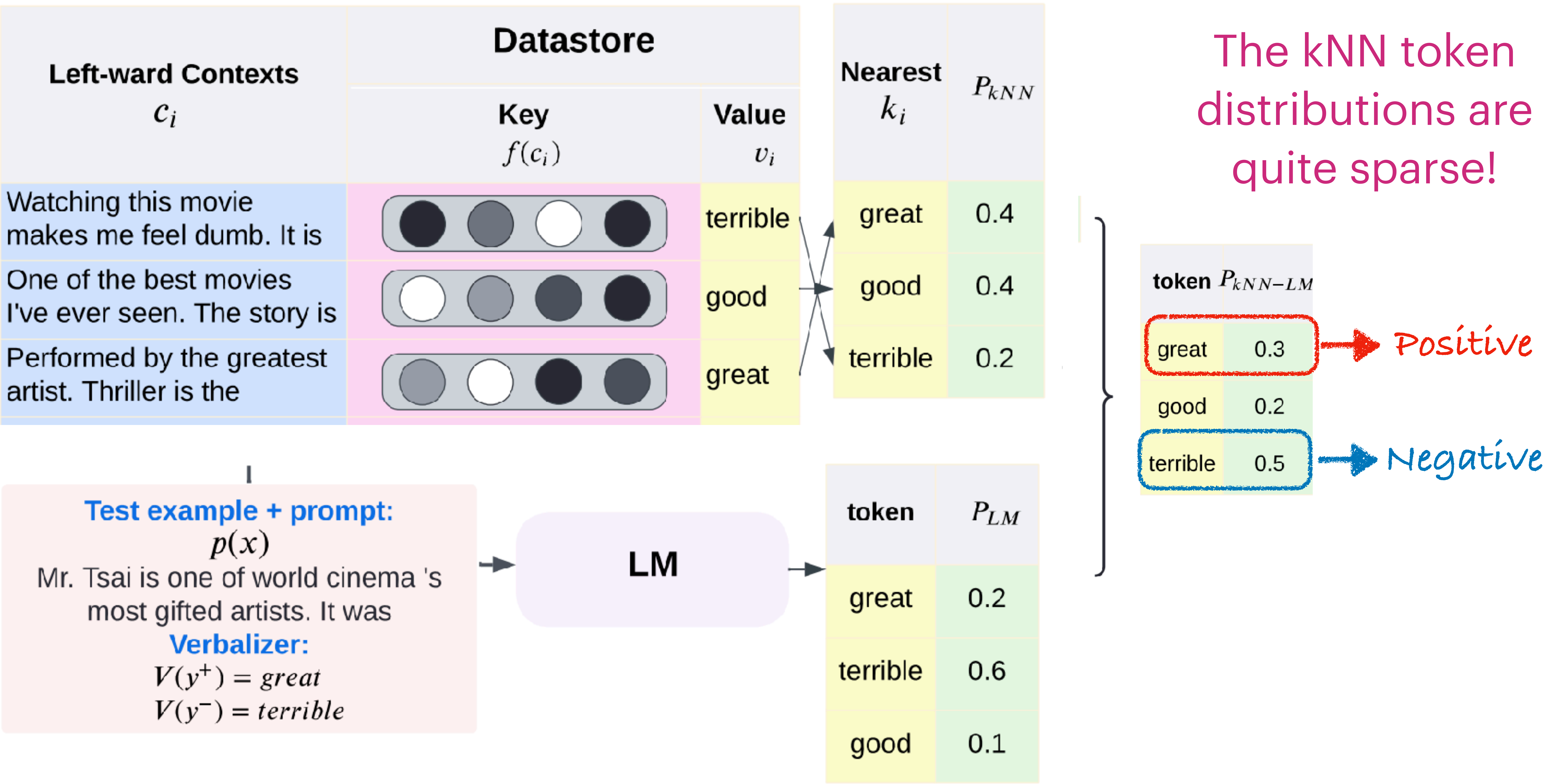
# kNN-Prompt (Shi et al., 2022)



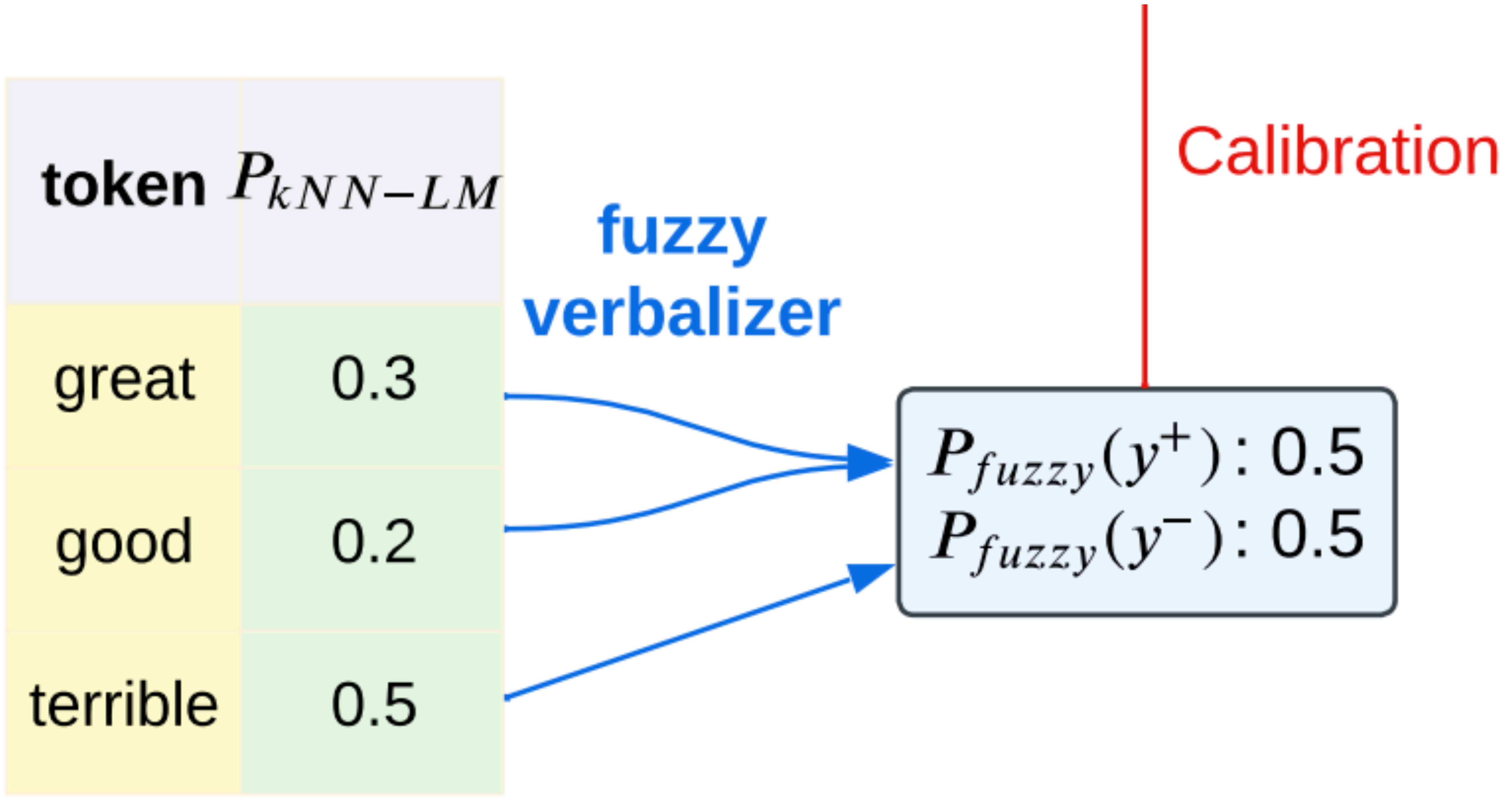

The kNN token distributions are quite sparse!

# kNN-Prompt (Shi et al., 2022)

The kNN token distributions are quite sparse!



# kNN-Prompt (Shi et al., 2022)

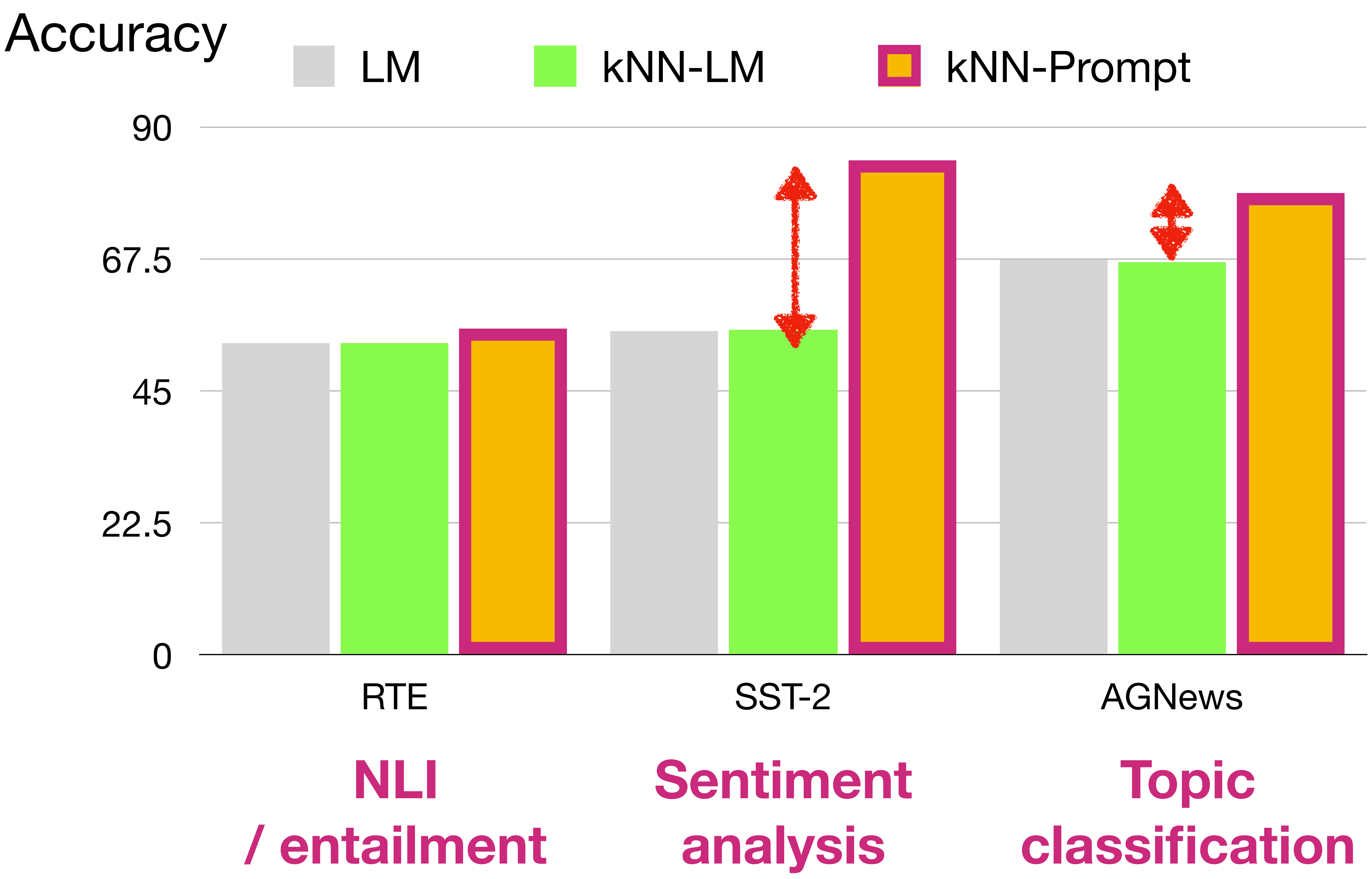



**Fuzzy verbalizer** maps token probability to target class labels

$$P_{FV}(y \mid x) \propto \sum_{v_i \in \mathcal{N}(v)} P(v_i \mid p(\mathbf{x}))$$

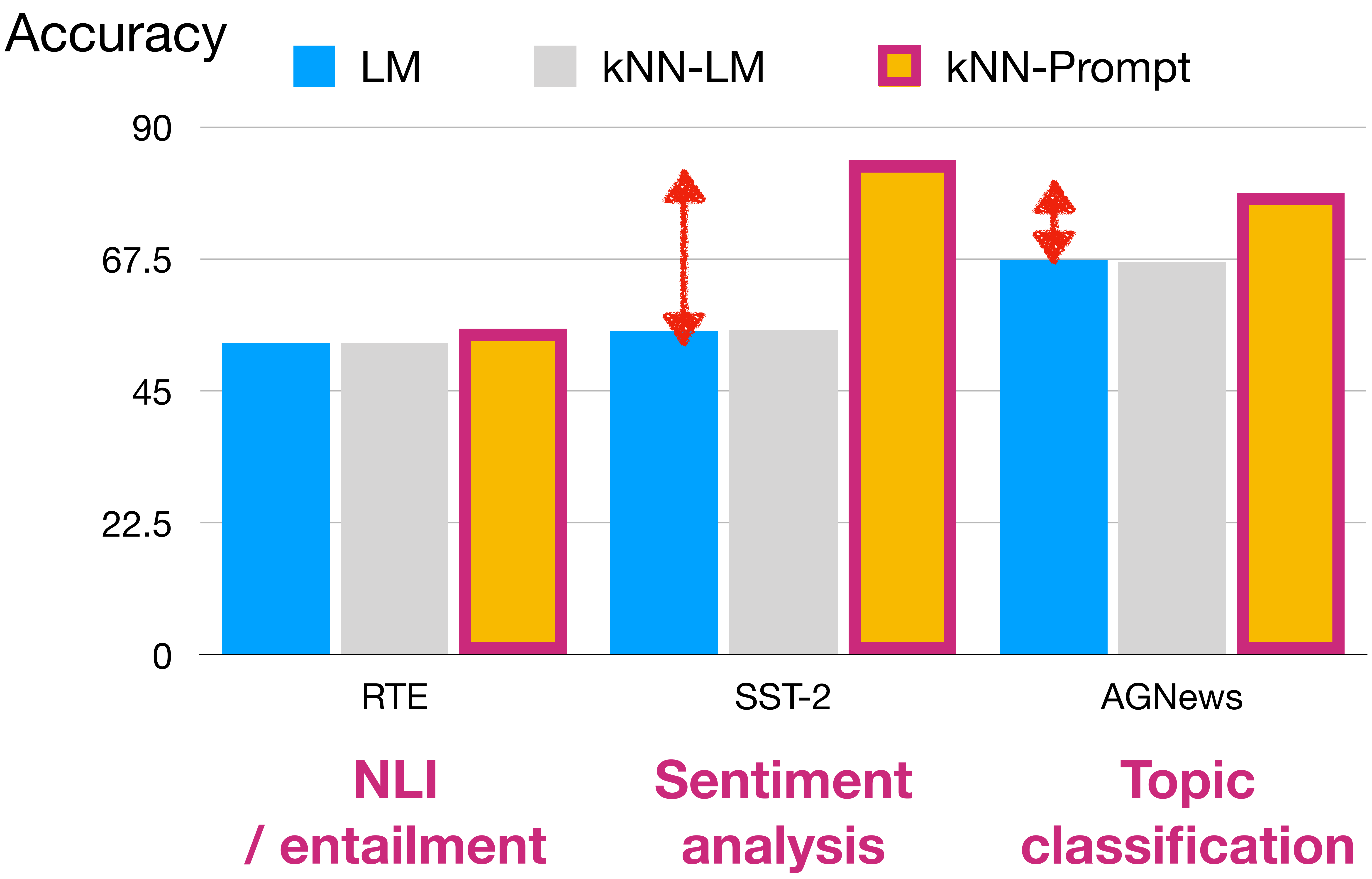
Find similar tokens using GloVe & ConceptNet

# Results on diverse classification tasks




Significant gains from kNN-LM

# Results on diverse classification tasks



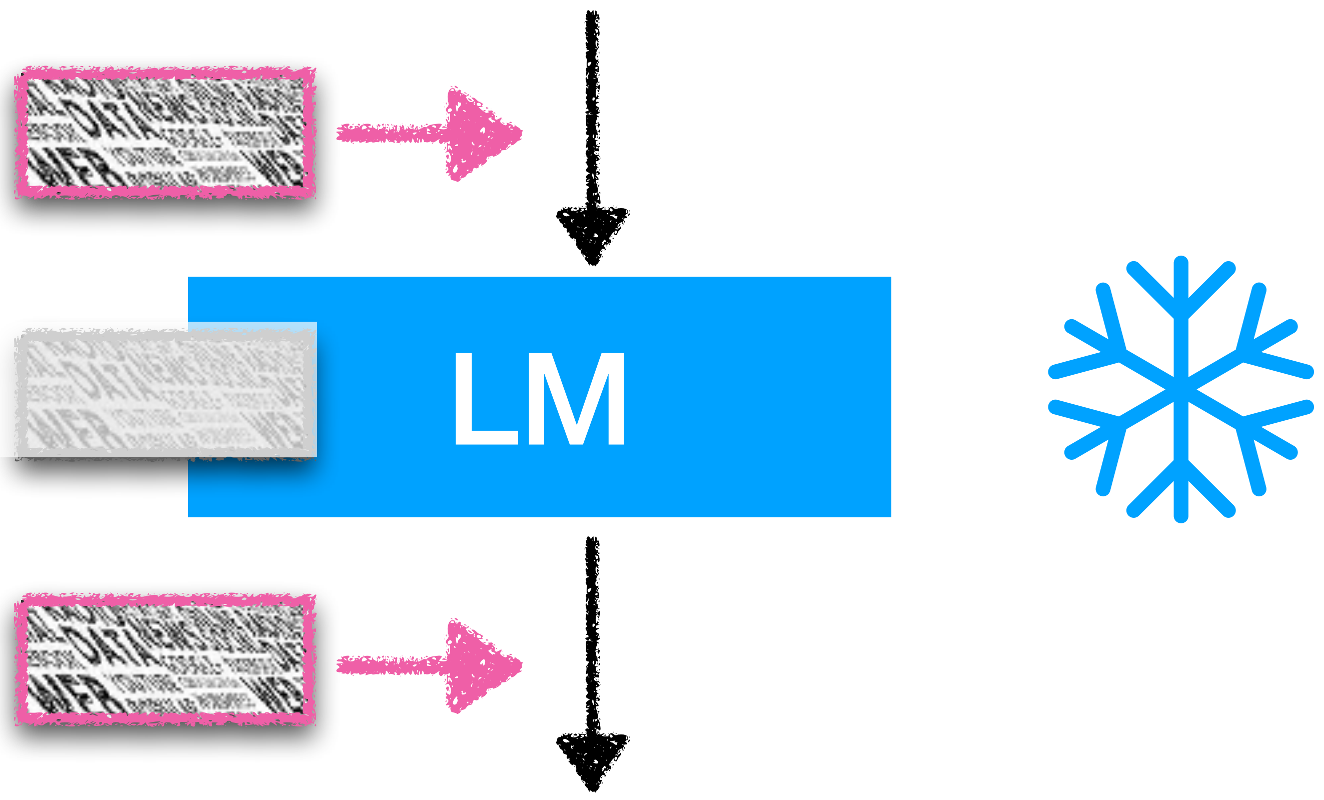

kNN-prompt largely outperforms vanilla LM in zero-shot classification

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |
| GopherCite (Menick et al., 2022) | Open-domain QA, Long-form QA | Fine-tuning + RL (LM) | Google Search Results |
| kNN-prompt (Shi et al., 2022) | Classification | Prompting (output) | Wikipedia \| Pile |

Retrieval-based LMs are effective in general NLU tasks!

# Retrieval-based Prompting

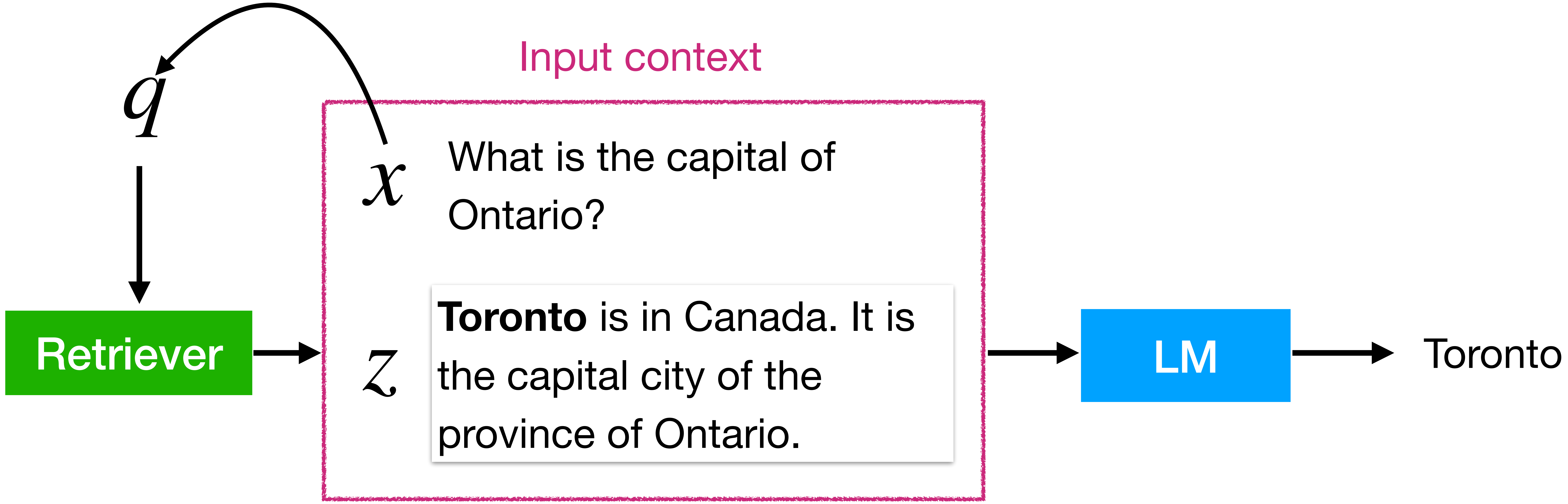



**Input space:**
Incorporate retrieved context in input space

**Intermediate layers:**
N/A

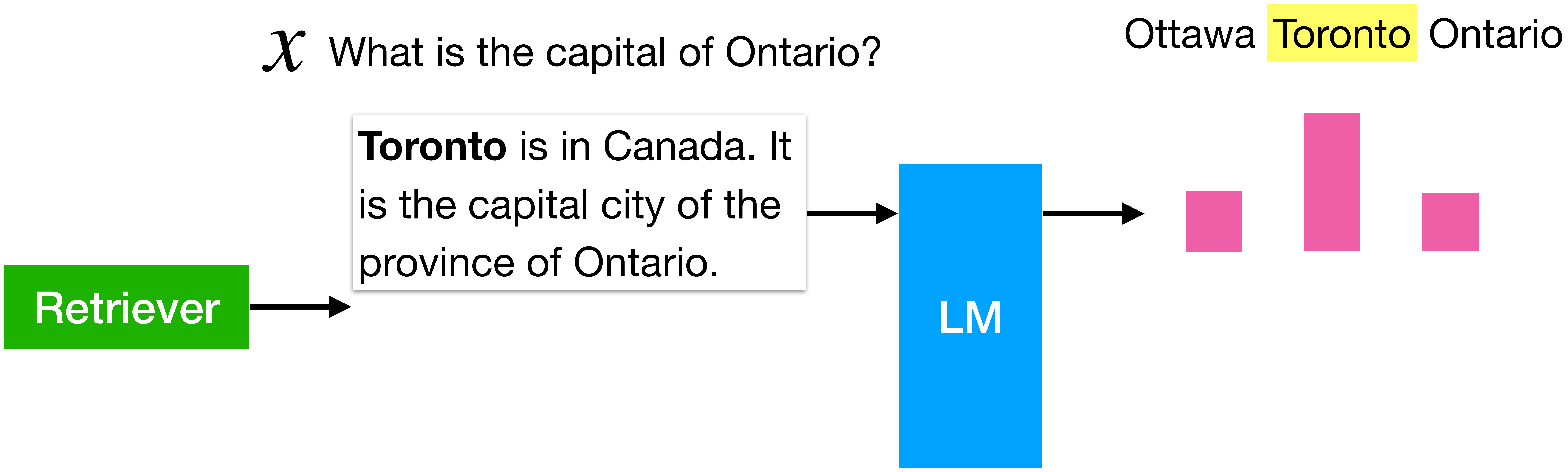
**Output space:**
Interpolate token probability distributions in output space

# Retrieval-in-context

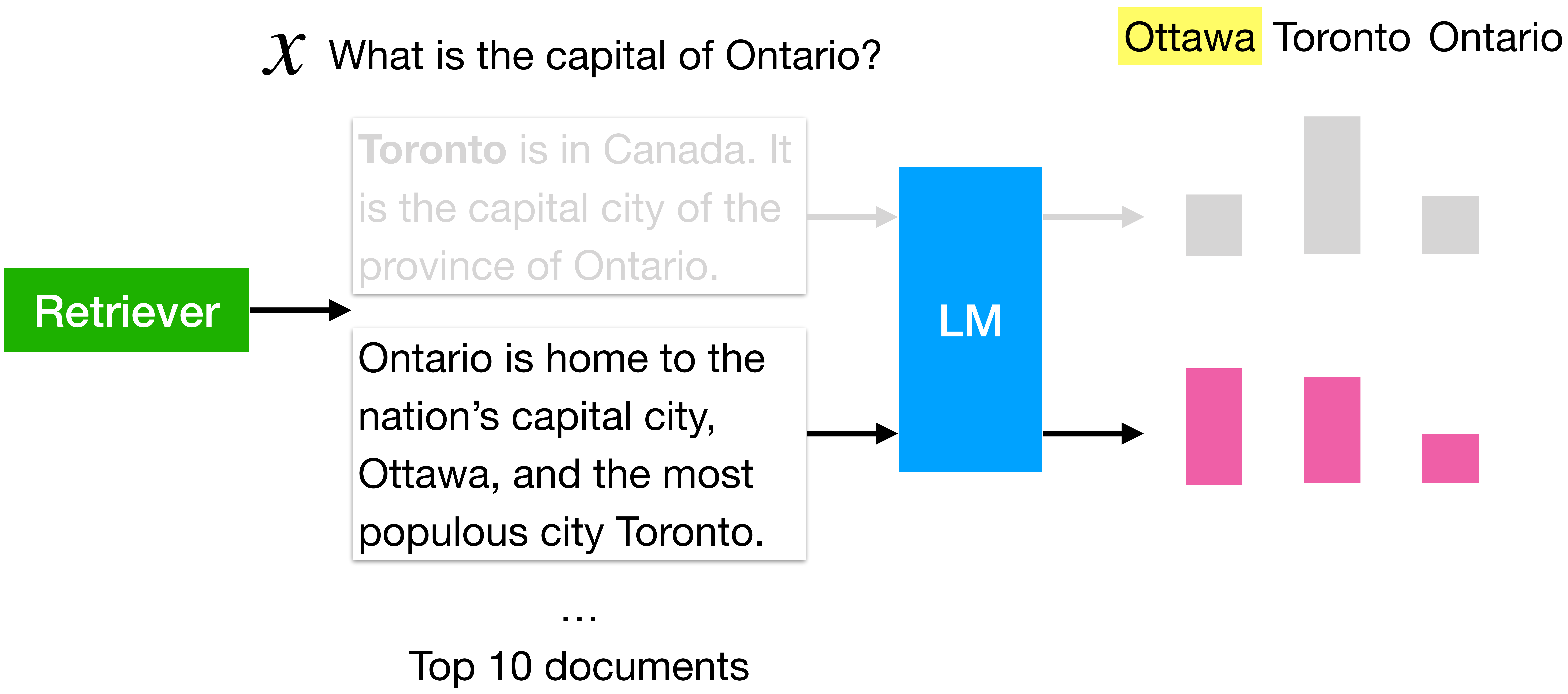



(Shi et al., 2023; Ram et al., 2022; Mallen et al., 2022; Yu et al., 2022; Press et al., 2022; *inter alia*)

# REPLUG (Shi et al., 2023; Section 3&4)



# REPLUG (Shi et al., 2023; Section 3&4)



# REPLUG (Shi et al., 2023; Section 3&4)

$x$ What is the capital of Ontario?

Retriever

**Toronto** is in Canada. It is the capital city of the province of Ontario.

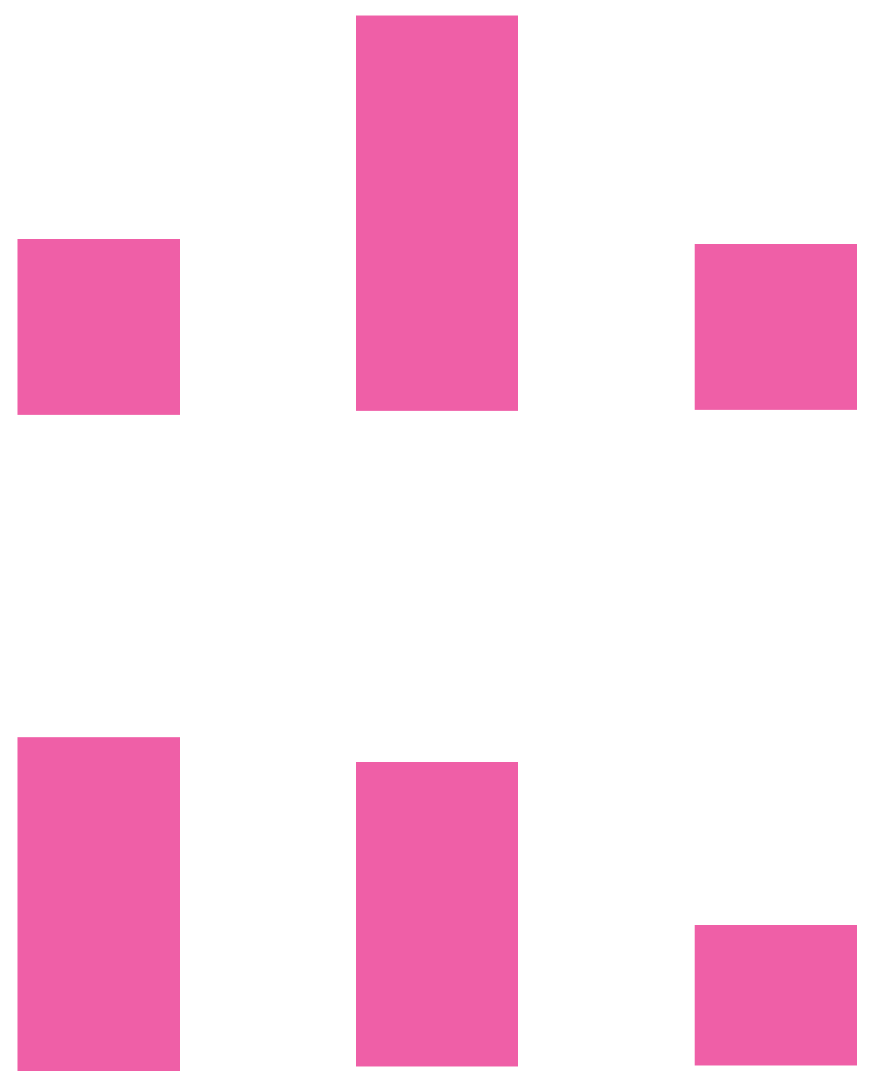
Ontario is home to the nation's capital city, Ottawa, and the most populous city Toronto.

...

Top 10 documents

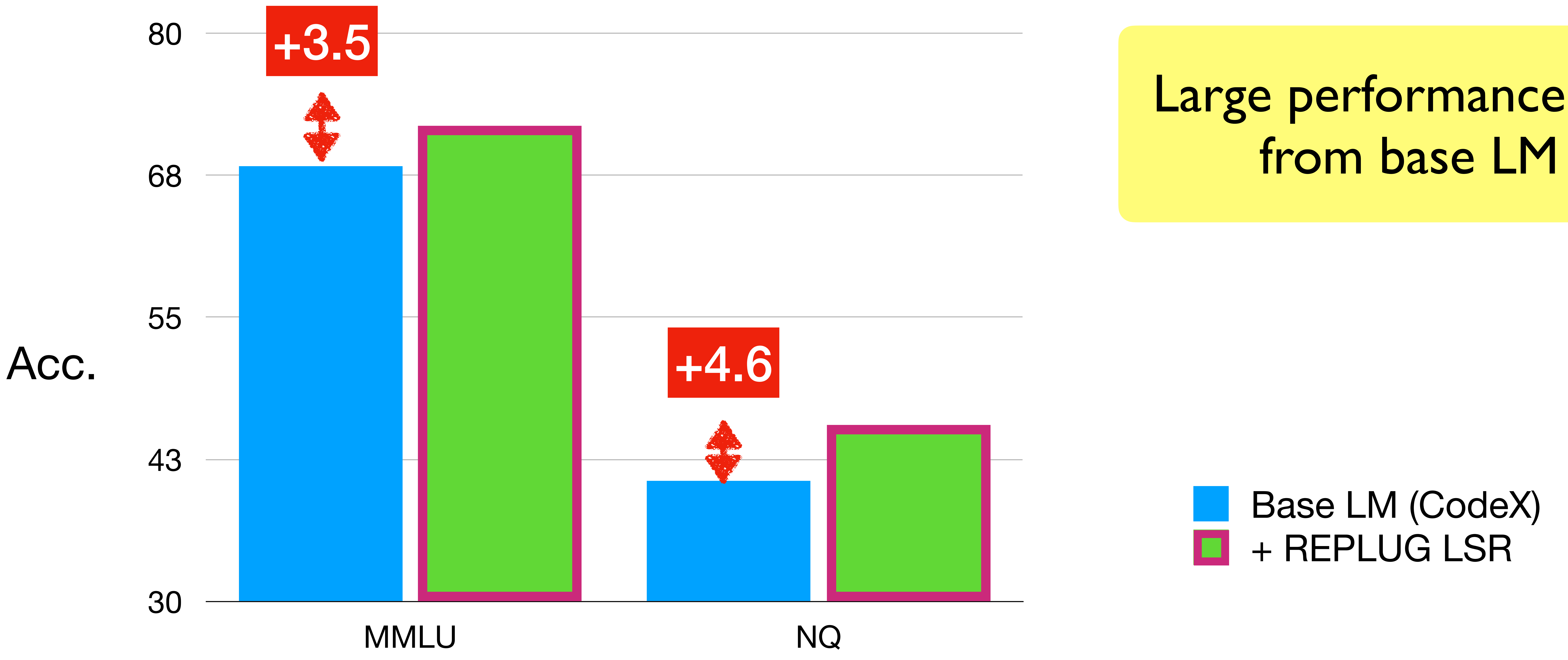
LM

Ottawa Toronto Ontario



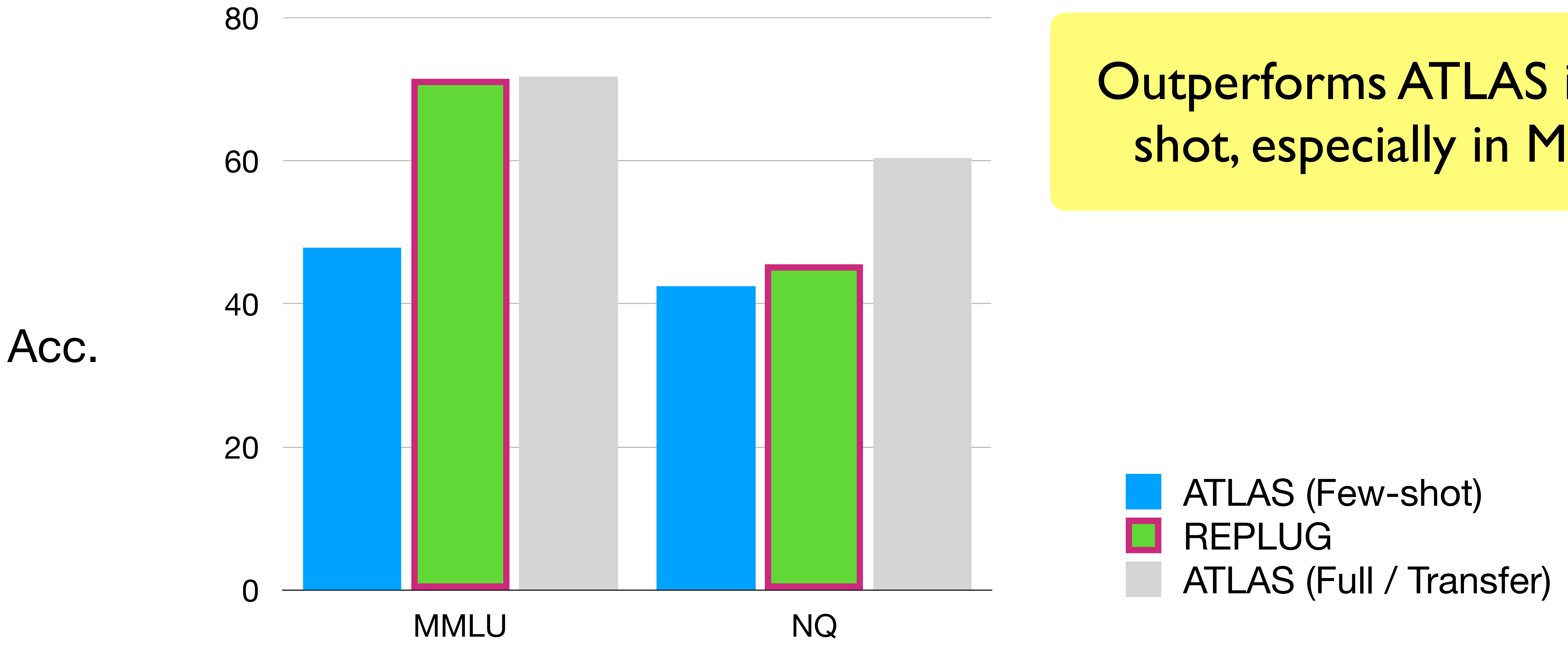
$$\sum_i p(\text{Toronto} \mid x, p_i) > \sum_i p(\text{Ottawa} \mid x, p_i)$$

# REPLUG: Results on QA & MMLU



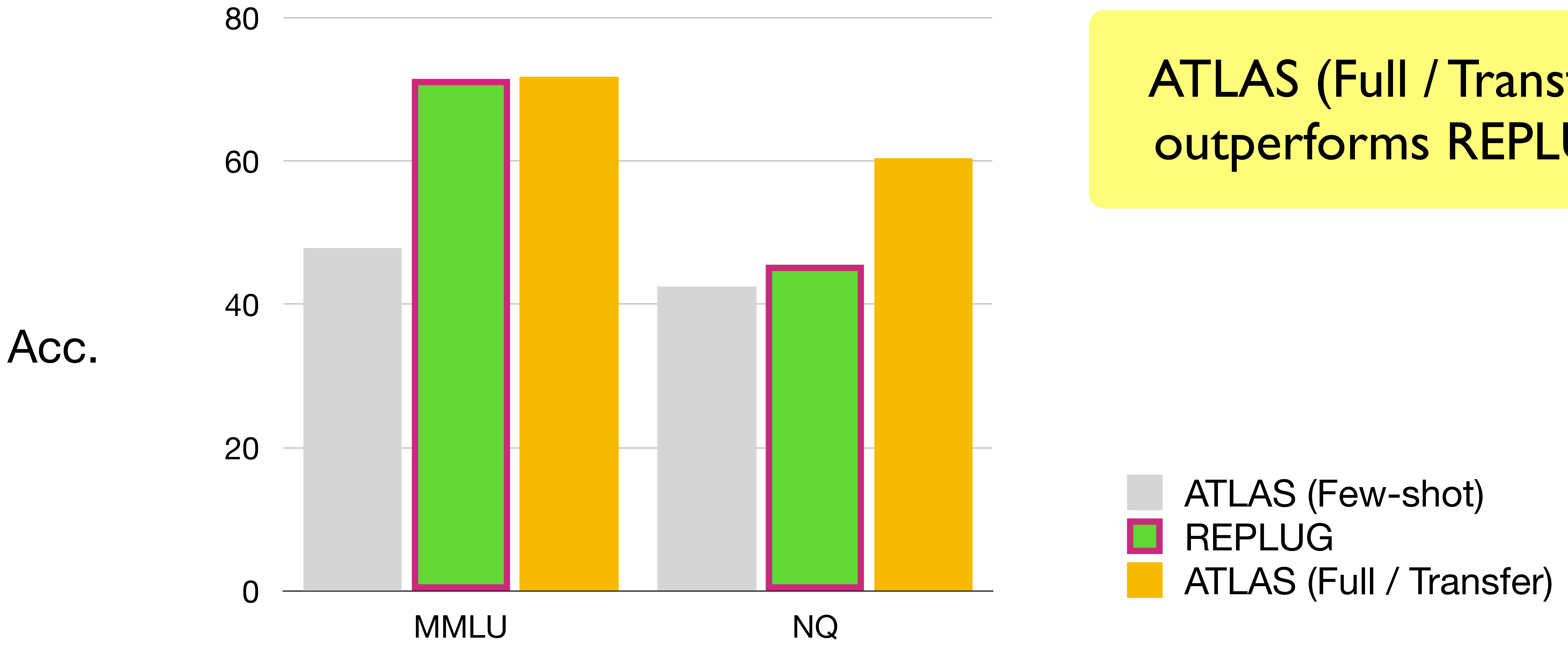

Large performance gain from base LM

# REPLUG: Comparison with ATLAS




Outperforms ATLAS in few-shot, especially in MMLU

# REPLUG: Comparison with ATLAS




ATLAS (Full / Transfer) outperforms REPLUG

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |
| GopherCite (Menick et al., 2022) | Open-domain QA, Long-form QA | Fine-tuning + RL (LM) | Google Search Results |
| kNN-prompt (Shi et al., 2022) | Classification | Prompting (output) | Wikipedia \| CC |
| REPLUG (Shi et al., 2023) | Knowledge-intensive | Prompting (input) | Wikipedia \| CC |

Benefit of **retrieval-based** prompting

No training & strong performance

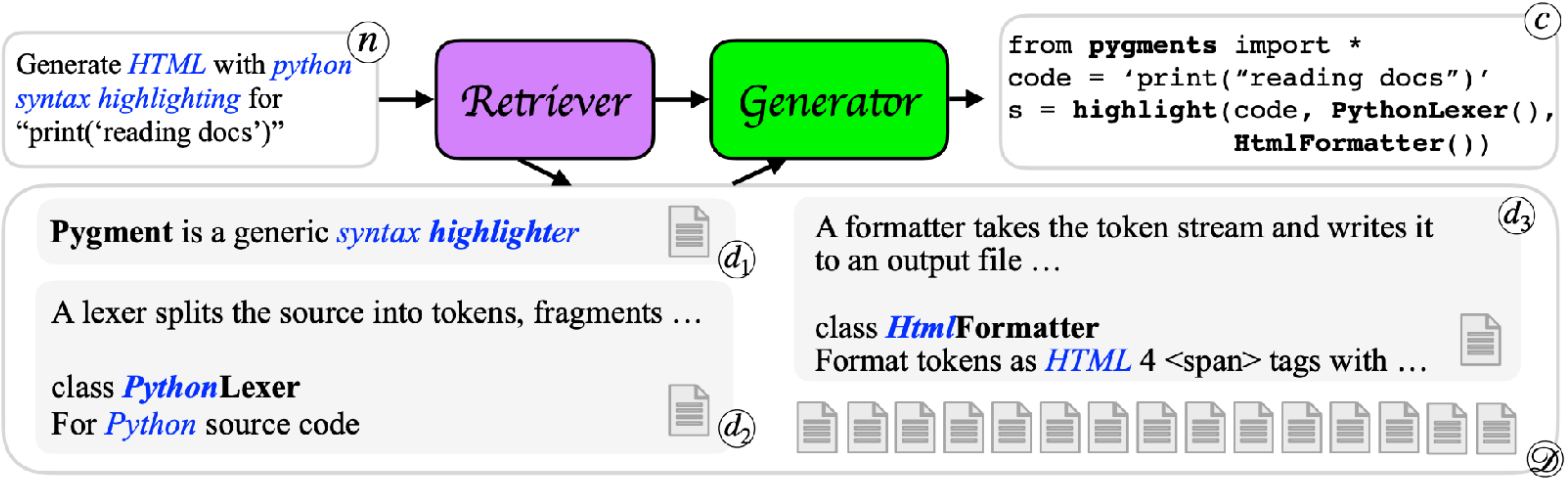
Hard to control, underperforming full FT model

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |
| GopherCite (Menick et al., 2022) | Open-domain QA, Long-form QA | Fine-tuning + RL (LM) | Google Search Results |
| kNN-prompt (Shi et al., 2022) | Classification | Prompting (output) | Wikipedia \| CC |
| REPLUG (Shi et al., 2023) | Knowledge-intensive | Prompting (input) | Wikipedia \| CC |

What can be other types of datastores?

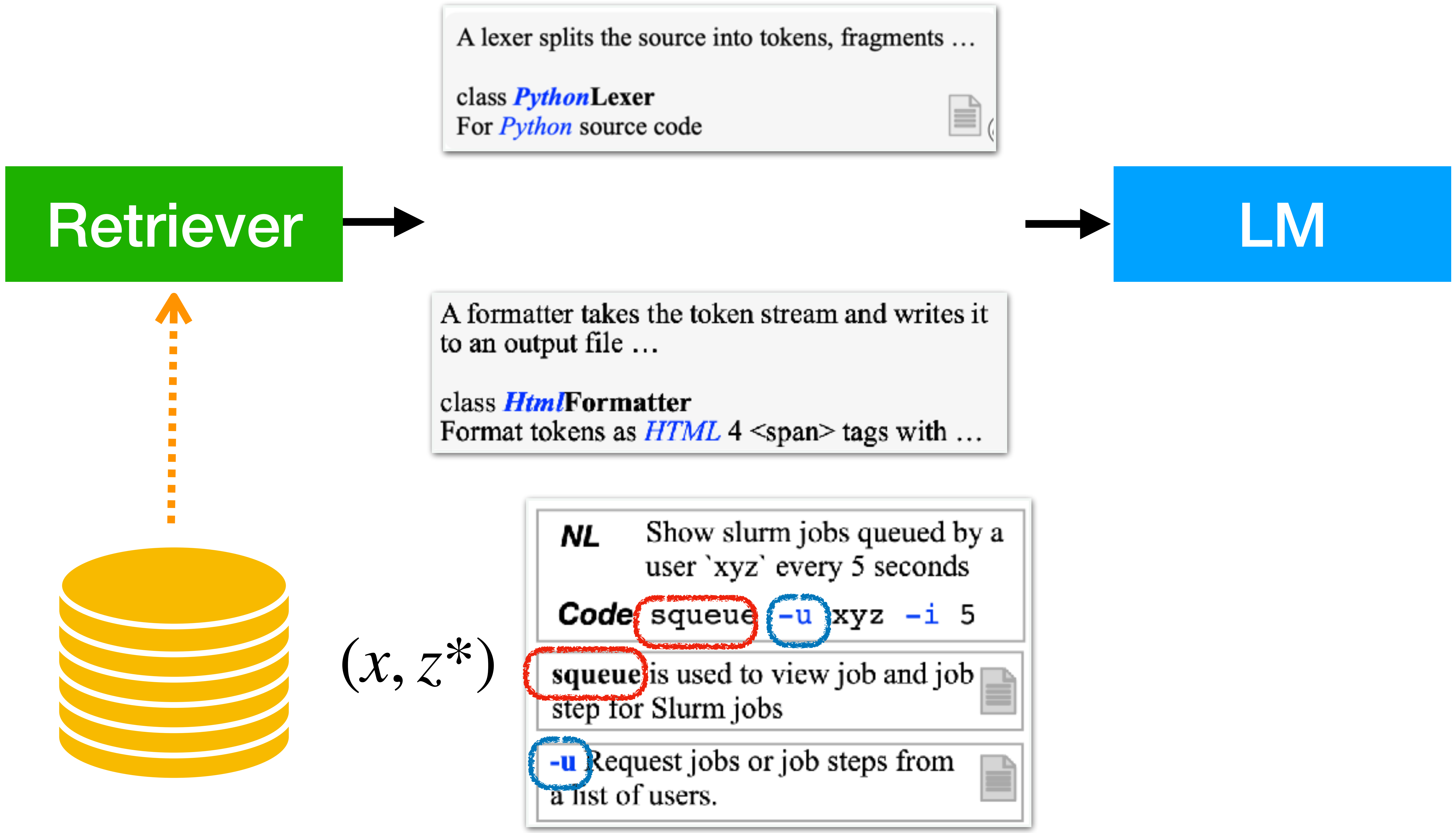
# DocPrompting (Zhou et al., 2023)



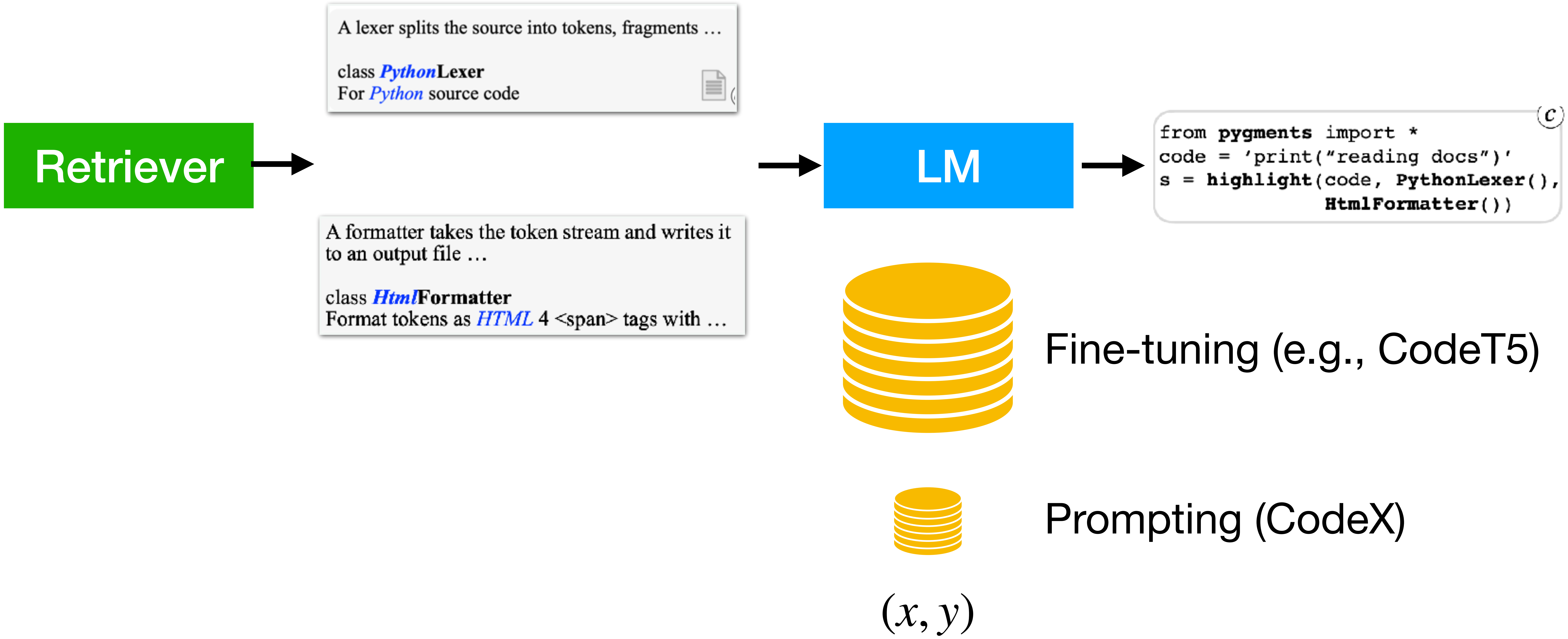

Retrieve **code documentations** about related functions

Zhou et al. 2023. "DocPrompting: Generating the Codes by Retrieving the Docs"

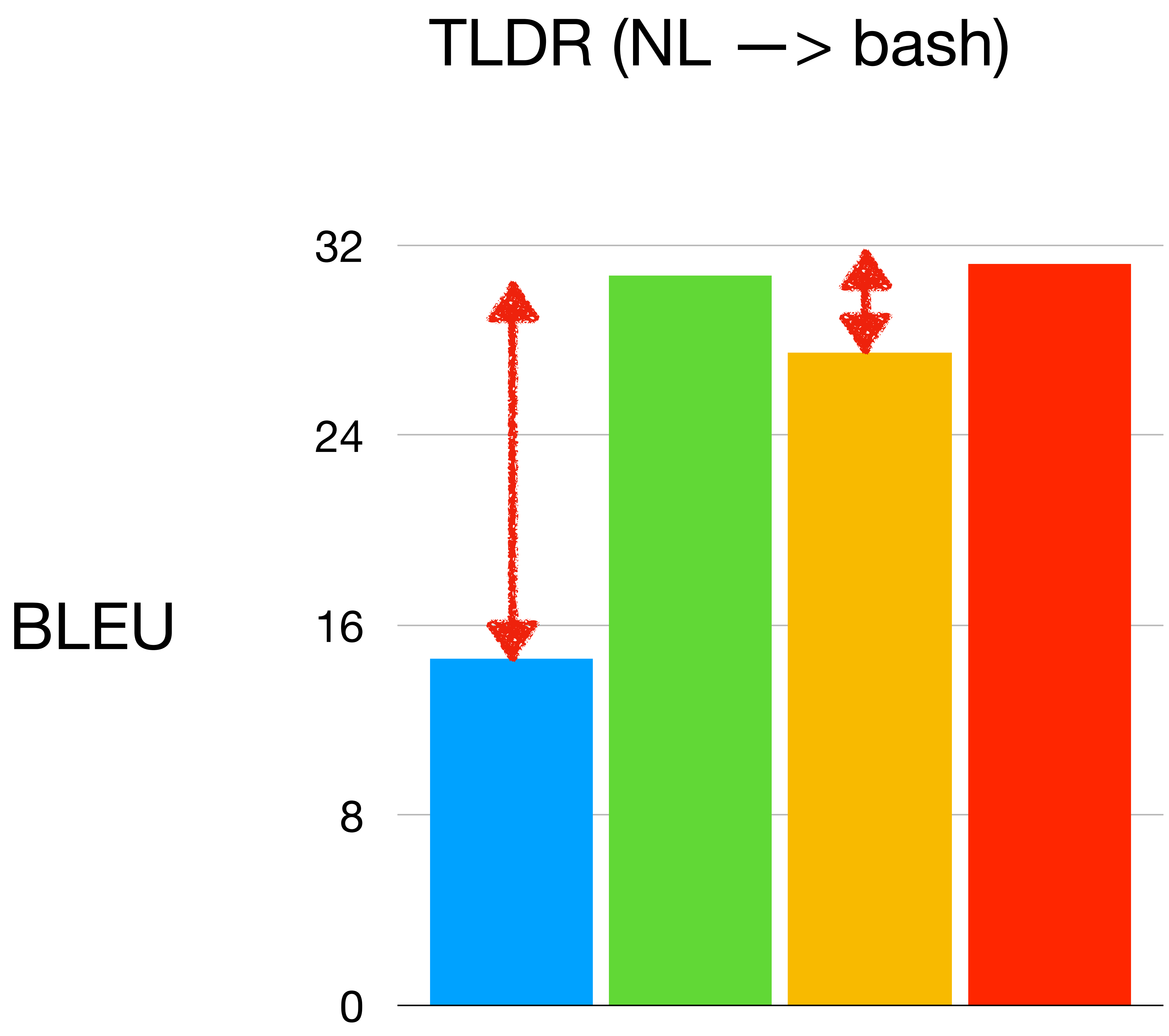
# DocPrompting (Zhou et al., 2023)



# DocPrompting (Zhou et al., 2023)



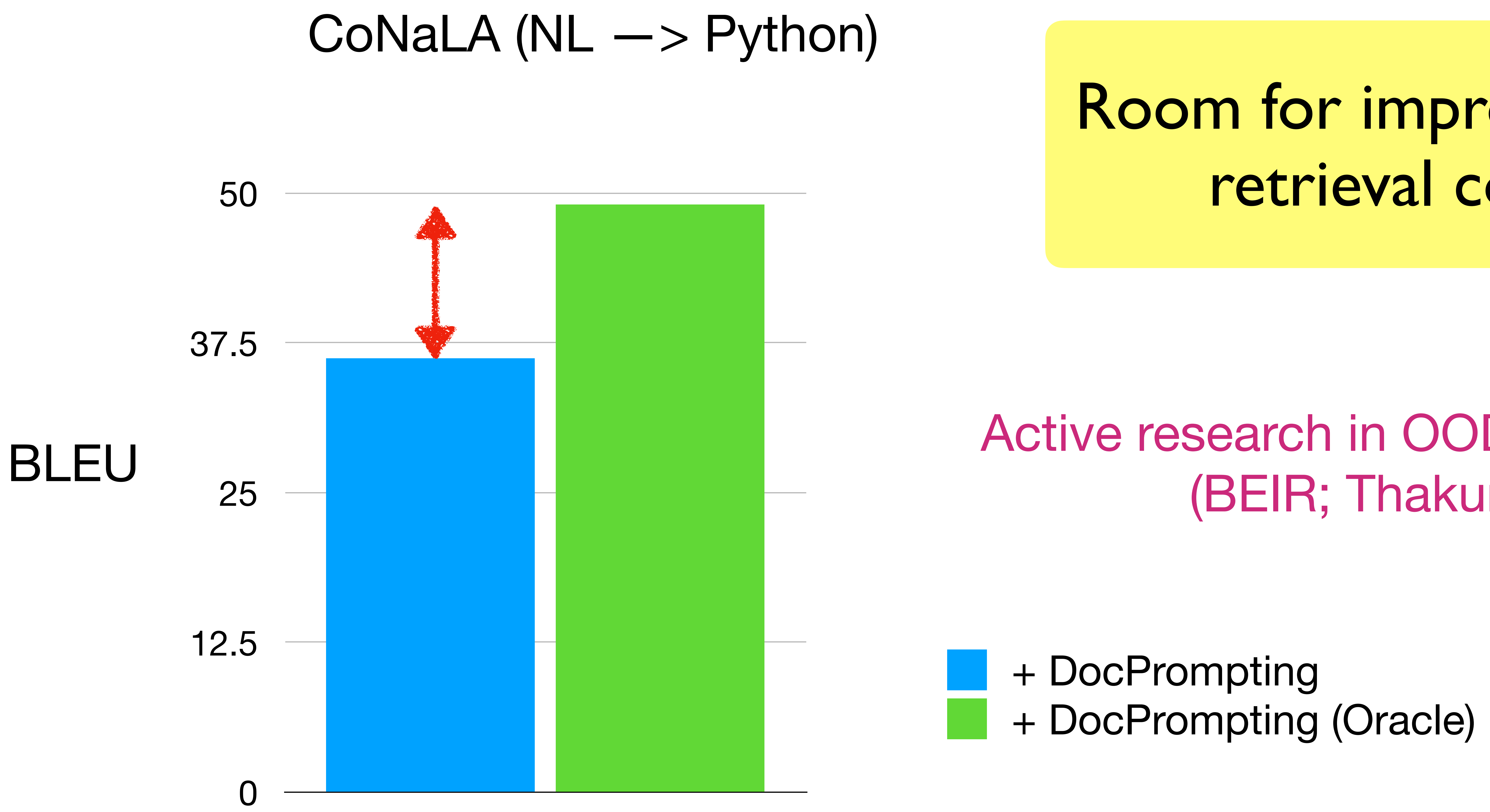
# DocPrompting (Zhou et al., 2023)




CodeT5
+ DocPrompting
CodeX
+ DocPrompting

Large gain given by DocPrompting for both CodeT5 & CodeX

# DocPrompting (Zhou et al., 2023)



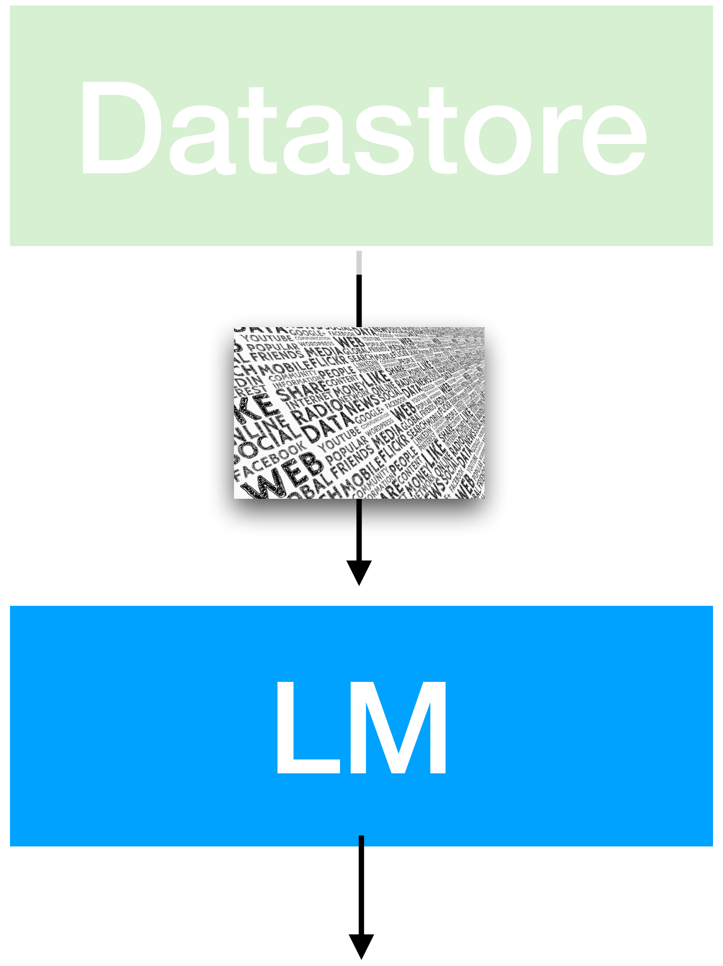

Room for improvement in the retrieval component

Active research in OOD / Zero-shot retrieval!
(BEIR; Thakur et al., 2021)

# Summary of downstream adaptations

| | Target task | Adaptation method | Datastore |
|---|---|---|---|
| ATLAS (Izacard et al., 2022) | Knowledge-intensive | Fine-tuning (Retriever & LM) | Wikipedia \| CC |
| GopherCite (Menick et al., 2022) | Open-domain QA, Long-form QA | Fine-tuning + RL (LM) | Google Search Results |
| kNN-prompt (Shi et al., 2022) | Classification | Prompting (output) | Wikipedia \| CC |
| REPLUG (Shi et al., 2023) | Knowledge-intensive | Prompting (input) | Wikipedia \| CC |
| DocPrompting (Zhou et al., 2023) | Code Generation | Fine-tuning (DS & LM), Prompting (Input) | Code documentations |

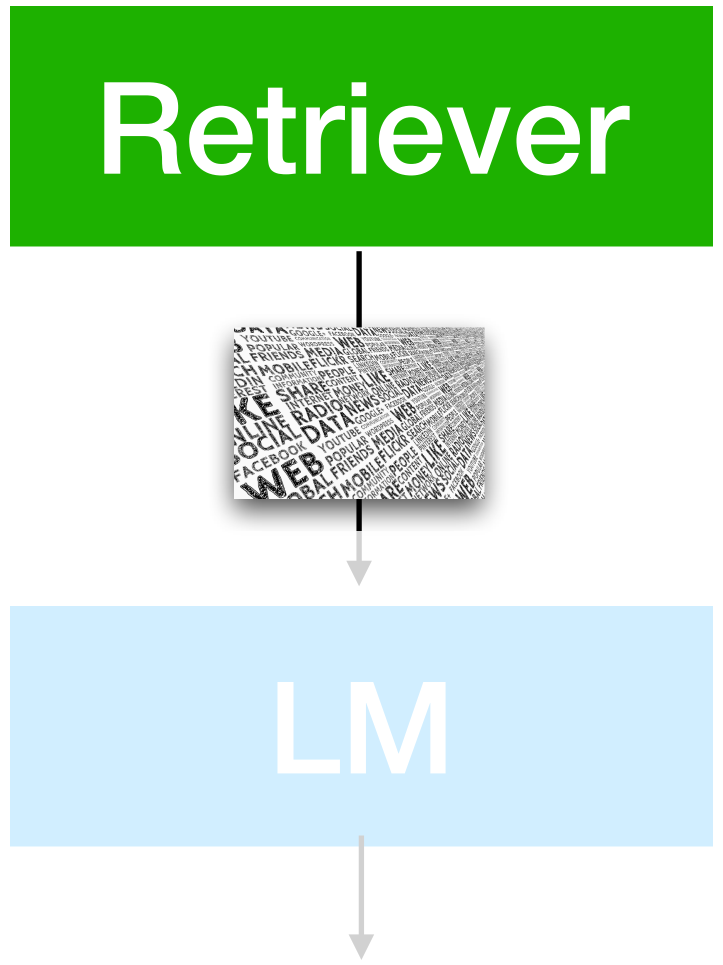
# How to adapt a retrieval-based LM for a task




- Retrieval-based prompting is easy and simple; no need to train but has higher variance

- Fine-tuning (+ RL) requires training but less variance & is completive with more data

# How to adapt a retrieval-based LM for a task




Training a retriever on downstream tasks helps

Datastore can be diverse (also in Section 6) while challenges remain in OOD retrieval

# Two key questions for downstream adaptations

**How** can we adapt a retrieval-based LM for a task?

**When** should we use a retrieval-based LM?

# **When** to use a retrieval-based LM

Long-tail

knowledge update

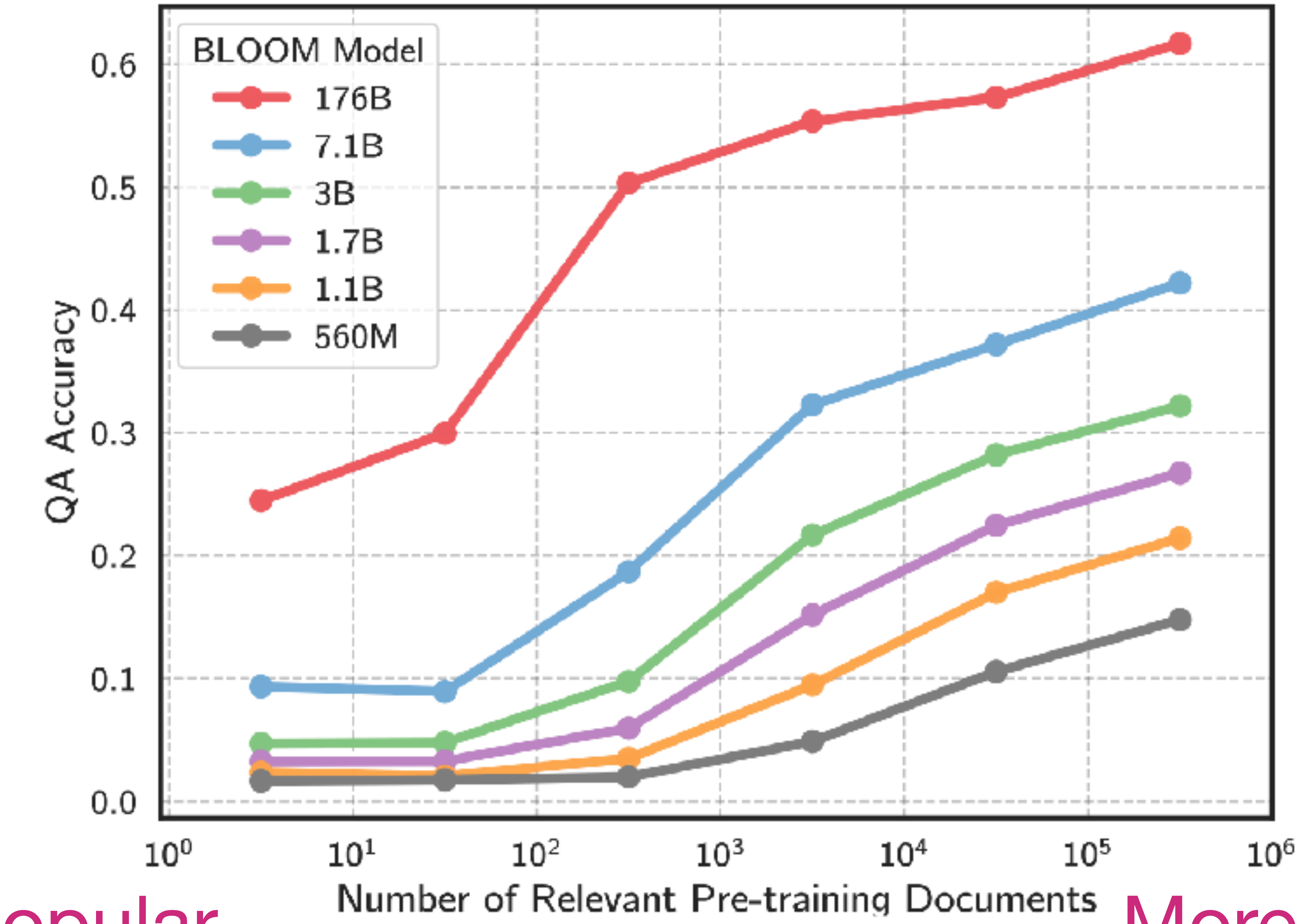
Verifiability

Parameter-efficiency

# Key effectiveness in downstream tasks

**Long-tail**

LLMs often struggle in **long-tail/less frequent entities**




<— less popular

More popular —>

Kandpal et al. 2023. "Large language models struggle to learn long-tail knowledge"

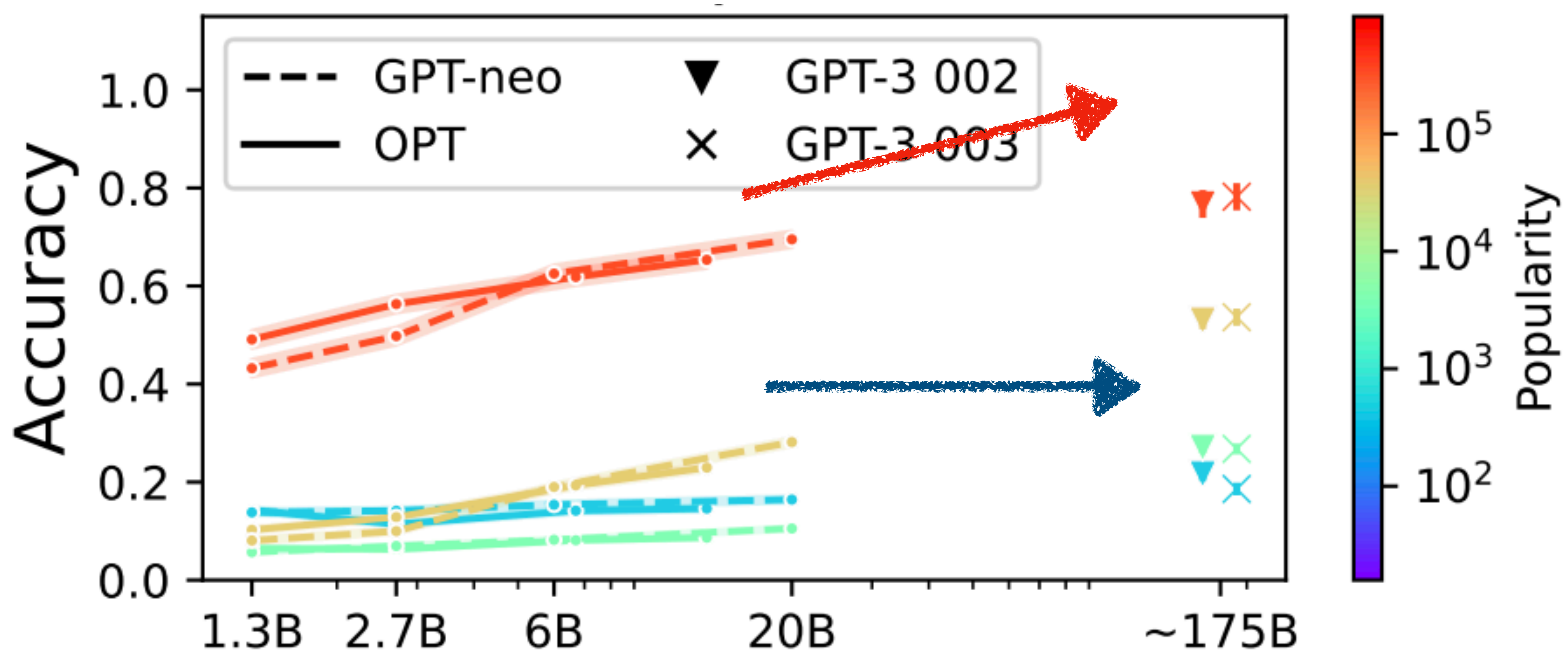
# Key effectiveness in downstream tasks

**Long-tail**

Scaling LLMs only helps for **popular knowledge;** for long tail, scaling gives marginal performance improvements

Performance on less popular questions (blue) doesn't improve over scale

PopQA (Open-domain QA)

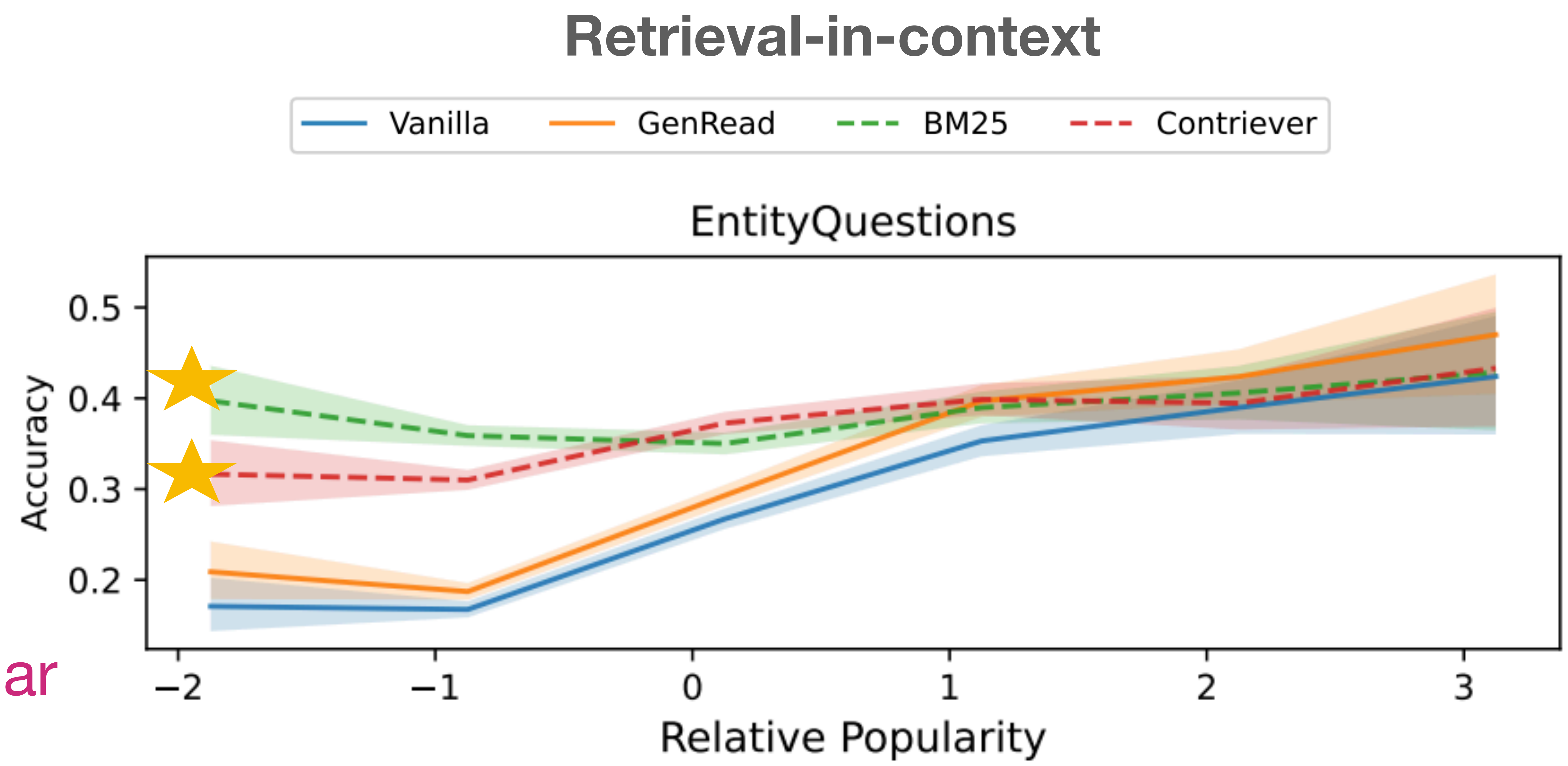



Mallen* and Asai* et al. 2023. "When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories"

# Key effectiveness in downstream tasks

**Long-tail**

Retrieval gives large performance gain in such **long-tail**




<— less popular  More popular —>

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

# Key effectiveness in downstream tasks

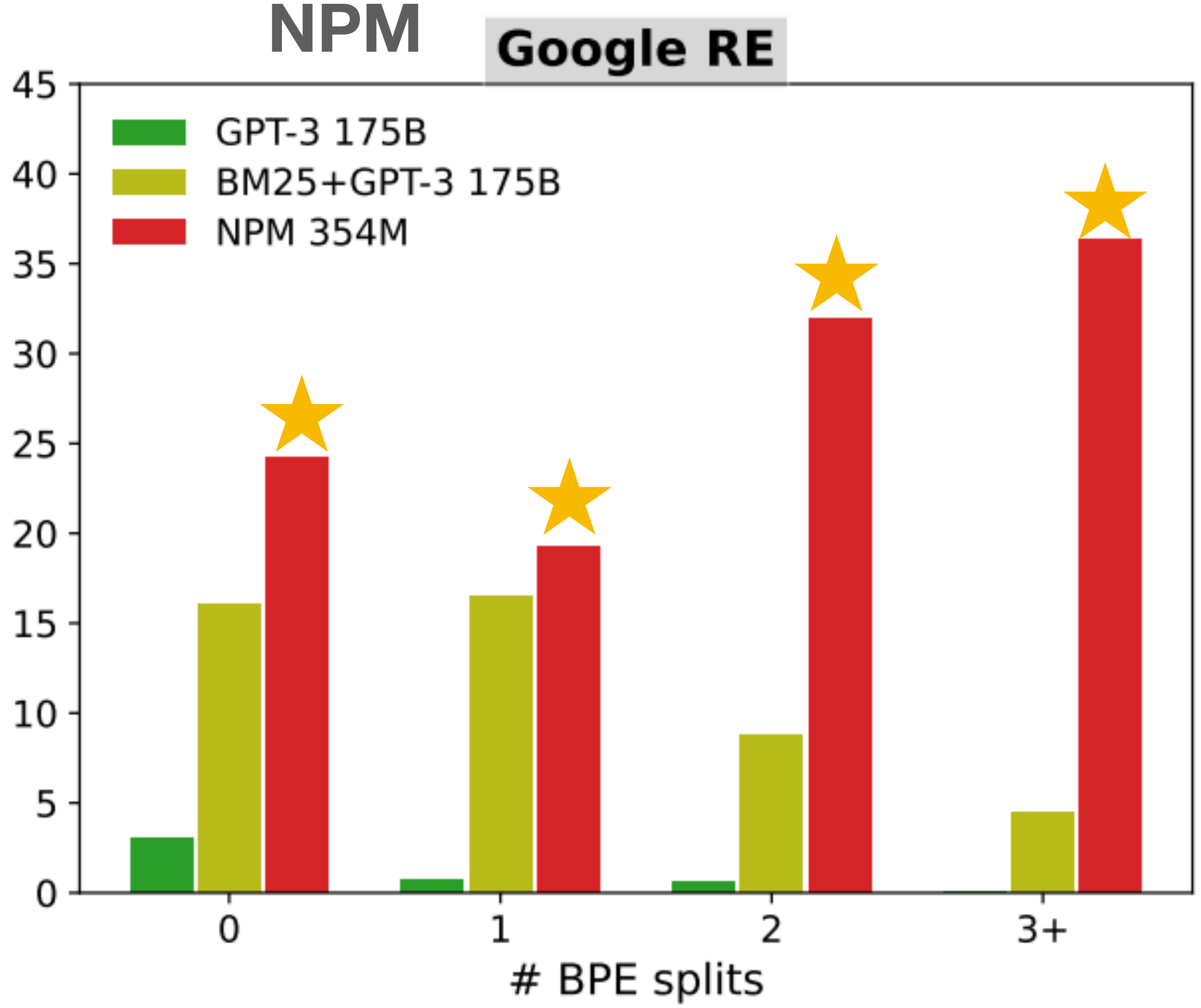
**Long-tail**

Retrieval gives large performance gain in such **long-tail**

Google RE

Joshua Mathiot died in [MASK].

NPM




<— more popular

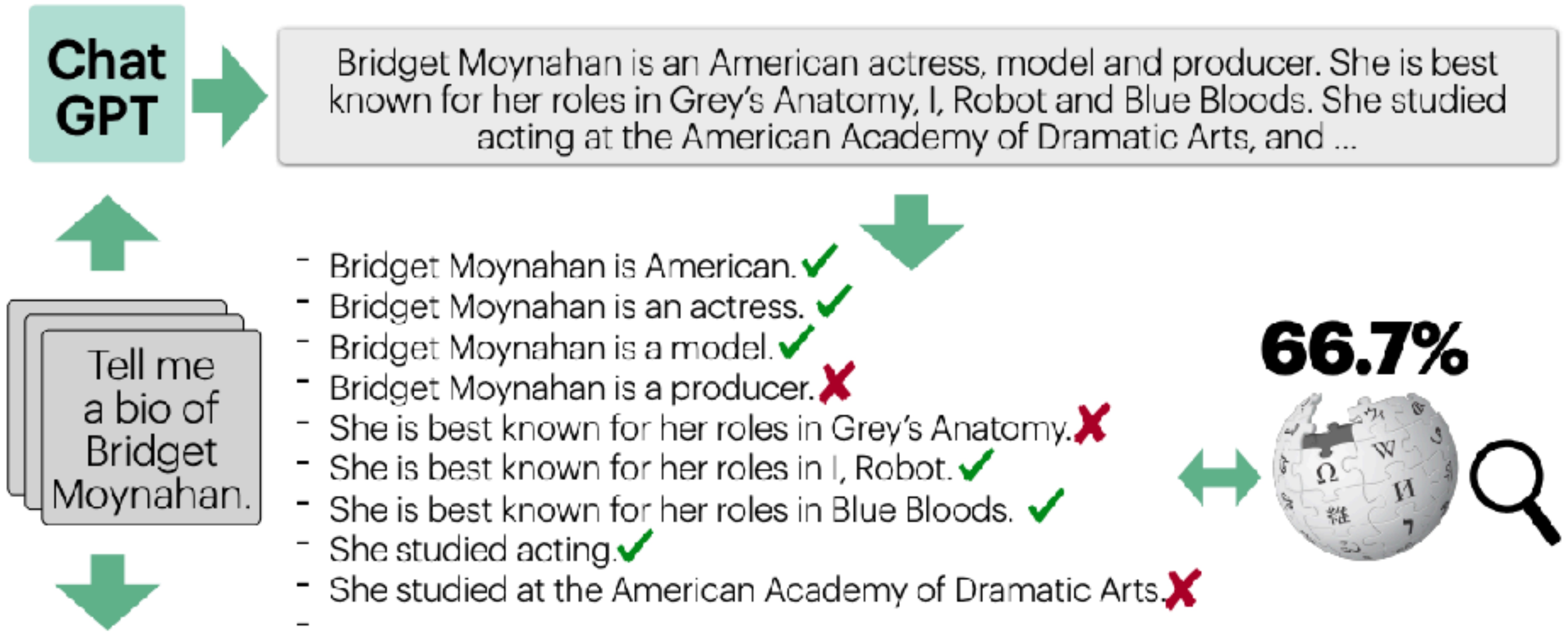
Less popular —>

Min et al. 2023. “Nonparametric Masked Language Modeling” ; Petroni et al. 2019. “Language Models as Knowledge Bases?”

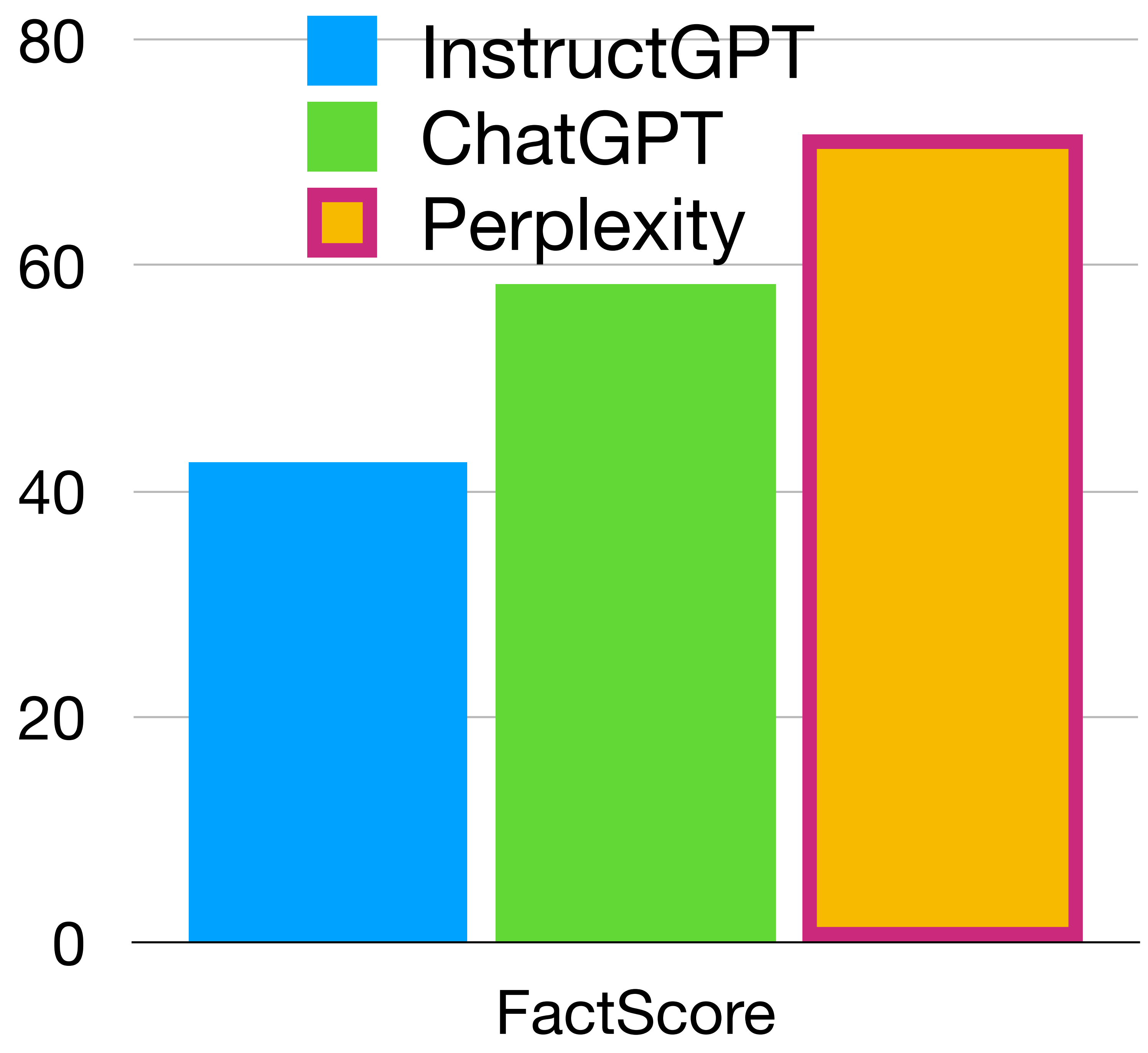
# Key effectiveness in downstream tasks

**Long-tail**

Largely reduce hallucinations in **long-form generations**




FactScore




Min et al. 2023. "FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation"

# Key effectiveness in downstream tasks

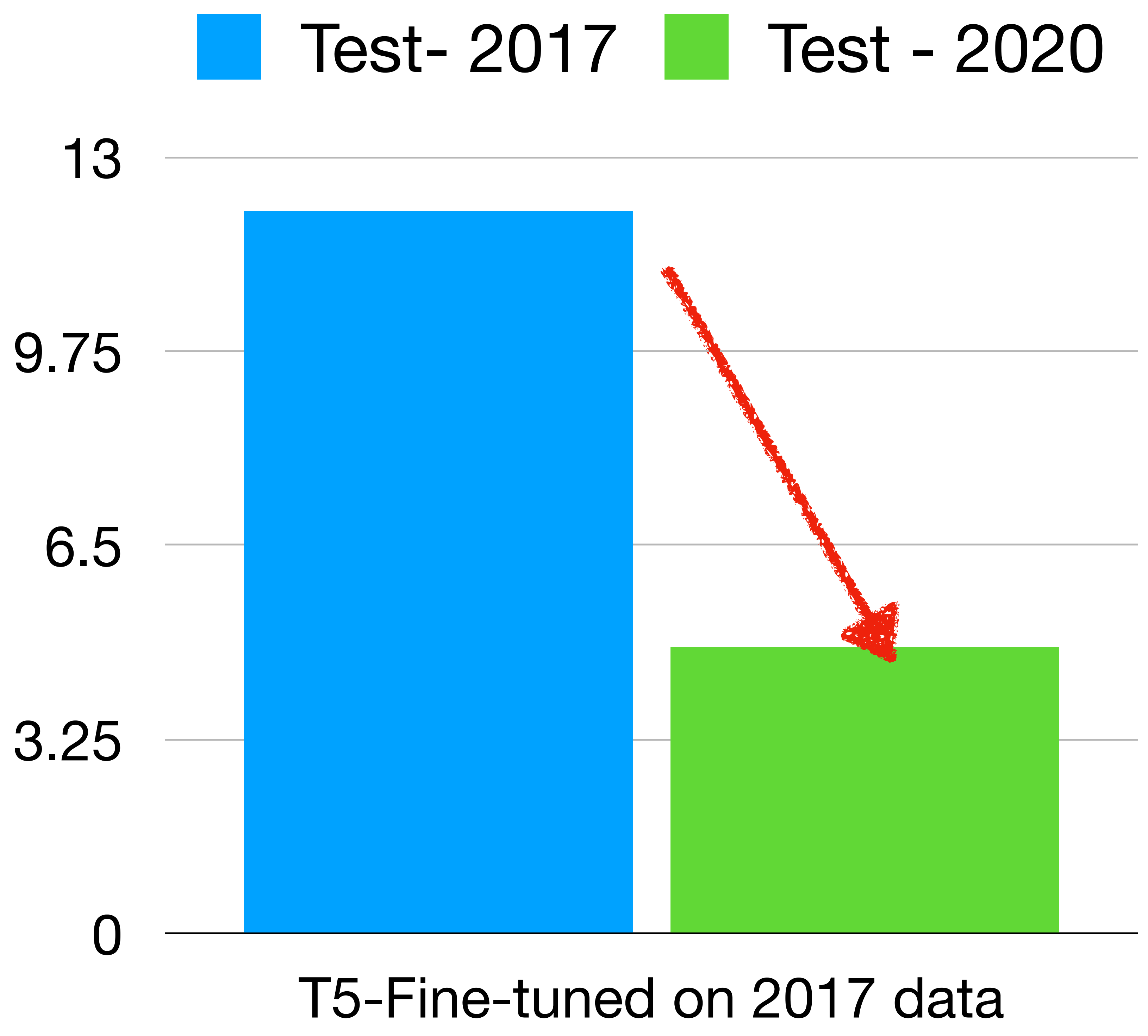
**Update**

Standard LLMs need to be **trained again** to adapt to evolving world knowledge

Temp LAMA

| | | |
|---|---|---|
| 2012 | Cristiano Ronaldo plays for _X_. | Real Madrid |
| 2019 | Cristiano Ronaldo plays for _X_. | Juventus FC |

Huge performance drop when test knowledge needs to be updated




Izacard et al. 2022. "Few-shot learning with retrieval augmented language models"
Dhingra et al. 2022. "Time-Aware language models as temporal knowledge bases"

# Key effectiveness in downstream tasks

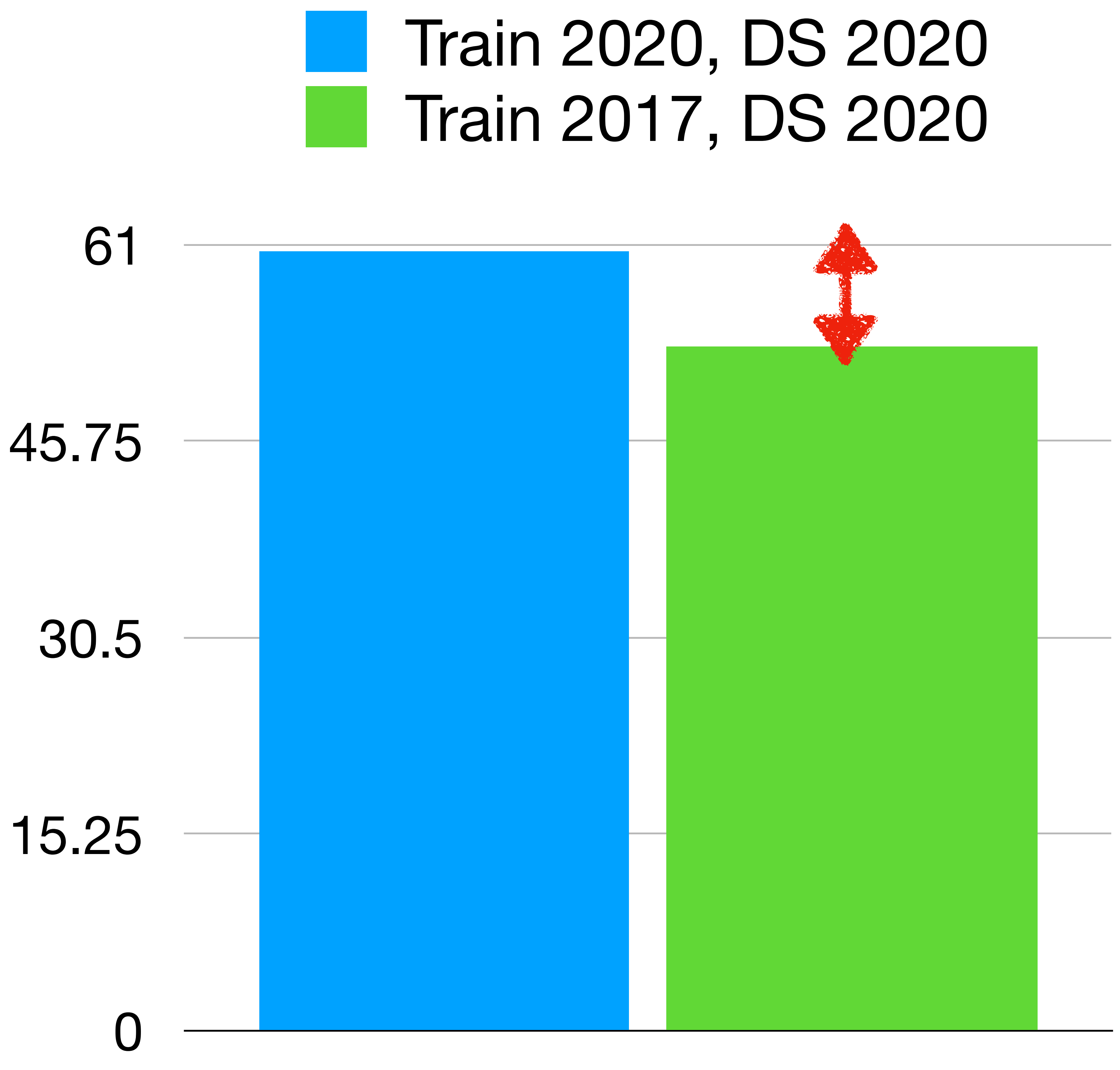
Update

Swapping the knowledge corpus to **accommodate temporal changes** without additional training.

Temp LAMA

| | | |
|---|---|---|
| 2012 | Cristiano Ronaldo plays for _X_. | Real Madrid |
| 2019 | Cristiano Ronaldo plays for _X_. | Juventus FC |

Swapping test datastore only retains strong performance



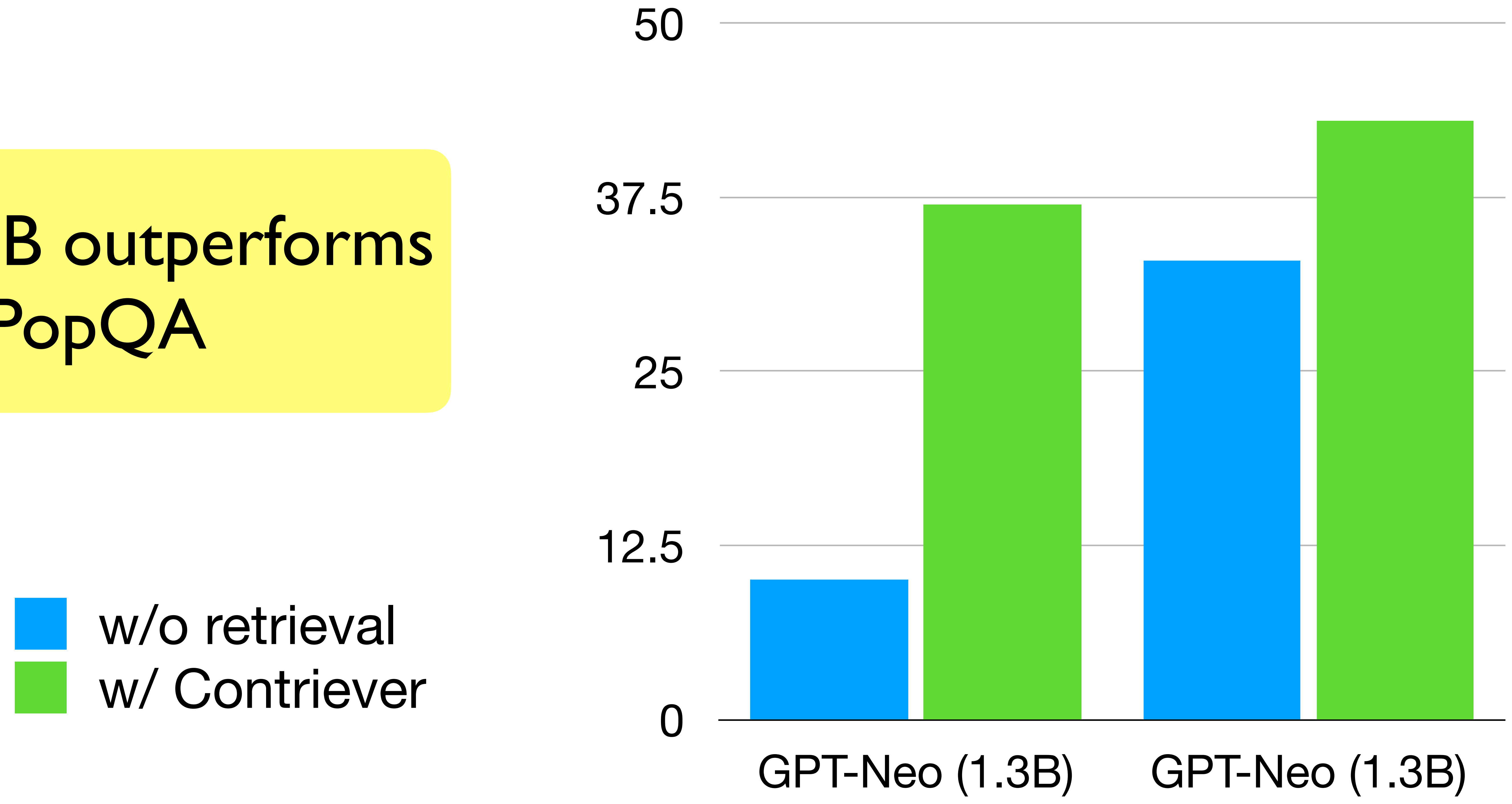

Izacard et al. 2022. "Few-shot learning with retrieval augmented language models"
Dhingra et al. 2022. "Time-Aware language models as temporal knowledge bases"

# Key effectiveness in downstream tasks

Parameter-efficiency

**Much smaller LMs with retrieval** can outperform much larger LMs in fact completions.

Retrieval + GPT-Neo 1.3B outperforms vanilla GPT3 on PopQA



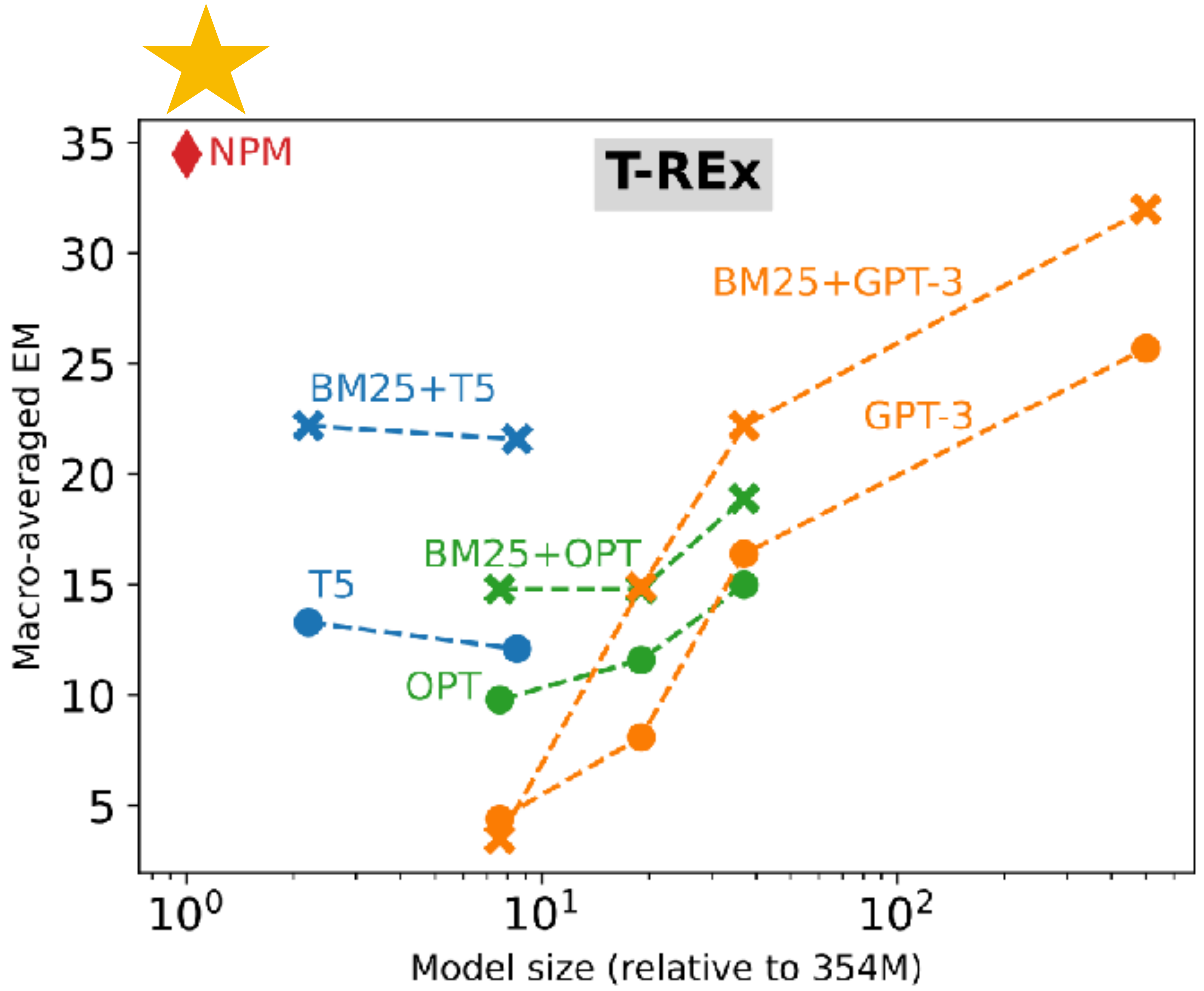

Mallen* and Asai* et al. 2023. “When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories”

# Key effectiveness in downstream tasks

Parameter-efficiency

**Much smaller LMs with retrieval** can outperform much larger LMs in fact completions.

NPM (354 M) outperforms GPT-3 on T-Rex.



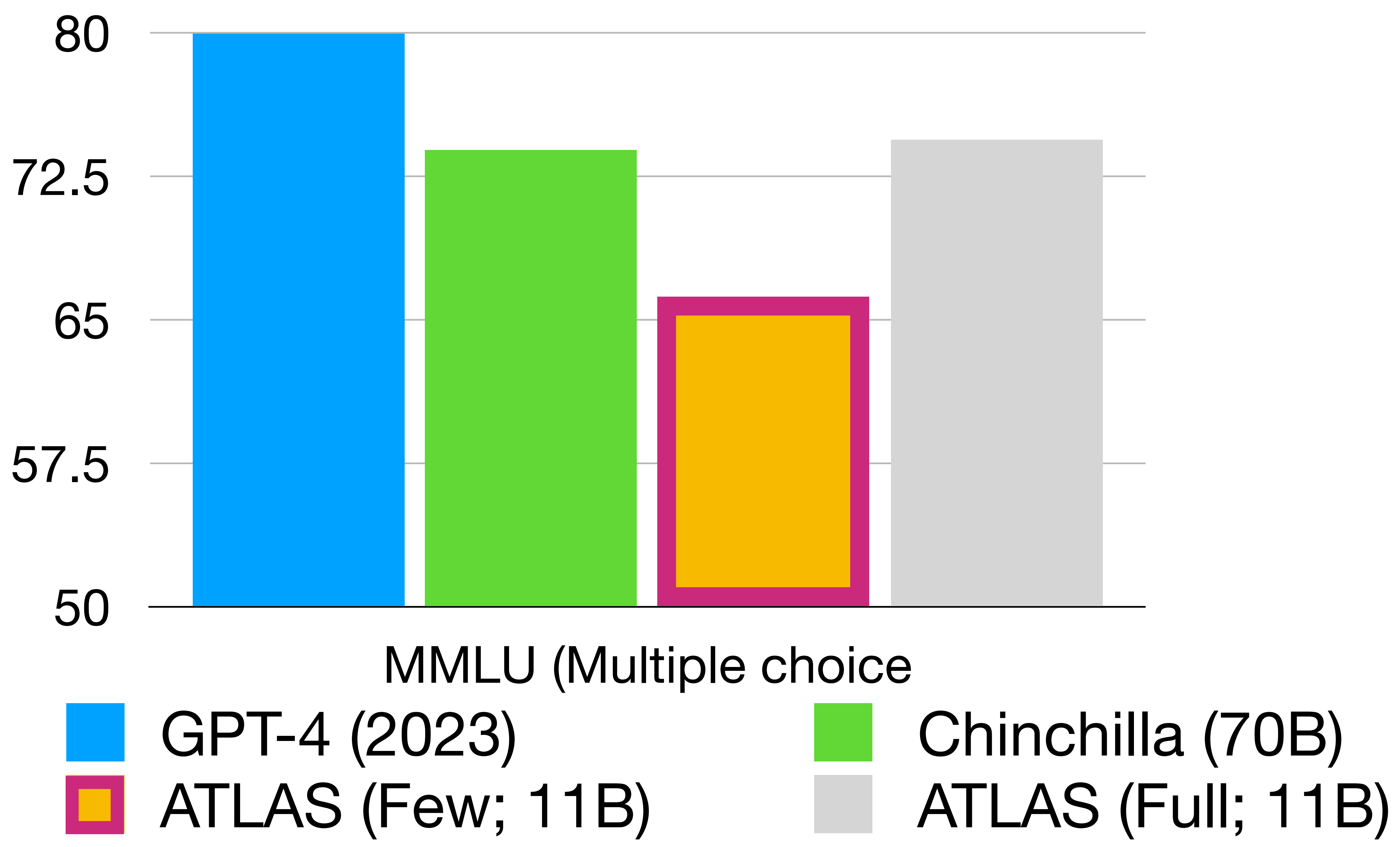

Min et al. 2023. “Nonparametric Masked Language Modeling”
Petroni et al. 2019. “Language Models as Knowledge Bases?”

# Key effectiveness in downstream tasks

Parameter-efficiency

**Much smaller LMs with retrieval** can outperform much larger LMs in fact completions.

Room for improvements for diverse task adaptations!

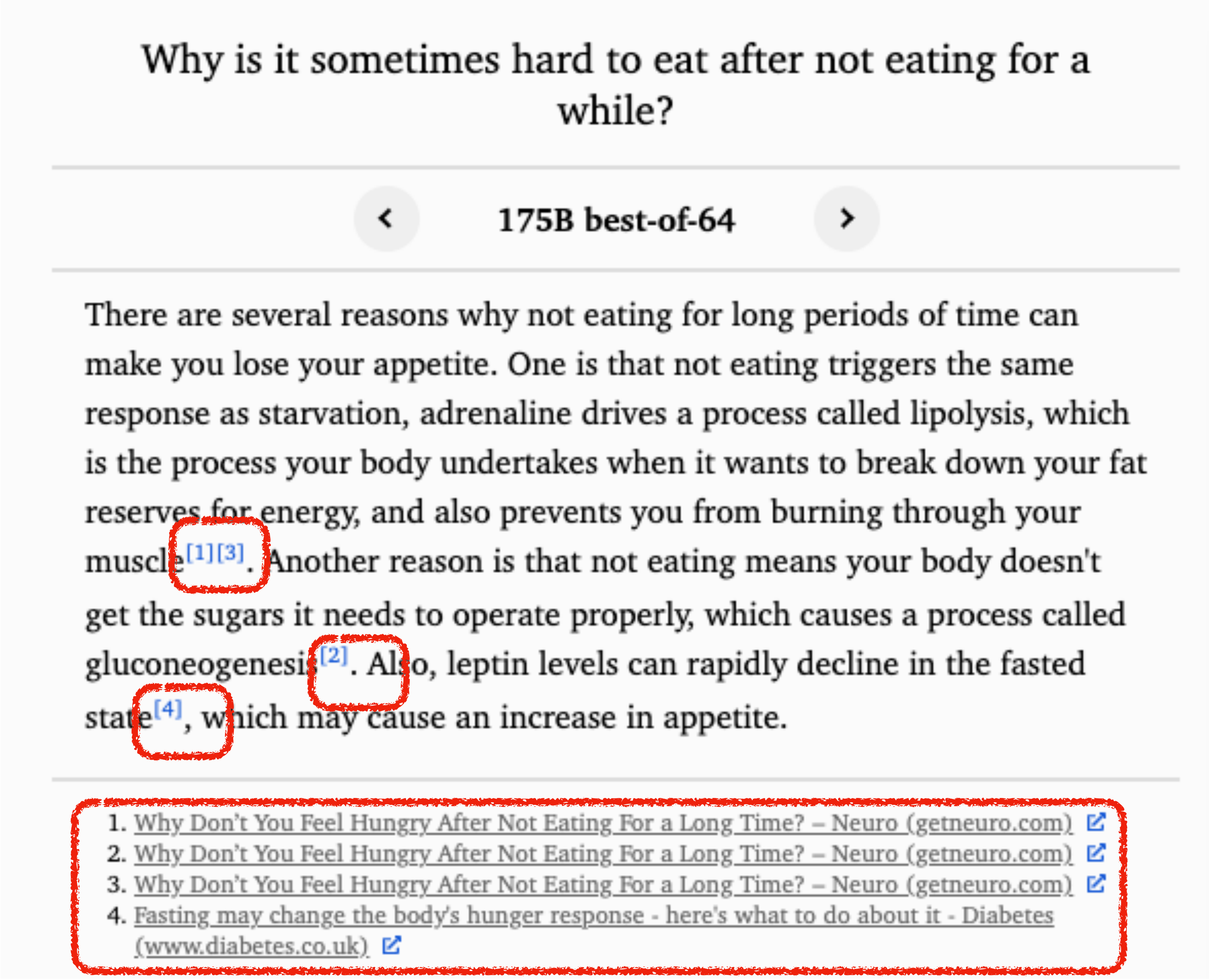



Izacard et al. 2022. "Few-shot learning with retrieval augmented language models"

# Key effectiveness in downstream tasks

**Verifiability**

Human and model can reliably assess the **factuality of the generations** using the retrieved evidence.




Nakano et al. 2021. "WebGPT: Browser-assisted question-answering with human feedback"

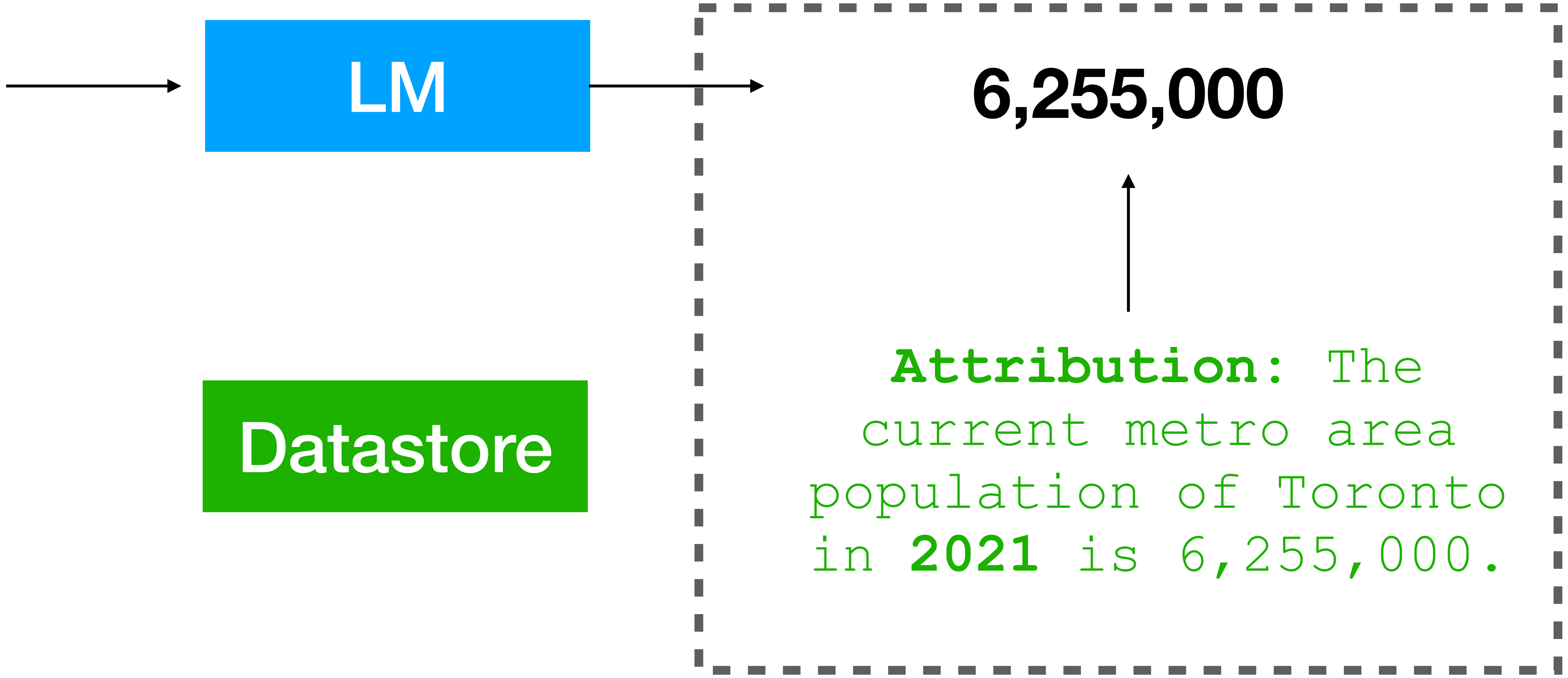
# Attributions: AttributedQA (Bohnet et al., 2022)

Expected Model Output

$$g(x) = (a, c)$$

**Q:** The population of Toronto is _

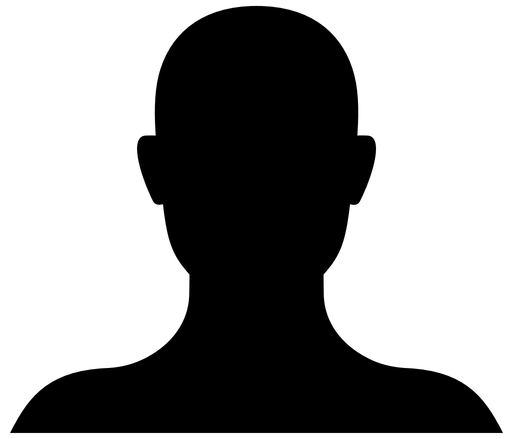
**Answer:**




Bohnet et al. 2022. “Attributed Question Answering: Evaluation and Modeling for Attributed Large Language Models”

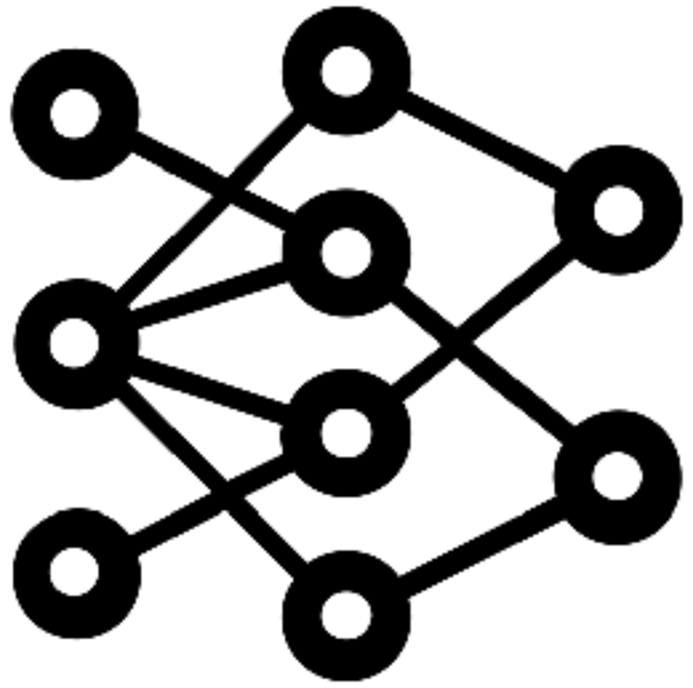
# Attributions: AttributedQA (Bohnet et al., 2022)

**Human Evaluation (AIS)**



1. Are all (a,c) interpretable?
2. Is any information in a supported by c?

**Automatic Evaluation (AutoAIS)**

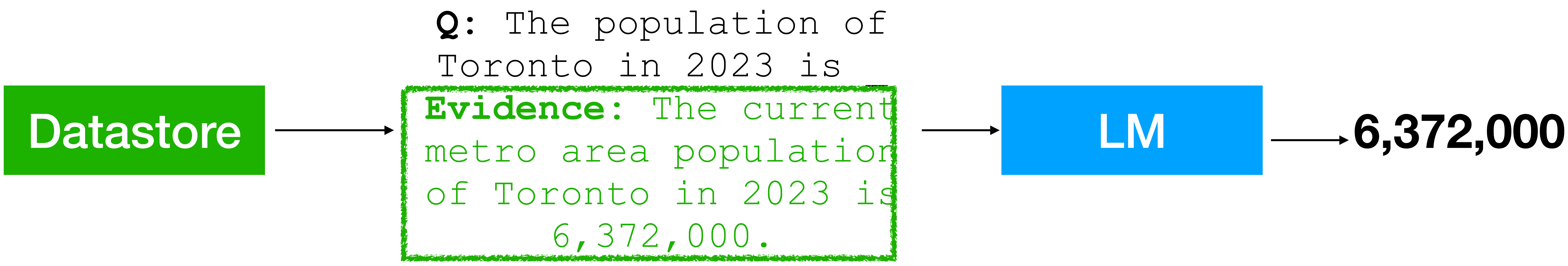


NLI model

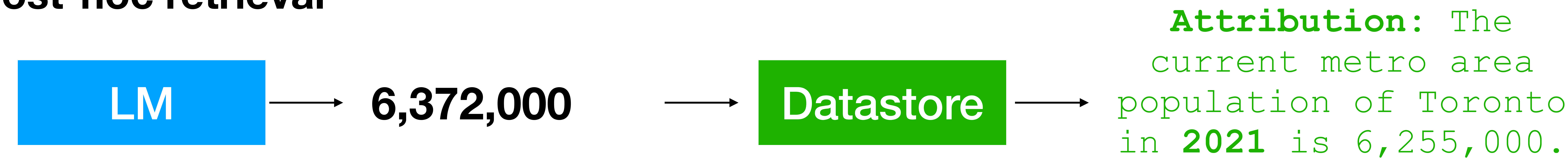
$$E^{\mathsf{A}}[g] = \frac{1}{n}\sum_{i=1}^{n} \texttt{AutoAIS}(x_i, g(x_i))$$

# AttributedQA (Bohnet et al., 2022)

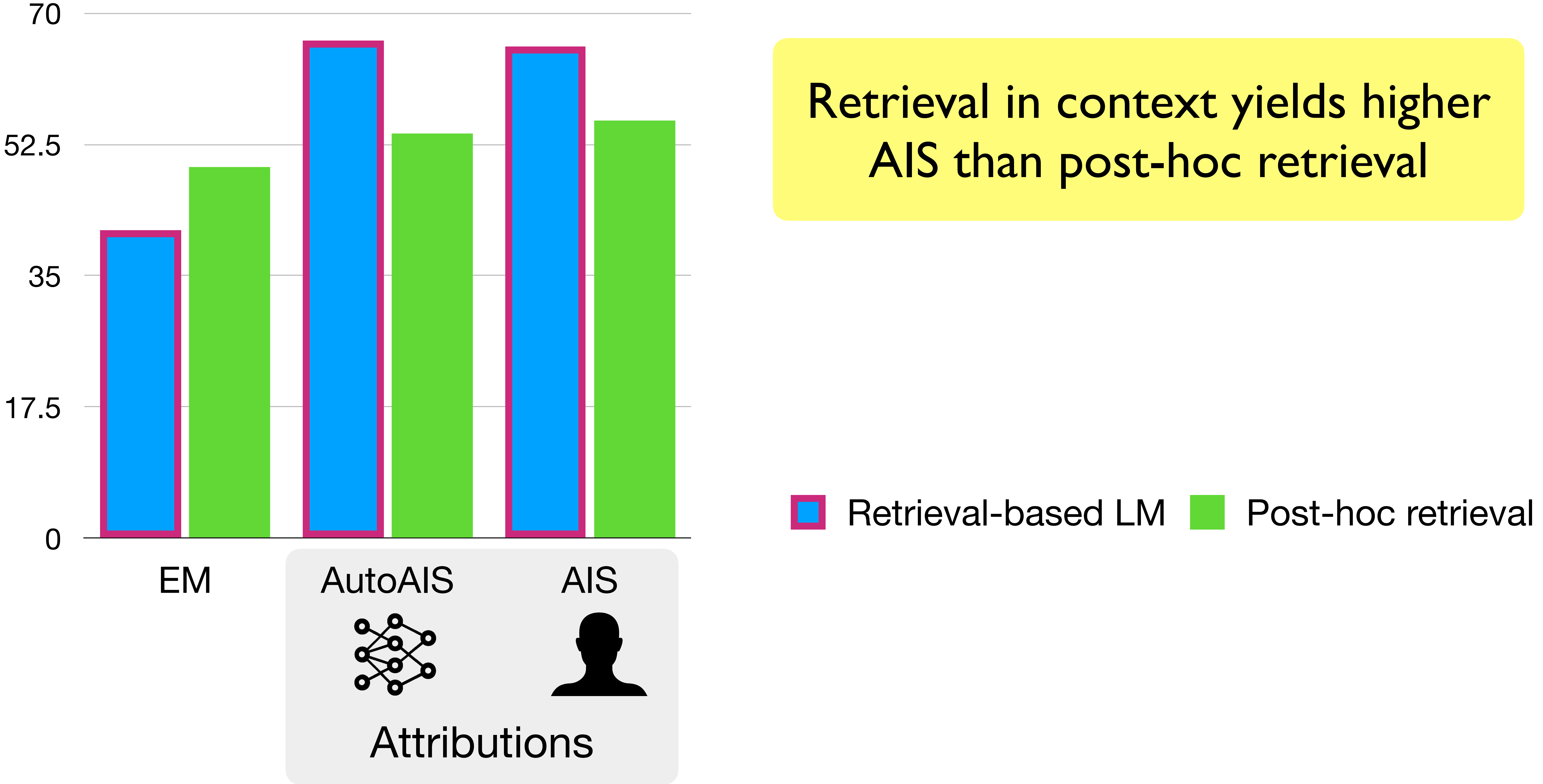
## Retrieval-based LM




## Post-hoc retrieval



# AttributedQA (Bohnet et al., 2022)




Retrieval in context yields higher AIS than post-hoc retrieval

# **When** to use a retrieval-based LM

Long-tail

knowledge update

Verifiability

Parameter-efficiency

# **When** to use a retrieval-based LM

Long-tail

knowledge update

Verifiability

Parameter-efficiency

Out of domain adaptations
(Shi et al., 2022; Zheng et al., 2021)

Khandelwal, et al.2020. "Nearest Neighbor Zero-shot Inference"

Shi et al. 2022. "Nearest Neighbor Zero-shot Inference"

# **When** to use a retrieval-based LM

| | | | |
|---|---|---|---|
| Long-tail | knowledge update | Verifiability | Parameter-efficiency |

| | |
|---|---|
| Out of domain adaptations (Shi et al., 2022; Zheng et al., 2021) | Privacy (Huang et al., 2023) |

Khandelwal, et al.2020. “Nearest Neighbor Zero-shot Inference”

Shi et al. 2022. “Nearest Neighbor Zero-shot Inference”

Huang et al. 2023. “Privacy Implications of Retrieval-Based Language Models”