CS11-711 Advanced NLP

Retrieval and RAG

Fernando Diaz

Carnegie Mellon University



Slides adapted from Graham Neubig's Fall 2024 course and Akari Asai's Spring 2026 lecture.

Limitations of LMs: Hallucinations

Who is Akari Asai?

(GPT 5.1, 2026/01/20)

Akari Asai is a prominent researcher in the field of **Natural Language Processing (NLP)** and **Artificial Intelligence (AI)**. She is particularly known for her work on **open-domain question answering**, **retrieval-augmented generation**, and **large language models**. Her research focuses on enabling machines to retrieve, understand, and reason over large corpora of unstructured text — a critical area for building systems like ChatGPT and other advanced NLP applications.

📄 Representative Papers by Akari Asai

Here are **five representative papers** by Akari Asai, with their **publication venues**:

1. **~~"Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks"~~**
   Venue: Proceedings of NeurIPS 2020 (Conference on Neural Information Processing Systems)
2. **~~"Learning to Retrieve Passages without Supervision"~~**
   Venue: Proceedings of ACL 2021 (Annual Meeting of the Association for Computational Linguistics)
3. **~~"Task-Aware Query Reformulation with Reinforcement Learning for Open-Domain Question~~ Answering"**
   Venue: Proceedings of EMNLP 2019 (Conference on Empirical Methods in Natural Language Processing)

Paper exists but not by Akari

Paper exists but not by Akari

Does not exist

Limitations of LMs: Hallucinations



ICLR

https://gptzero.me/news/iclr-2026/

Title	Average Review Rating	Paper Link	Citation Check Scan Link	Example of Verified Hallucination	Comment
TamperTok: Forensics-Driven Tokenized Autoregressive Framework for Image Tampering Localization	8.0	TamperTok: Forensics-Driven Tokenized Autoregressive Framework for Image Tampering Localization | OpenReview	https://app.gptzero.me/documents/4645494f-70eb-40bb-aea7-0007e13f7179/share	Chong Zou, Zhipeng Wang, Ziyu Li, Nan Wu, Yuling Cai, Shan Shi, Jiawei Wei, Xia Sun, Jian Wang, and Yizhou Wang. Segment everything everywhere all at once. In Advances in Neural Information Processing Systems (NeurIPS), volume 36, 2023.	This paper exists, but all authors are wrong.
MixtureVitae: Open Web-Scale Pretraining Dataset With High Quality Instruction and Reasoning Data Built from Permissive Text Sources	8.0	MixtureVitae: Open Web-Scale Pretraining Dataset With High Quality Instruction and Reasoning Data Built from Permissive Text Sources | OpenReview	https://app.gptzero.me/documents/bfd10666-ea2d-454c-9ab2-75faa8b84281/share	Dan Hendrycks, Collin Burns, Steven Basart, Andy Critch, Jerry Li, Dawn Ippolito, Aina Lapedriza, Florian Tramer, Rylan Macfarlane, Eric Jiang, et al. Measuring massive multitask language understanding. In Proceedings of the International Conference on Learning Representations (ICLR), 2021.	The paper and first 3 authors match. The last 7 authors are not on the paper, and some of them do not exist
Catch-Only-One: Non-Transferable Examples for Model-Specific Authorization	6.0	Catch-Only-One: Non-Transferable Examples for Model-Specific Authorization | OpenReview	https://app.gptzero.me/documents/9afb1d51-c5c8-48f2-9b75-250d95062521/share	Dinghuai Zhang, Yang Song, Inderjit Dhillon, and Eric Xing. Defense against adversarial attacks using spectral regularization. In International Conference on Learning Representations (ICLR), 2020.	No Match

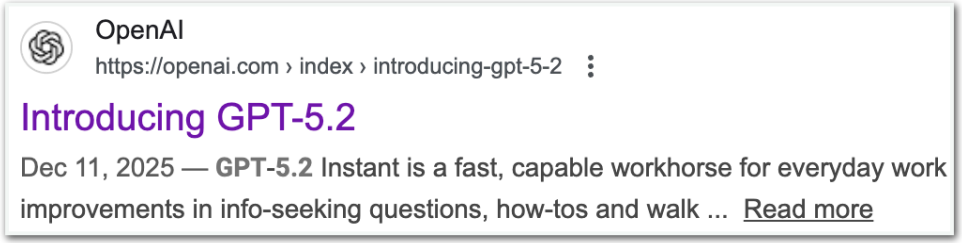
Limitations of LMs: Obsolete knowledge

When did GPT 5.2 come out?

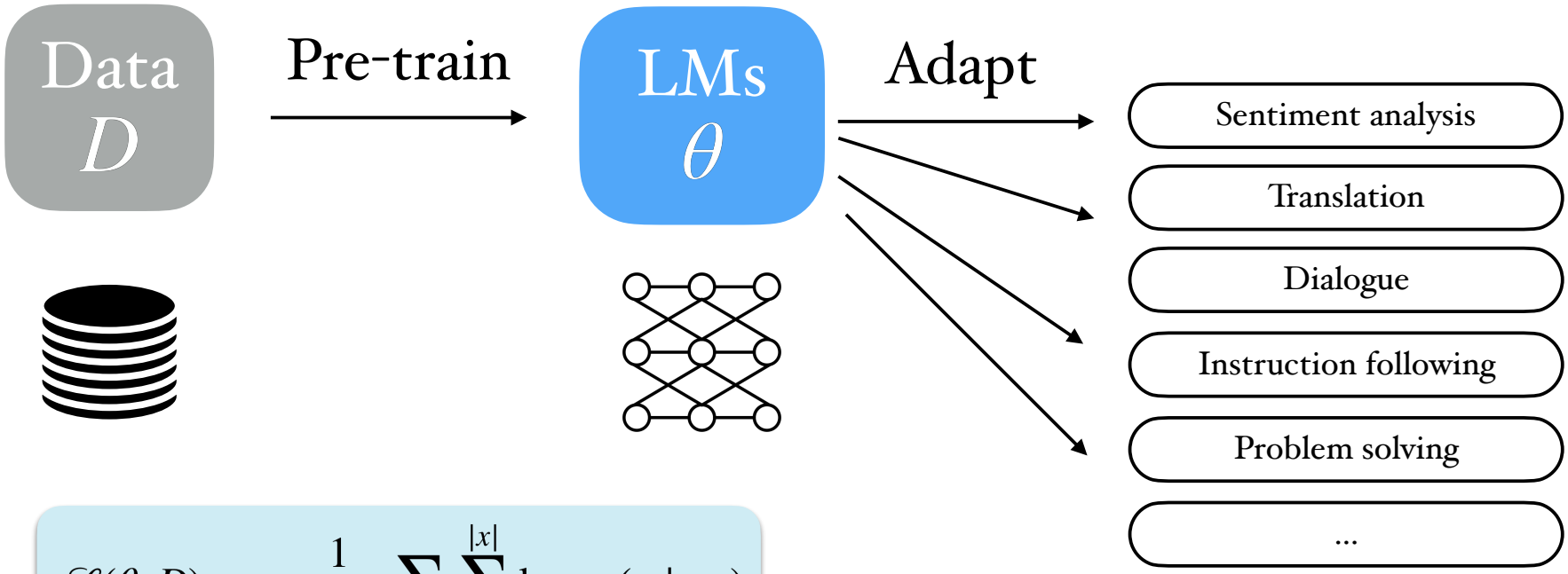
(GPT 5.2, 2026/01/20)

There's **no confirmed public release date for "GPT-5.2"** in the information I was trained on.

As of mid-2024, OpenAI hadn't officially announced a version named GPT-5.2, so I don't have a specific release date for it. If it was released after that, the exact date isn't in my training.



Limitations of Monolithic LMs



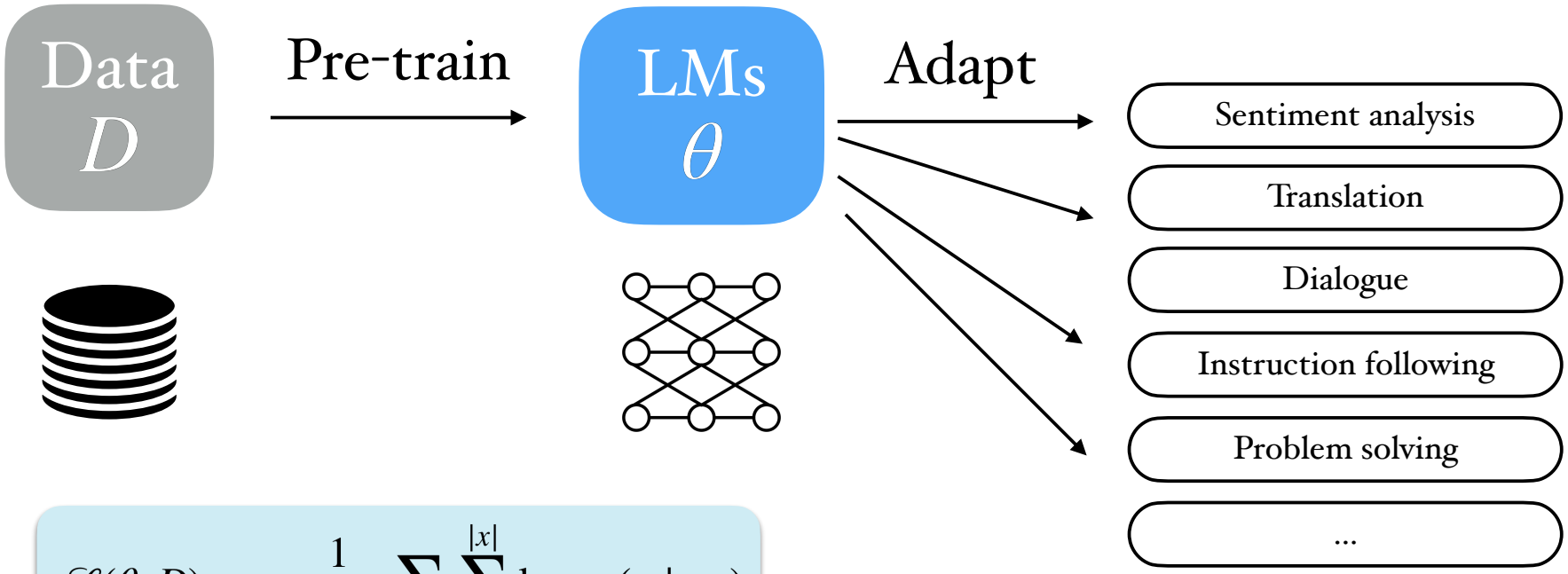
$$\mathcal{L}(\theta; D) = -\frac{1}{|D|} \sum_{x \in D} \sum_{t=1}^{|x|} \log p_\theta(x_t \mid x_{<t})$$

θ can't store infinite information

Allen-Zhu et al 2024. Physics of Language Models: Part 3.3, Knowledge Capacity Scaling Laws.

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

Limitations of Monolithic LMs



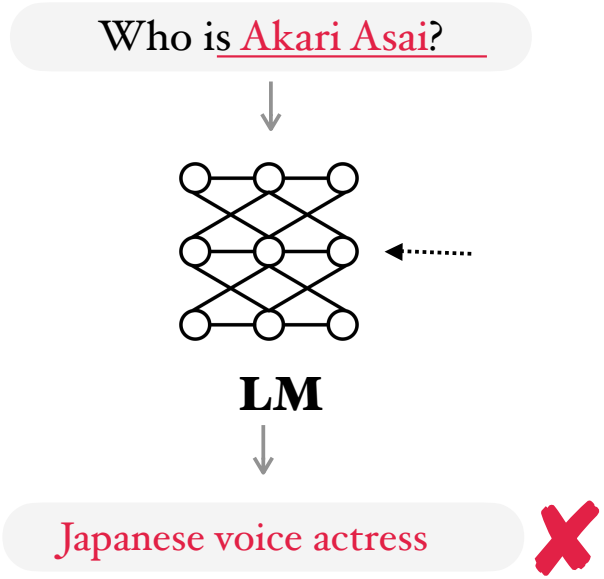
$$\mathcal{L}(\theta; D) = -\frac{1}{|D|}\sum_{x \in D}\sum_{t=1}^{|x|}\log p_\theta(x_t \mid x_{<t})$$

θ can't store infinite information

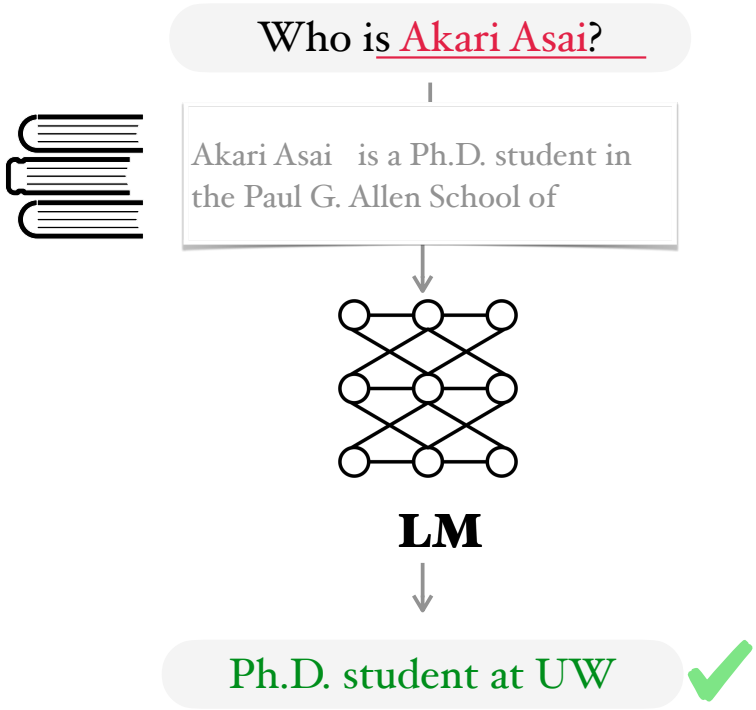
D only capture information as of the data cutoff time

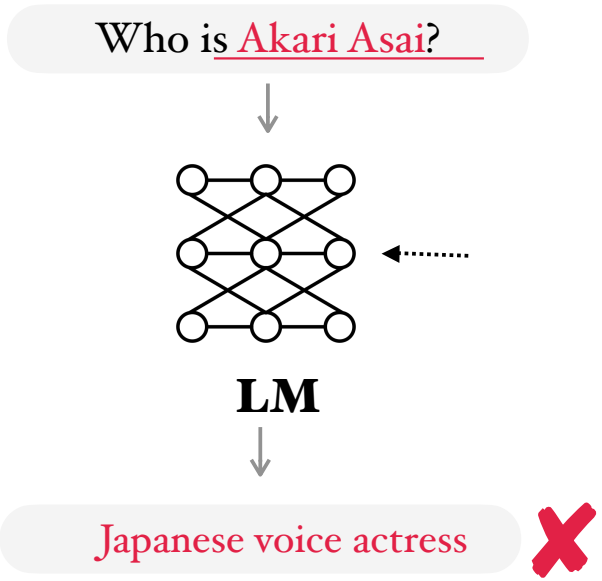
Kasai et al 2023. RealTime QA: What's the Answer Right Now?

Retrieval-Augmented LMs: Intuitions



Retrieval-Augmented LMs: Intuitions

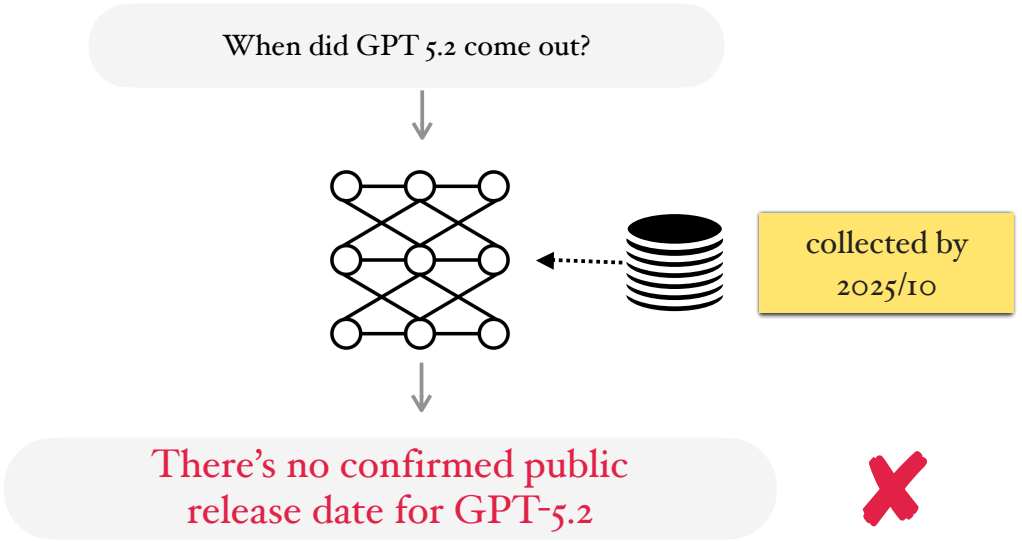




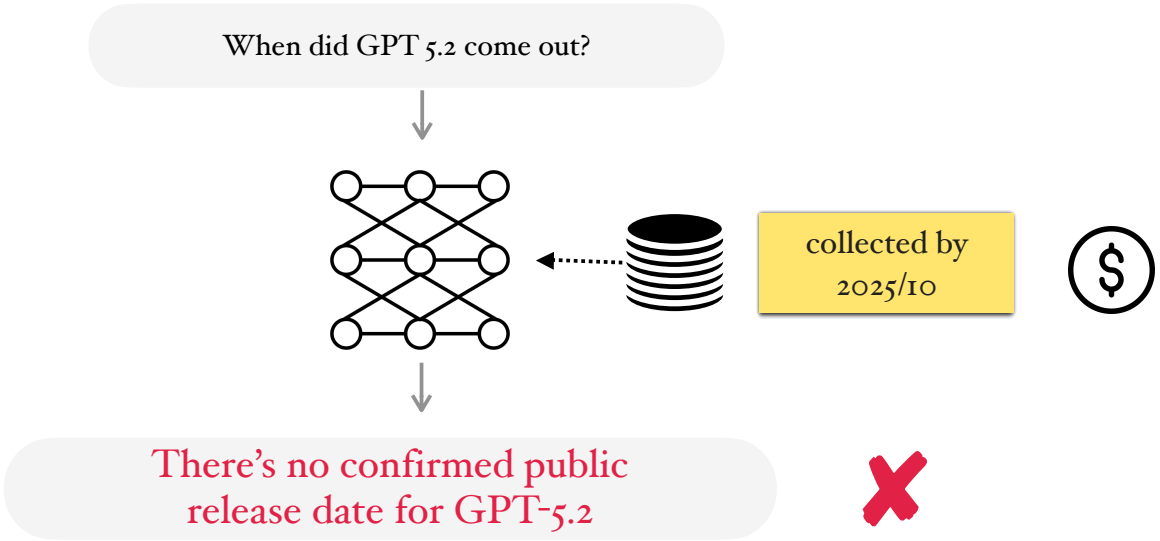
Reduces hallucinations

Parameter efficiency

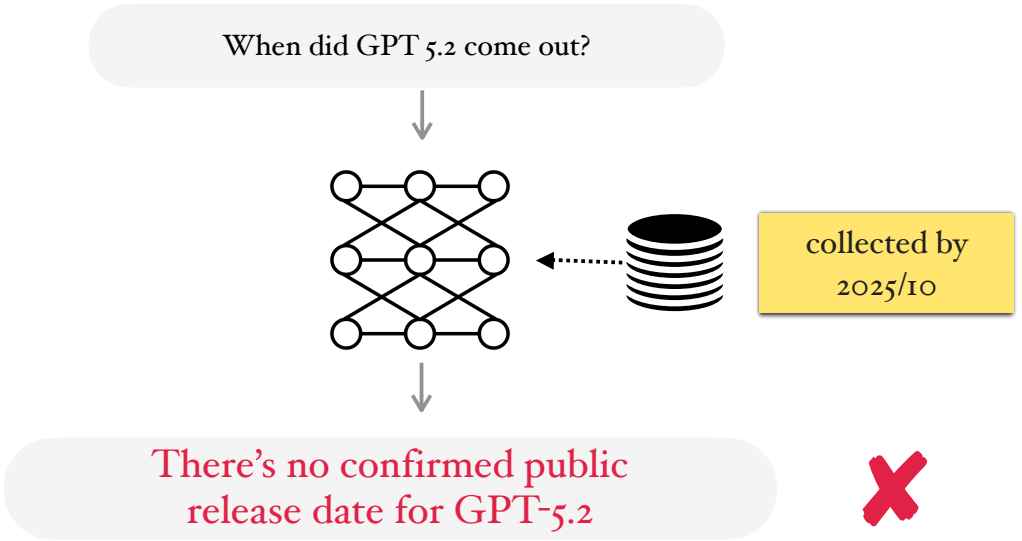
Retrieval-Augmented LMs: Intuitions



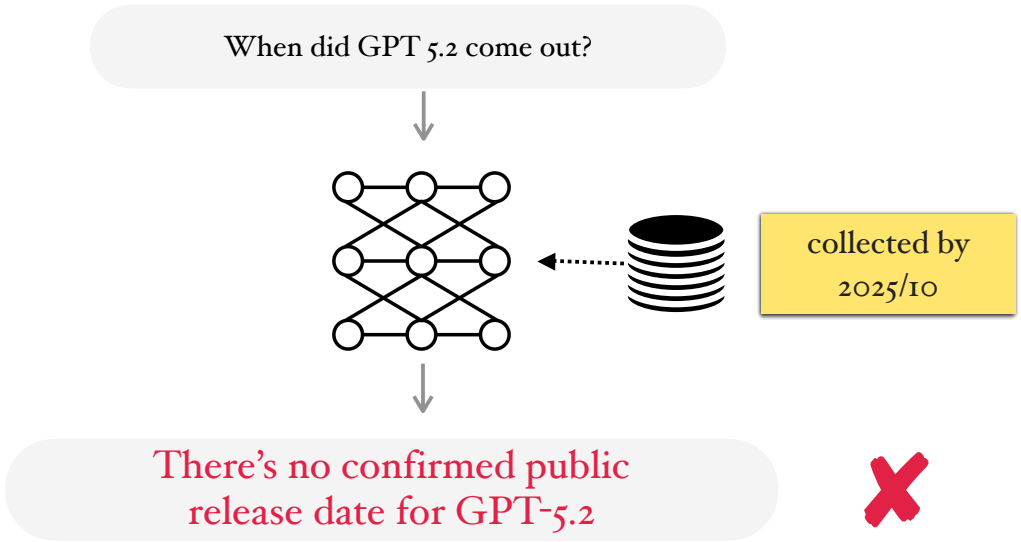
Retrieval-Augmented LMs: Intuitions



Retrieval-Augmented LMs: Intuitions



Retrieval-Augmented LMs: Intuitions

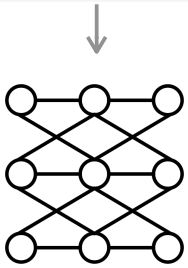


Retrieval-Augmented LMs: Intuitions

When did GPT 5.2 come out?



When did GPT 5.2 come out?



collected by 2025/10

There's no confirmed public release date for GPT-5.2



Retrieval-Augmented LMs: Intuitions

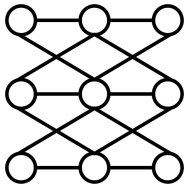
When did GPT 5.2 come out?

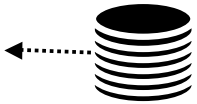


Dec 11, 2025 — GPT-5.2 Instant is a fast, capable workhorse for

Updated by 2026/01

When did GPT 5.2 come out?



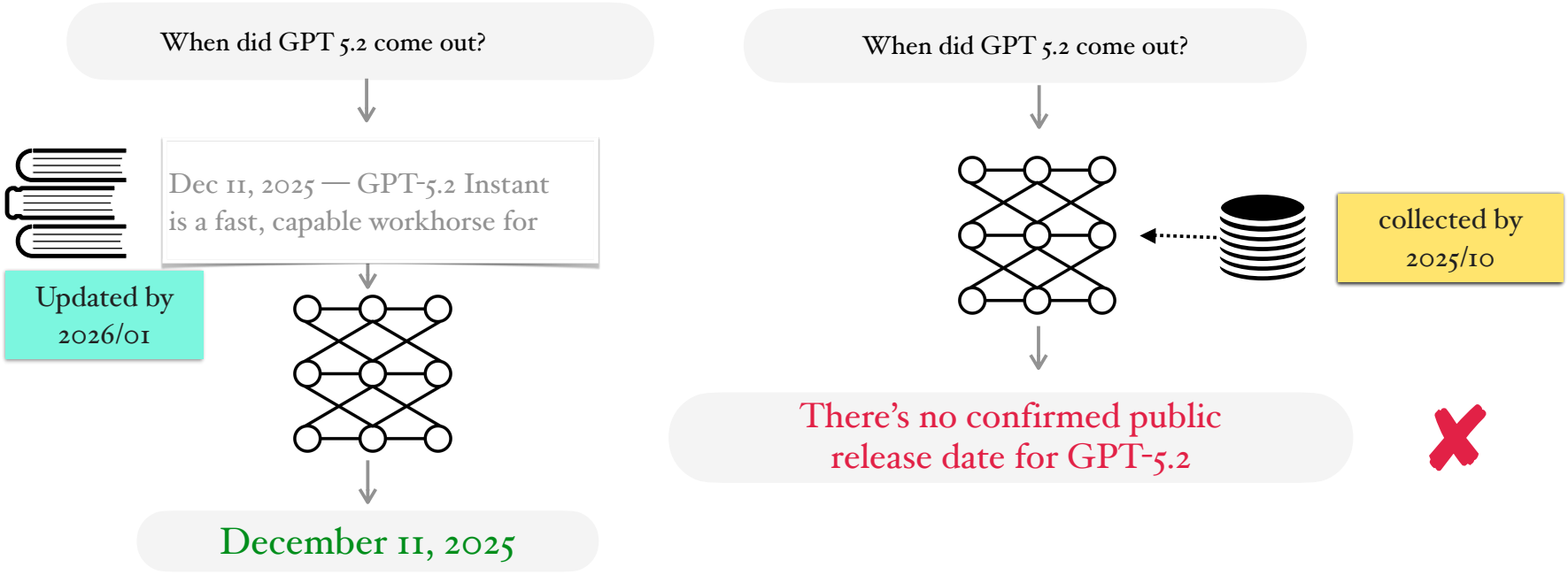


collected by 2025/10

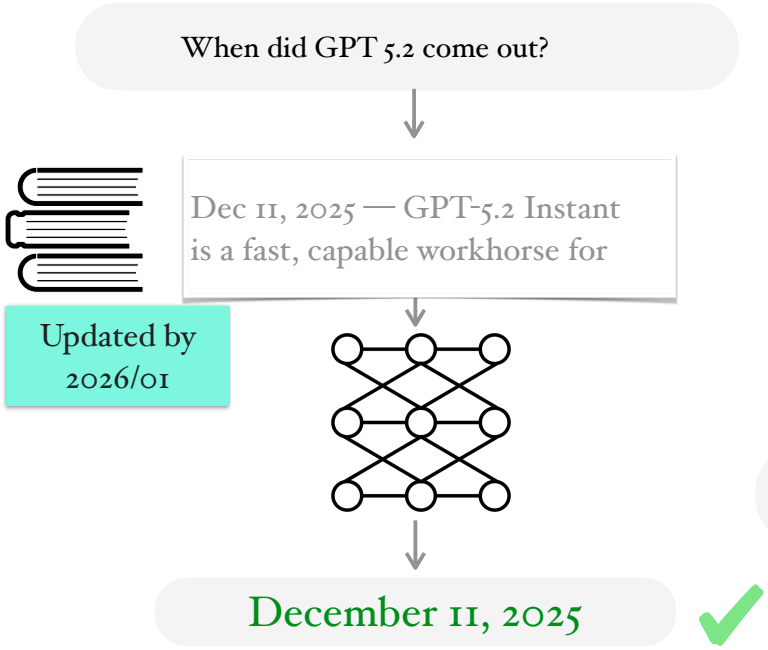
There's no confirmed public release date for GPT-5.2

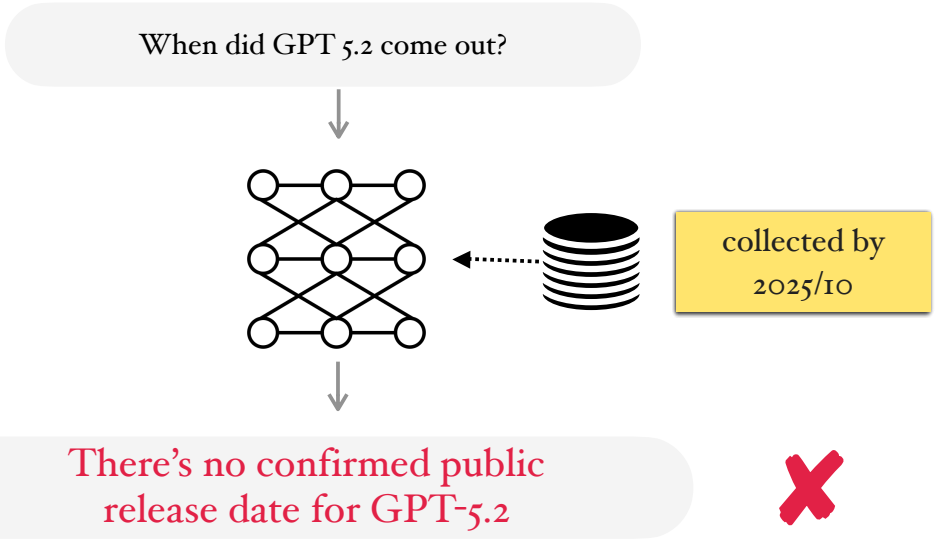


Retrieval-Augmented LMs: Intuitions

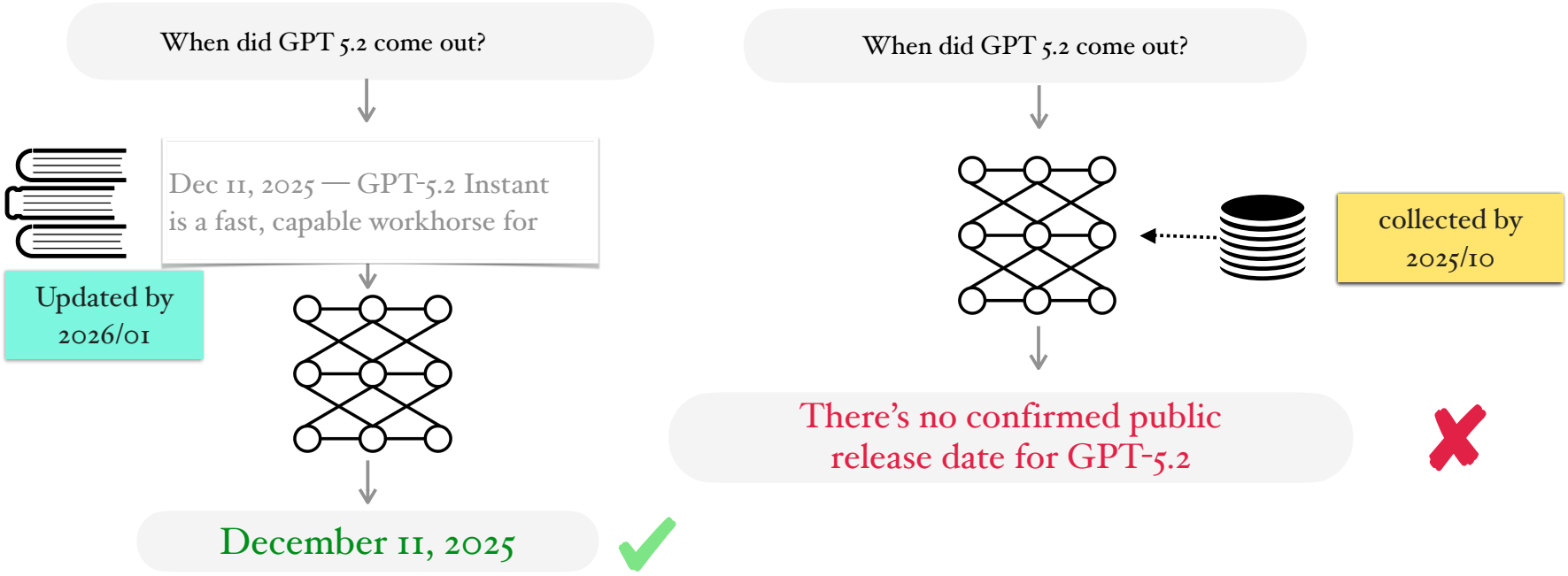


Retrieval-Augmented LMs: Intuitions



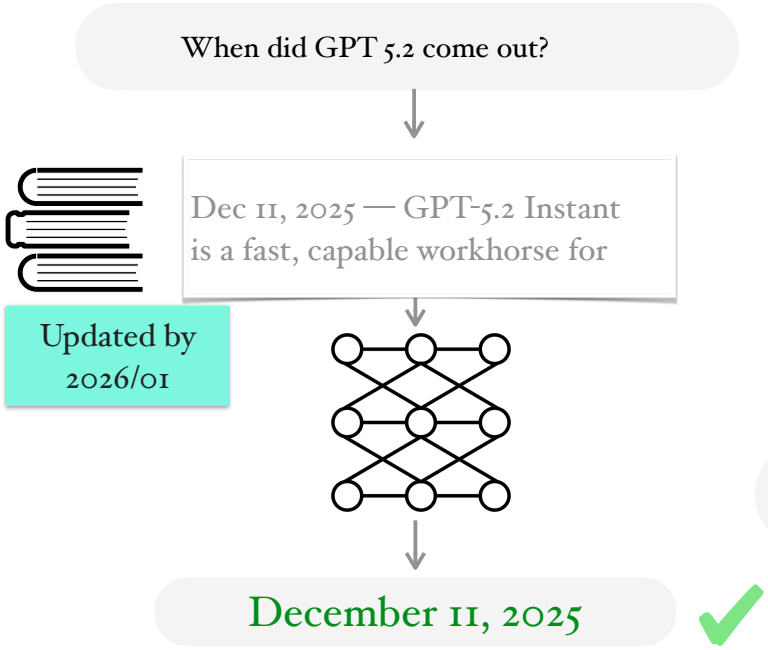


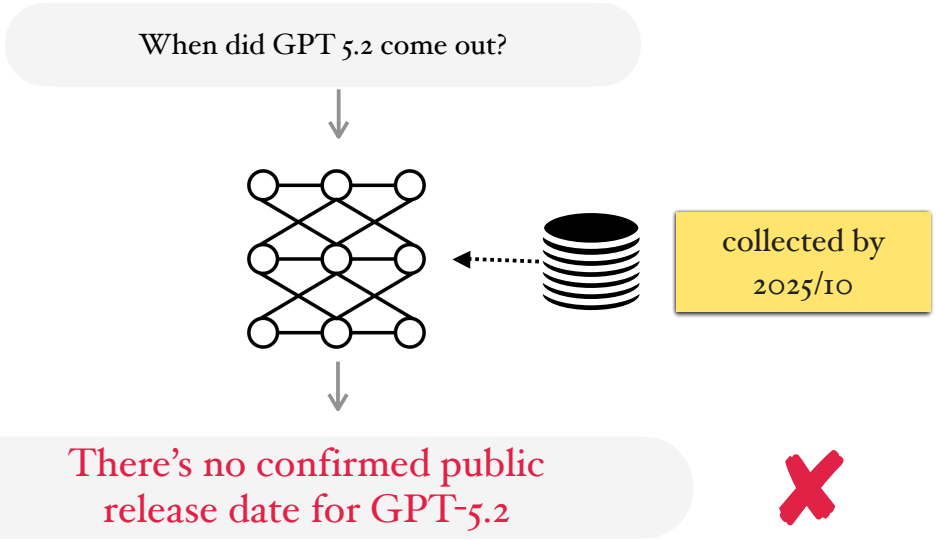
Retrieval-Augmented LMs: Intuitions



Update knowledge w/o retraining

Retrieval-Augmented LMs: Intuitions





Update knowledge w/o retraining

Improve verifiability

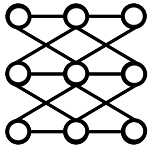
Overview



Datastore



Retriever



LM

Overview

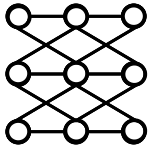
x : When did GPT 5.2 come out?



Datastore



Retriever



LM

Overview

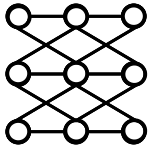
x : When did GPT 5.2 come out?



Datastore



Retriever



LM

$\text{Sim}(\cdot \mid x)$

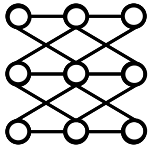
Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for	0.9
Ruby on Rails 5.2 Release Notes Ruby on Rails 5.2 Release Notes	0.1
GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was	0.7

Overview

x : When did GPT 5.2 come out?







Datastore **Retriever** **LM**

$$D \in \text{Top}_k \text{Sim}(\cdot \mid x)$$

Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for

0.9

Ruby on Rails 5.2 Release Notes Ruby on Rails 5.2 Release Notes

0.1

GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was

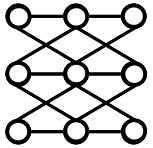
0.7

Overview

x : When did GPT 5.2 come out?







Datastore **Retriever** **LM**

$D \in \text{Top}_k \text{Sim}(\cdot \mid x)$

Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for

0.9

Ruby on Rails 5.2 Release Notes Ruby on Rails 5.2 Release Notes

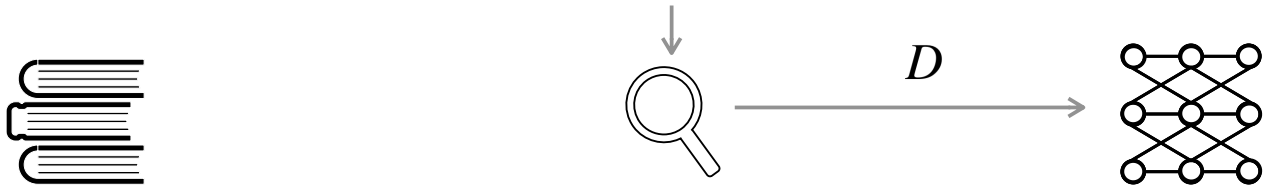
0.1

GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was

0.7

Overview

x : When did GPT 5.2 come out?



Datastore **Retriever** **LM**

$$D \in \text{Top}_k \text{Sim}(\cdot \mid x)$$

Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for

0.9

Ruby on Rails 5.2 Release Notes Ruby on Rails 5.2 Release Notes

0.1

GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was

0.7

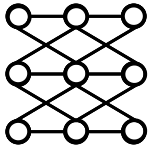
Overview

x : When did GPT 5.2 come out?





D



y : Dec, 2026

Datastore **Retriever** **LM**

$$D \in \text{Top}_k \text{Sim}(\cdot \mid x)$$

Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for

0.9

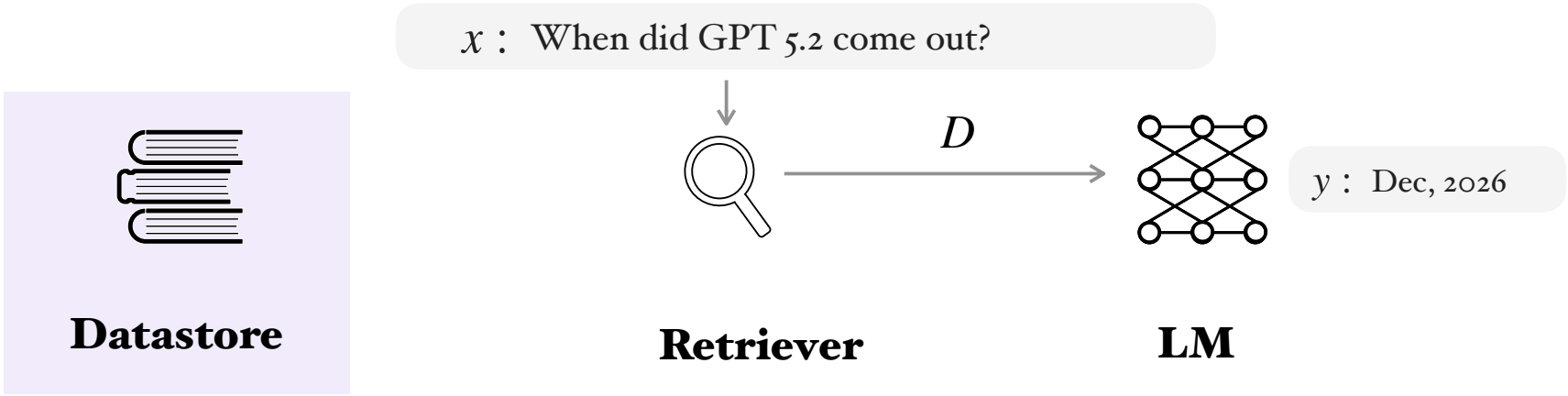
Ruby on Rails 5.2 Release Notes Ruby on Rails 5.2 Release Notes

0.1

GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was

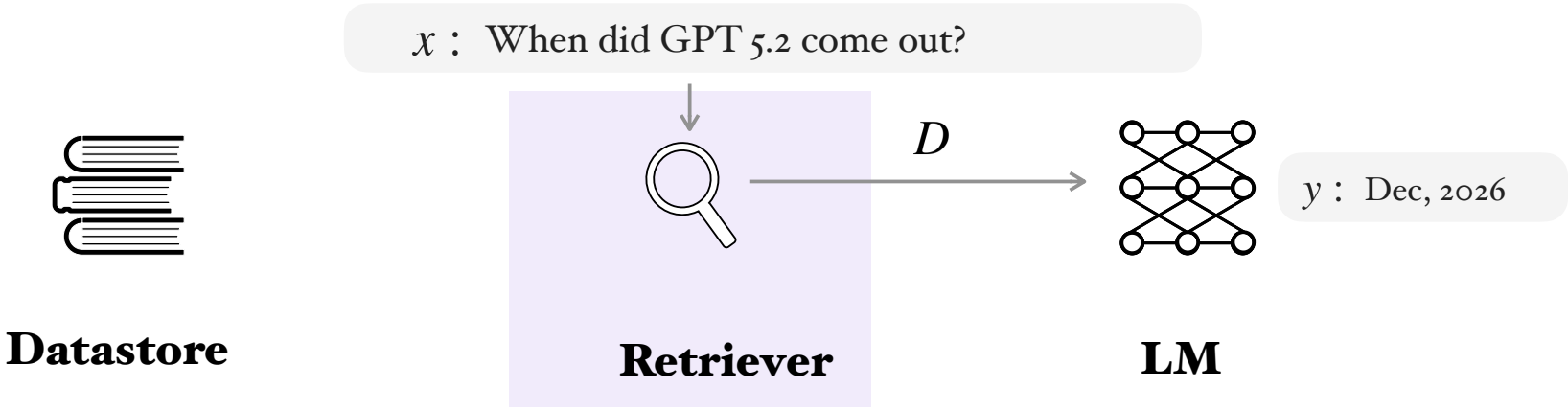
0.7

Key Factors & Design Choices



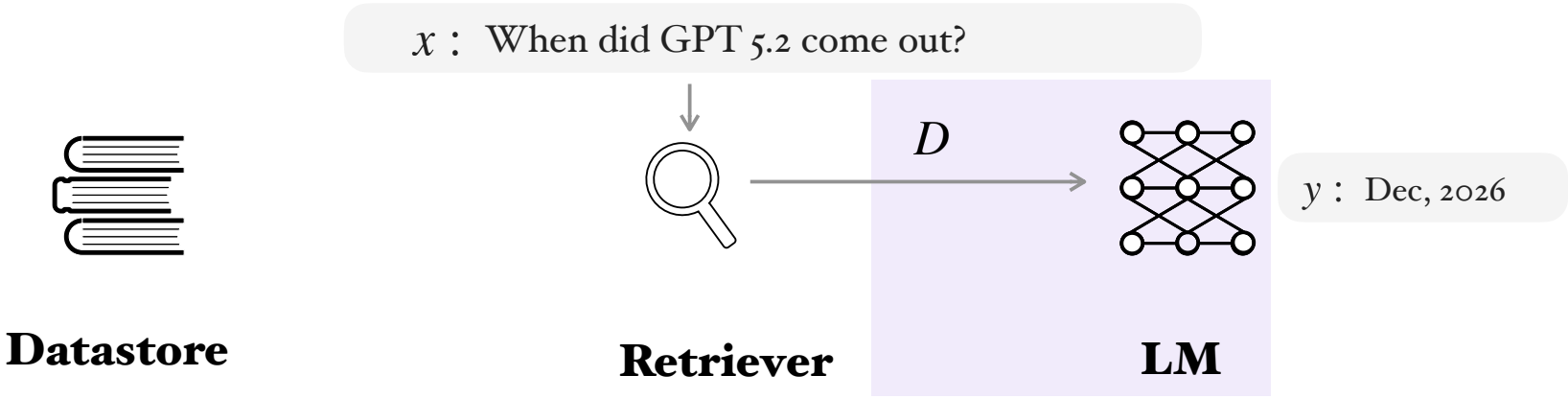
- ✓ Sources of datastore
- ✓ Processing
- ✓ Scaling

Key Factors & Design Choices



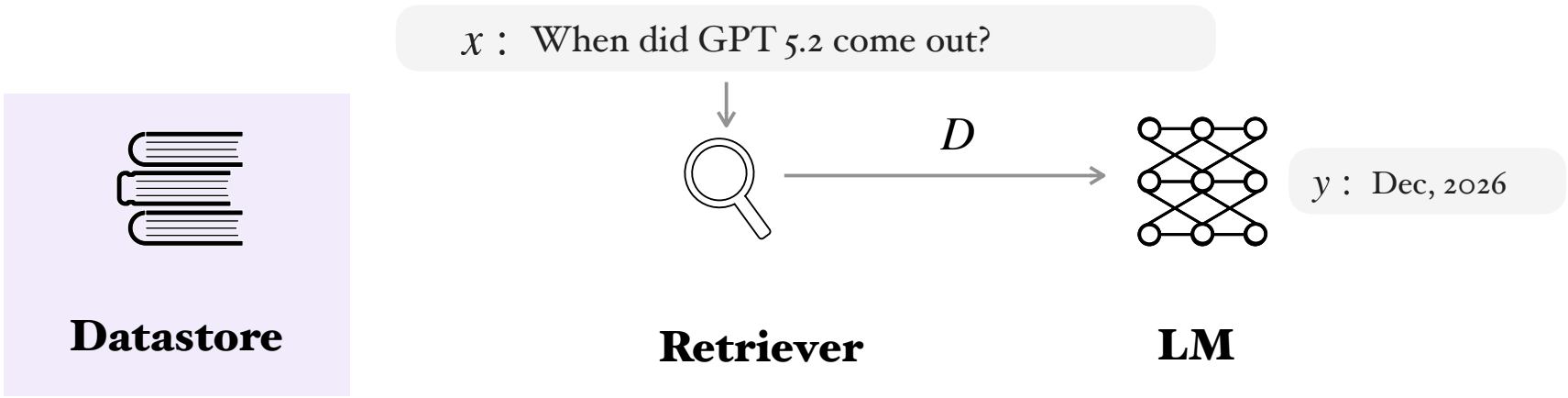
- ✓ Types of retrievers
- ✓ Training
- ✓ Evaluations

Key Factors & Design Choices



- ✓ Architectures
- ✓ Training
- ✓ Inference

Part 1: Datastore



- ✓ Sources of datastore
- ✓ Processing
- ✓ Scaling

What Should be in “data store”?

x : when did GPT 5.2 come out?

x : How should I implement RAG using LlamaIndex?

English Wikipedia



Chen et al., 2017; Gu et al., 2020; Asai et al., 2020; Guu et al., 2021; Lewis et al., 2021 ... etc

https://dumps.wikimedia.org/

Code snippets



Documentations

LangChain

Community forums

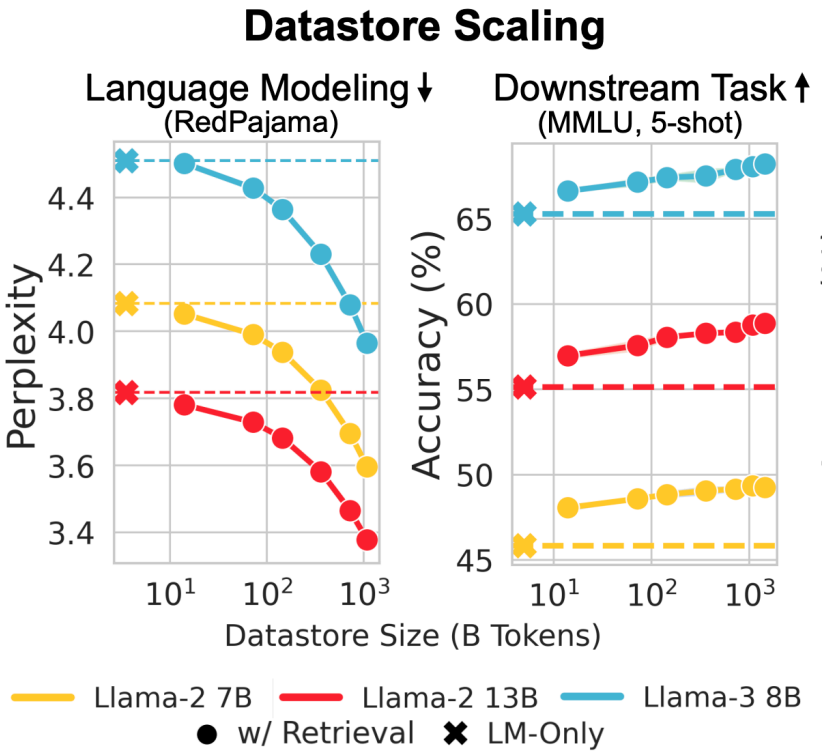
stackoverflow

Scaling Up Datastore



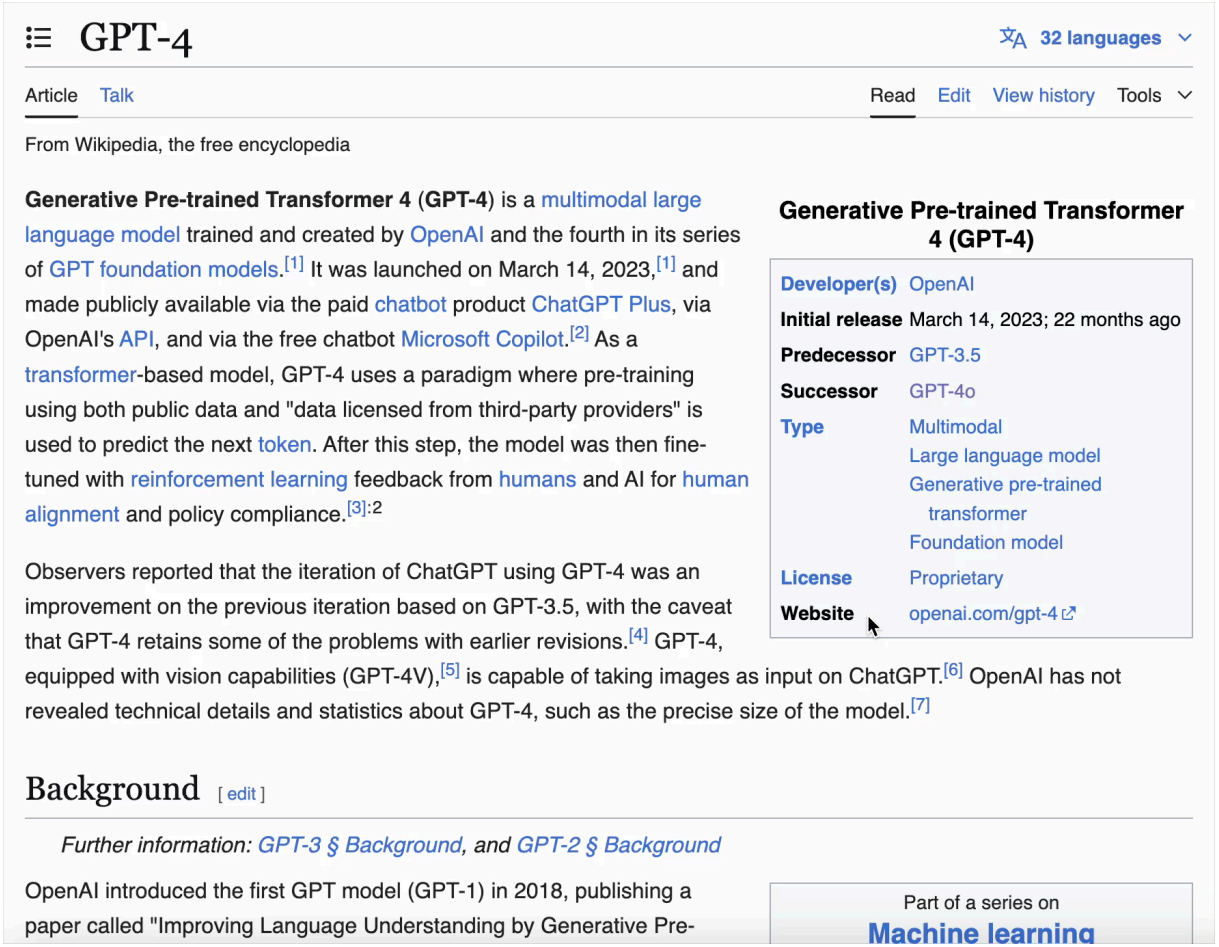
MassiveDS

1.4 trillion tokens



Shao et al. 2024. Scaling Retrieval-Based Language Models with a Trillion-Token Datastore

Processing Documents



GPT-4

32 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

Generative Pre-trained Transformer 4 (**GPT-4**) is a multimodal large language model trained and created by OpenAI and the fourth in its series of GPT foundation models.[1] It was launched on March 14, 2023,[1] and made publicly available via the paid chatbot product ChatGPT Plus, via OpenAI's API, and via the free chatbot Microsoft Copilot.[2] As a transformer-based model, GPT-4 uses a paradigm where pre-training using both public data and "data licensed from third-party providers" is used to predict the next token. After this step, the model was then fine-tuned with reinforcement learning feedback from humans and AI for human alignment and policy compliance.[3]:2

Generative Pre-trained Transformer 4 (GPT-4)

Developer(s)	OpenAI
Initial release	March 14, 2023; 22 months ago
Predecessor	GPT-3.5
Successor	GPT-4o
Type	Multimodal Large language model Generative pre-trained transformer Foundation model
License	Proprietary
Website	openai.com/gpt-4

Observers reported that the iteration of ChatGPT using GPT-4 was an improvement on the previous iteration based on GPT-3.5, with the caveat that GPT-4 retains some of the problems with earlier revisions.[4] GPT-4, equipped with vision capabilities (GPT-4V),[5] is capable of taking images as input on ChatGPT.[6] OpenAI has not revealed technical details and statistics about GPT-4, such as the precise size of the model.[7]

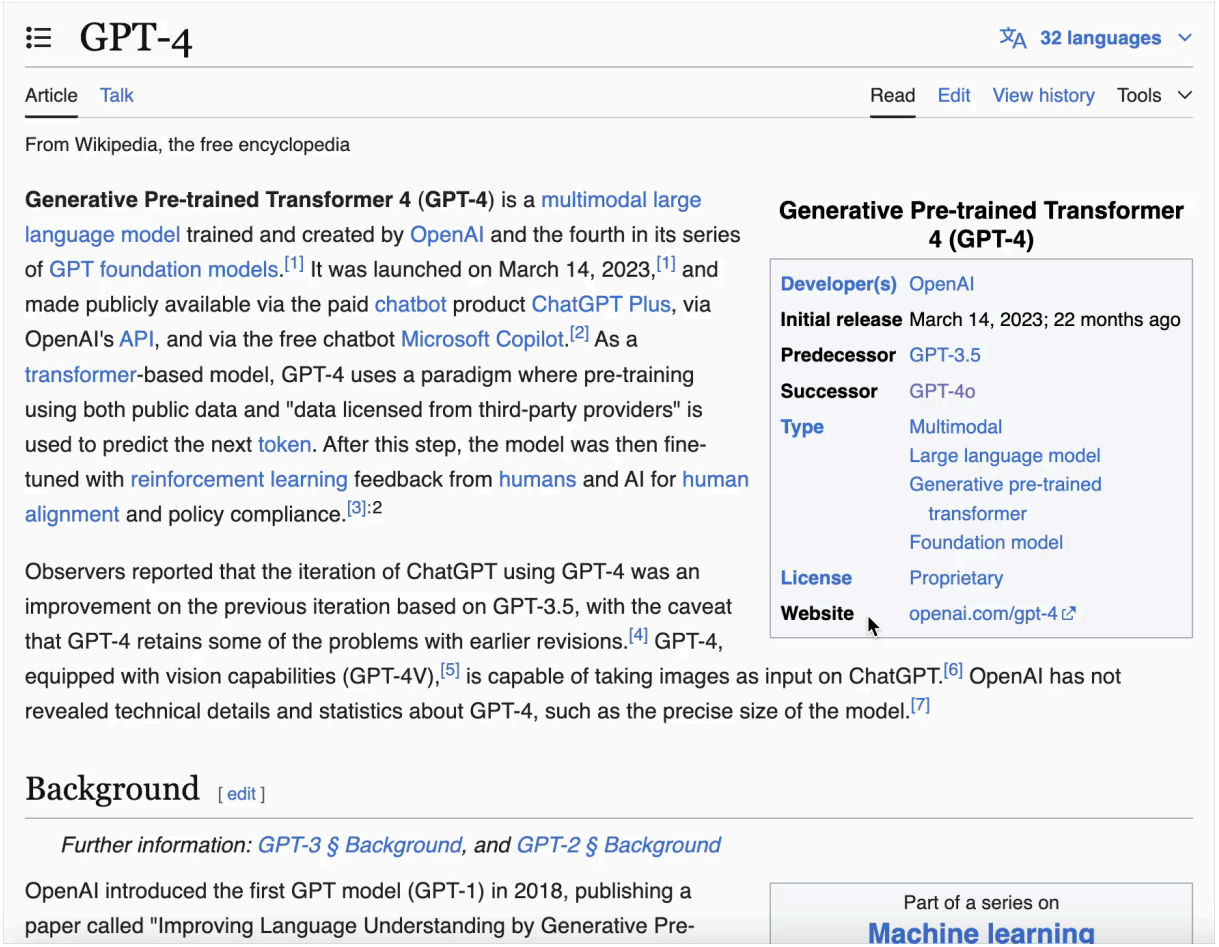
Background [edit]

Further information: GPT-3 § Background, and GPT-2 § Background

OpenAI introduced the first GPT model (GPT-1) in 2018, publishing a paper called "Improving Language Understanding by Generative Pre-

Part of a series on
Machine learning

Processing Documents



GPT-4

32 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

Generative Pre-trained Transformer 4 (**GPT-4**) is a multimodal large language model trained and created by OpenAI and the fourth in its series of GPT foundation models.[1] It was launched on March 14, 2023,[1] and made publicly available via the paid chatbot product ChatGPT Plus, via OpenAI's API, and via the free chatbot Microsoft Copilot.[2] As a transformer-based model, GPT-4 uses a paradigm where pre-training using both public data and "data licensed from third-party providers" is used to predict the next token. After this step, the model was then fine-tuned with reinforcement learning feedback from humans and AI for human alignment and policy compliance.[3]:2

Generative Pre-trained Transformer 4 (GPT-4)

Developer(s)	OpenAI
Initial release	March 14, 2023; 22 months ago
Predecessor	GPT-3.5
Successor	GPT-4o
Type	Multimodal Large language model Generative pre-trained transformer Foundation model
License	Proprietary
Website	openai.com/gpt-4

Observers reported that the iteration of ChatGPT using GPT-4 was an improvement on the previous iteration based on GPT-3.5, with the caveat that GPT-4 retains some of the problems with earlier revisions.[4] GPT-4, equipped with vision capabilities (GPT-4V),[5] is capable of taking images as input on ChatGPT.[6] OpenAI has not revealed technical details and statistics about GPT-4, such as the precise size of the model.[7]

Background [edit]

Further information: GPT-3 § Background, and GPT-2 § Background

OpenAI introduced the first GPT model (GPT-1) in 2018, publishing a paper called "Improving Language Understanding by Generative Pre-

Part of a series on
Machine learning

Processing Documents

Curate and preprocess data → **Chunking** → **Post-processing**

e.g., HTML -> Plain text

Paragraph-level (e.g., \n)
Every k words (e.g., 100-250)

e.g., Remove short documents



GPT-4o is a pre-trained

Transformers is a series of science fiction action films based

GPT4o was released by OpenAI in May

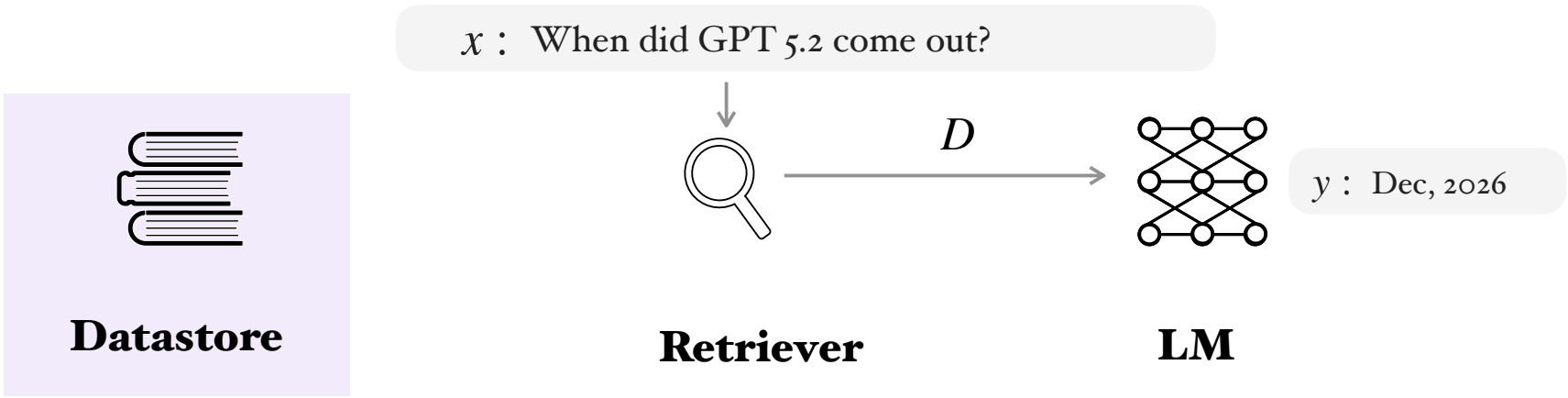
@I$O@

GPT-4o is a pre-trained

Transformers is a series of science fiction action films based

GPT4o was released by OpenAI in May

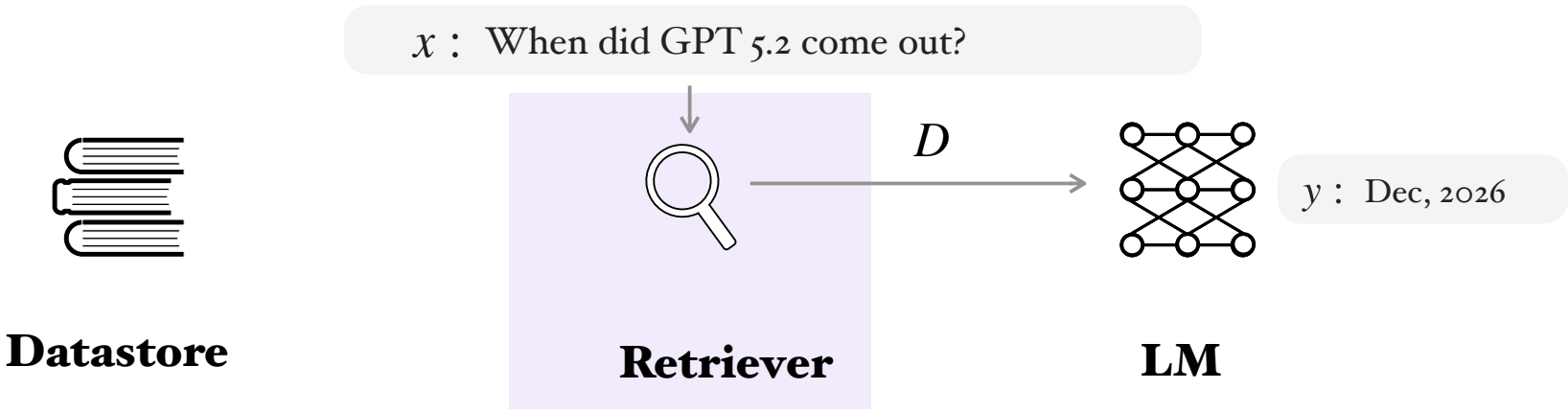
Summary of Part 1



- ✓ Sources of datastore
- ✓ Processing
- ✓ Scaling

- Choosing **the right datastore**
- **Chunking** and **filtering**
- **Scaling** datastores offer performance gain while adding challenges

Part 2: Retriever



- ✓ Types of retrievers
- ✓ Training
- ✓ Evaluations

Types of Retrievers

$$D \in \text{Top}_k\text{Sim}(\cdot \mid x)$$

Sparse retrievers

- **Sim:** Term-frequency based embeddings

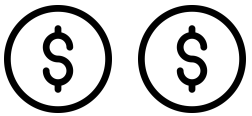
e.g., TF-IDF, BM25



Dense retrievers

- **Sim**: dense embeddings encoded by pre-

e.g., DPR, Contriever, ColBERT



Rerankers

- **Sim**: Scores based on jointly encoded query and doc

e.g., cross-encoder reranker



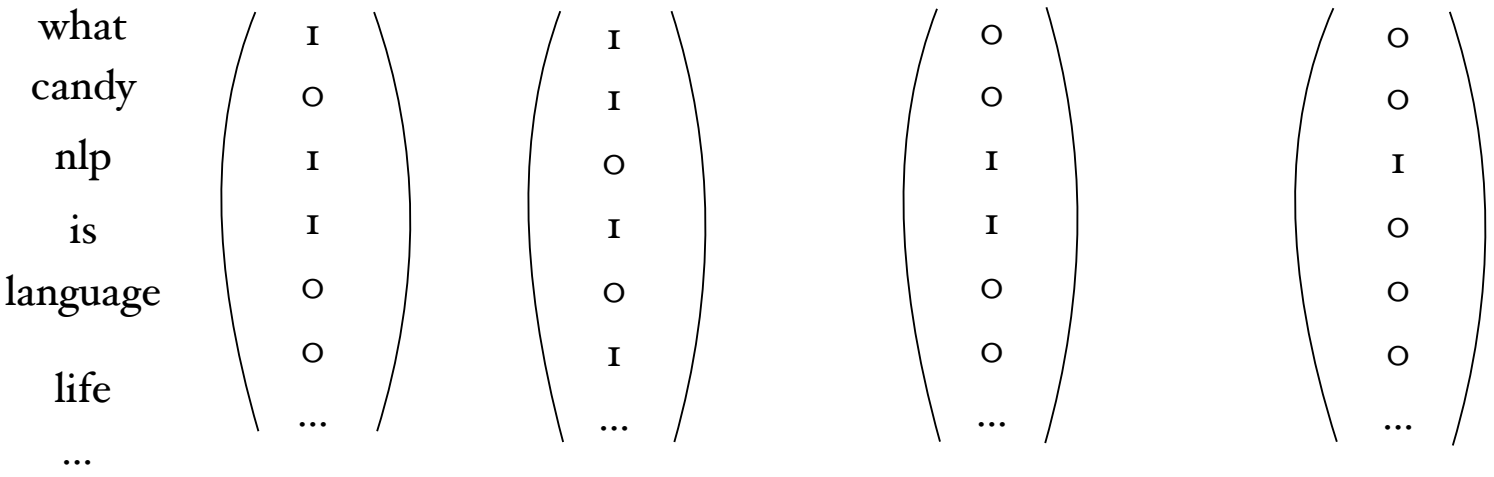
Sparse Retrievers

q=what is nlp

d_1 = what is life ? candy is life !

d_2 = nlp is an acronym for natural language

d_3 = I like to do good research on



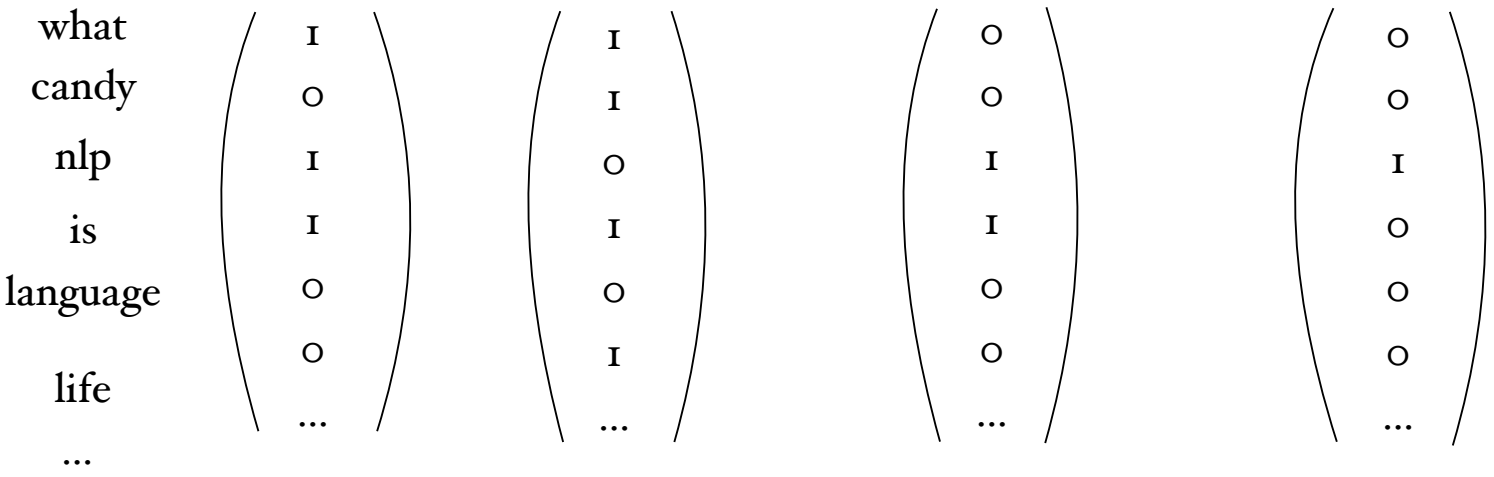
Sparse Retrievers

q=what is nlp

d_1 = what is life ? candy is life !

d_2 = nlp is an acronym for natural language

d_3 = I like to do good research on



Check if a term appears in a document

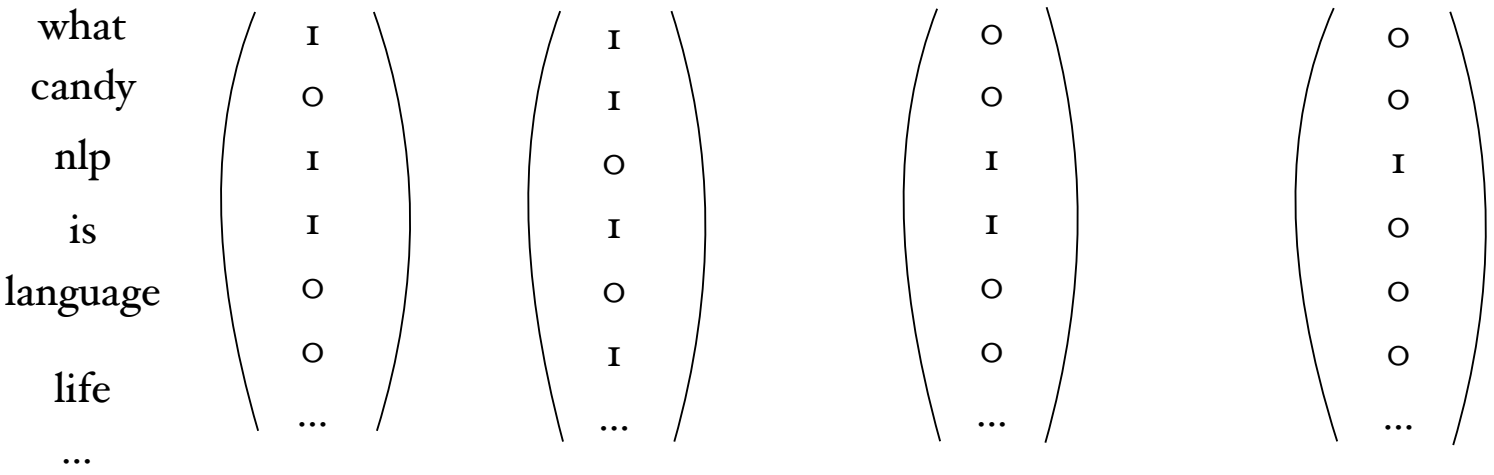
One-hot Vector

q=what is nlp

d_1 = what is life ? candy is life !

d_2 = nlp is an acronym for natural language

d_3 = I like to do good research on



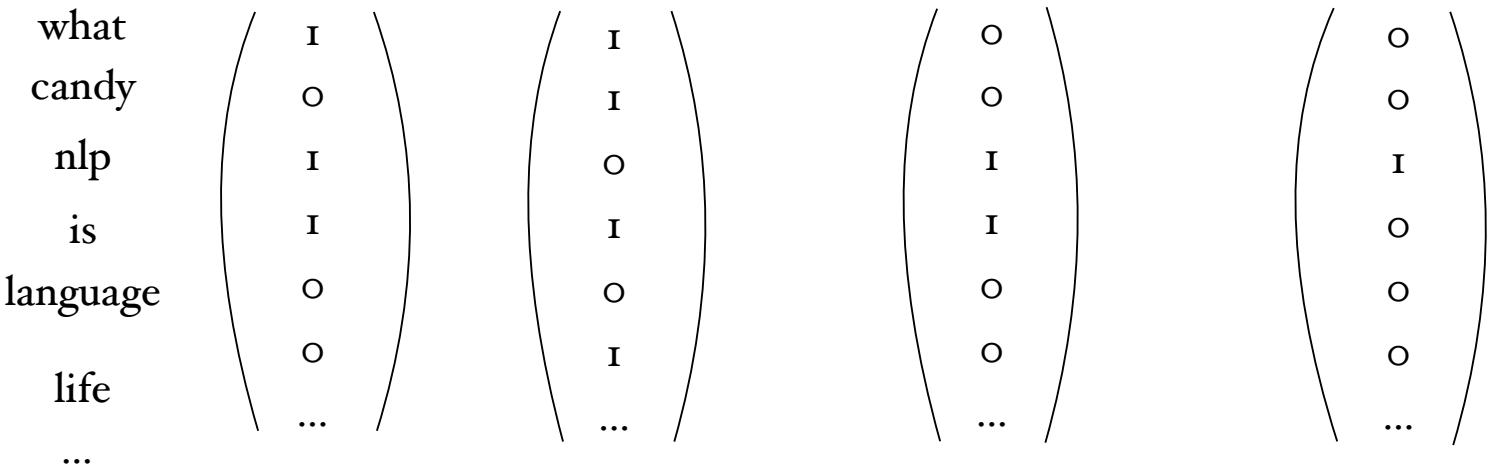
One-hot Vector

q=what is nlp

d_1 = what is life ? candy is life !

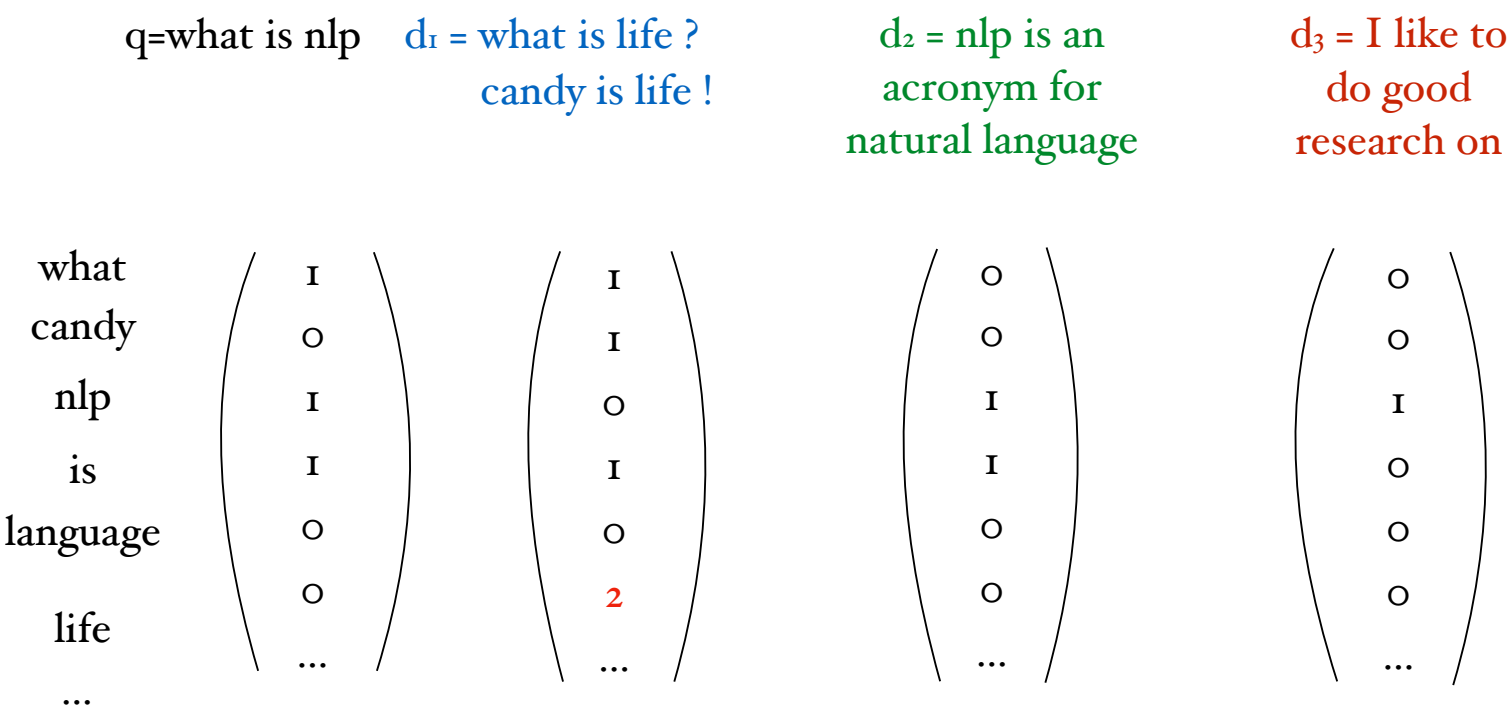
d_2 = nlp is an acronym for natural language

d_3 = I like to do good research on



Check if a term appears in a document

Term-count Vector



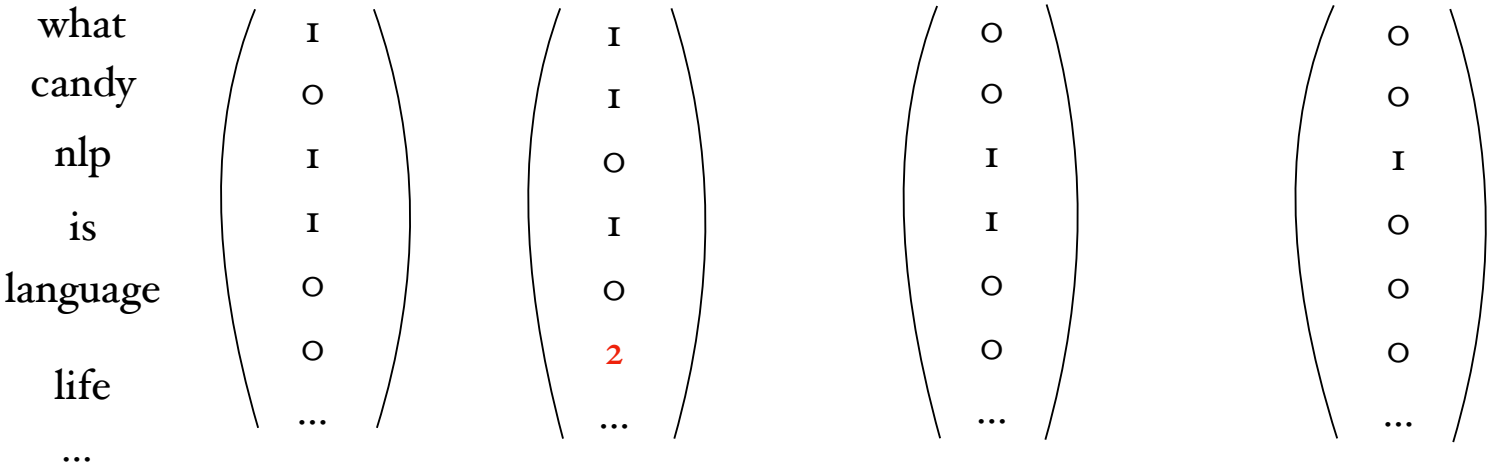
Term-count Vector

q=what is nlp

d_1 = what is life ? candy is life !

d_2 = nlp is an acronym for natural language

d_3 = I like to do good research on



Count the number of appearances in a doc

Weighted-term Score

$$\mathrm{TF}(t,d) = \frac{\mathrm{freq}(t,d)}{\sum_{t'} \mathrm{freq}(t',d)} \qquad \mathrm{IDF}(t) = \log\left(\frac{|D|}{\sum_{d' \in D} \delta(\mathrm{freq}(t,d') > 0)}\right)$$

$$\text{TF-IDF}(t,d) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t)$$

$$\text{BM-25}(t,d) = \mathrm{IDF}(t) \cdot \frac{\mathrm{freq}(t,d) \cdot (k_1 + 1)}{\mathrm{freq}(t,d) + k_1 \cdot \left(1 - b + b \cdot \frac{|d|}{\mathrm{avgdl}}\right)}$$

Robertson et al. 2009. The Probabilistic Relevance Framework: BM25 and Beyond.

Weighted-term Score

d_1 = what is life ?
candy is life !

t_1 = what

$$\mathrm{TF}(t,d) = \frac{\mathrm{freq}(t,d)}{\sum_{t'} \mathrm{freq}(t',d)} \qquad \mathrm{IDF}(t) = \log\left(\frac{|D|}{\sum_{d' \in D} \delta(\mathrm{freq}(t,d') > 0)}\right)$$

$$\text{TF-IDF}(t,d) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t)$$

$$\text{BM-25}(t,d) = \mathrm{IDF}(t) \cdot \frac{\mathrm{freq}(t,d) \cdot (k_1 + 1)}{\mathrm{freq}(t,d) + k_1 \cdot \left(1 - b + b \cdot \frac{|d|}{\mathrm{avgdl}}\right)}$$

Robertson et al. 2009. The Probabilistic Relevance Framework: BM25 and Beyond.

Weighted-term Score

d_1 = what is life ?
candy is life !

t_1 = what

$$\mathrm{TF}(t,d) = \frac{\mathrm{freq}(t,d)}{\sum_{t'} \mathrm{freq}(t',d)}$$

# of documents

$$\mathrm{IDF}(t) = \log\left(\frac{|D|}{\sum_{d' \in D} \delta(\mathrm{freq}(t,d') > 0)}\right)$$

$$\text{TF-IDF}(t,d) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t)$$

$$\text{BM-25}(t,d) = \mathrm{IDF}(t) \cdot \frac{\mathrm{freq}(t,d) \cdot (k_1 + 1)}{\mathrm{freq}(t,d) + k_1 \cdot \left(1 - b + b \cdot \frac{|d|}{\mathrm{avgdl}}\right)}$$

Robertson et al. 2009. The Probabilistic Relevance Framework: BM25 and Beyond.

Weighted-term Score

d_1 = what is life ?
candy is life !

t_1 = what

$$\mathrm{TF}(t,d) = \frac{\mathrm{freq}(t,d)}{\sum_{t'} \mathrm{freq}(t',d)}$$

# of documents

$$\mathrm{IDF}(t) = \log\left(\frac{|D|}{\sum_{d' \in D} \delta(\mathrm{freq}(t,d') > 0)}\right)$$

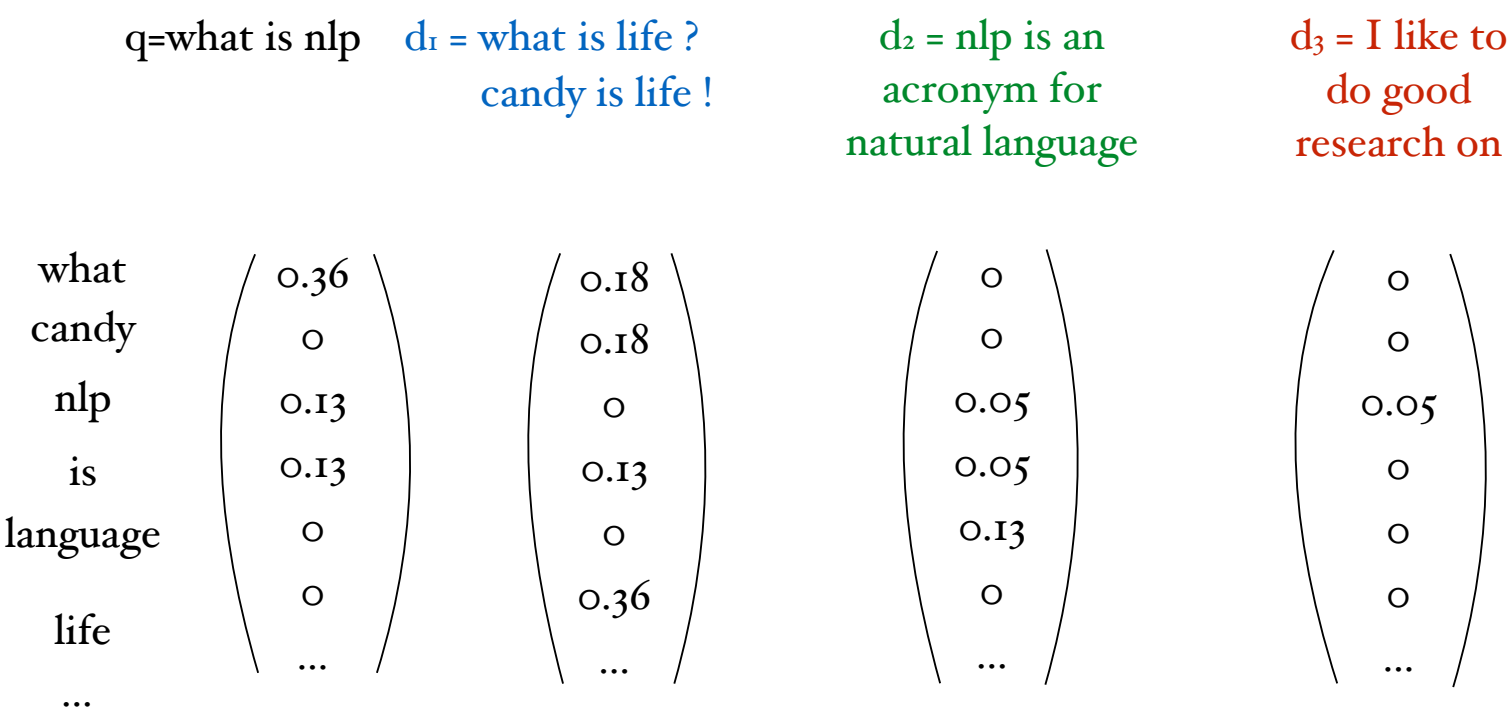
# of documents **where term t appears**

$$\text{TF-IDF}(t,d) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t)$$

$$\text{BM-25}(t,d) = \mathrm{IDF}(t) \cdot \frac{\mathrm{freq}(t,d) \cdot (k_1+1)}{\mathrm{freq}(t,d) + k_1 \cdot \left(1 - b + b \cdot \frac{|d|}{\text{avgdl}}\right)}$$

Robertson et al. 2009. The Probabilistic Relevance Framework: BM25 and Beyond.

Weighted-term Vectors



Compute TF-IDF weights to build weighted vectors

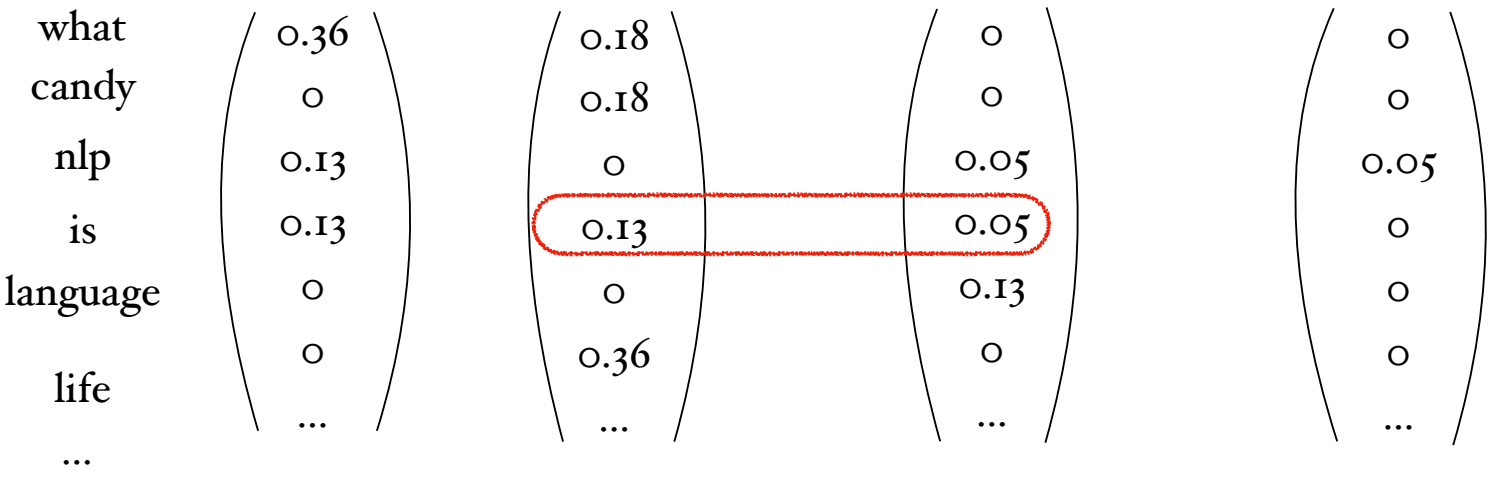
Weighted-term Vectors

q=what is nlp

d_1 = what is life ? candy is life !

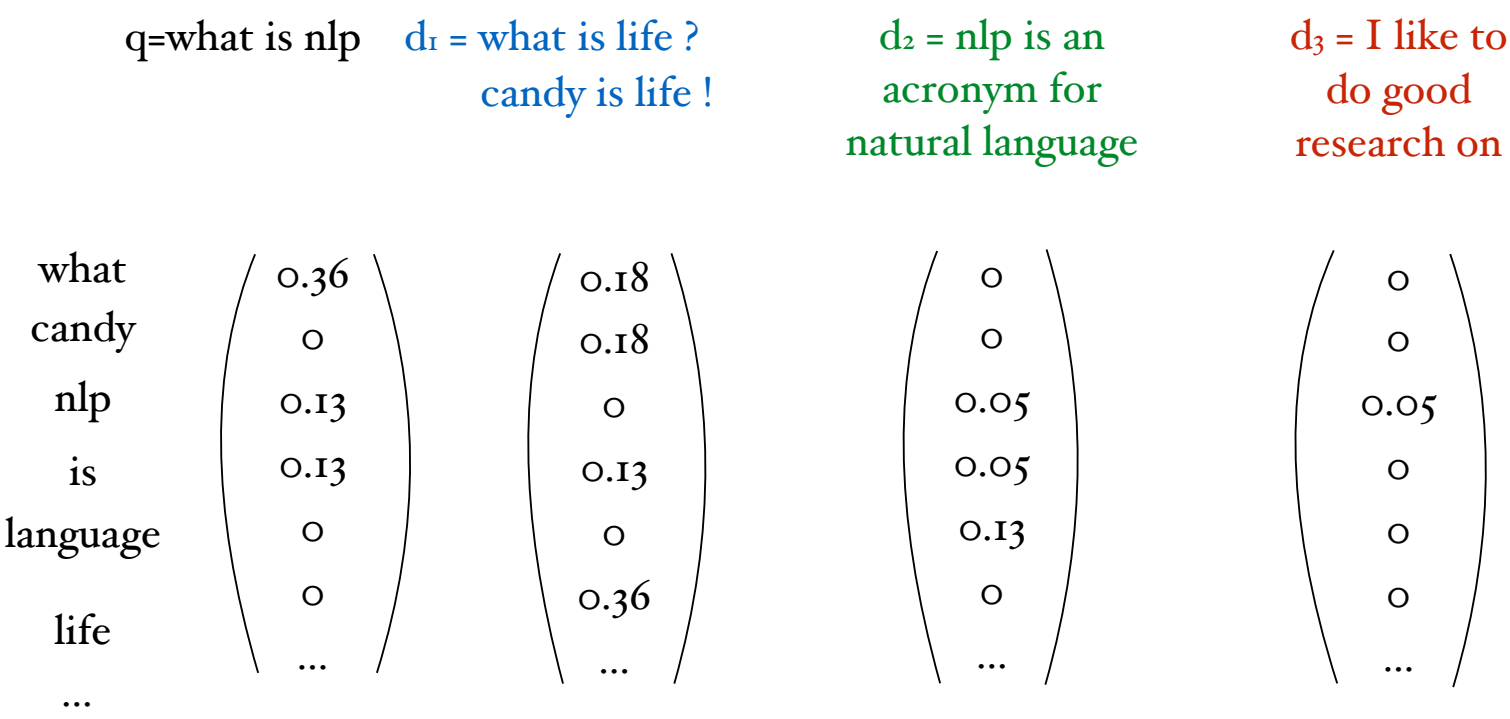
d_2 = nlp is an acronym for natural language

d_3 = I like to do good research on



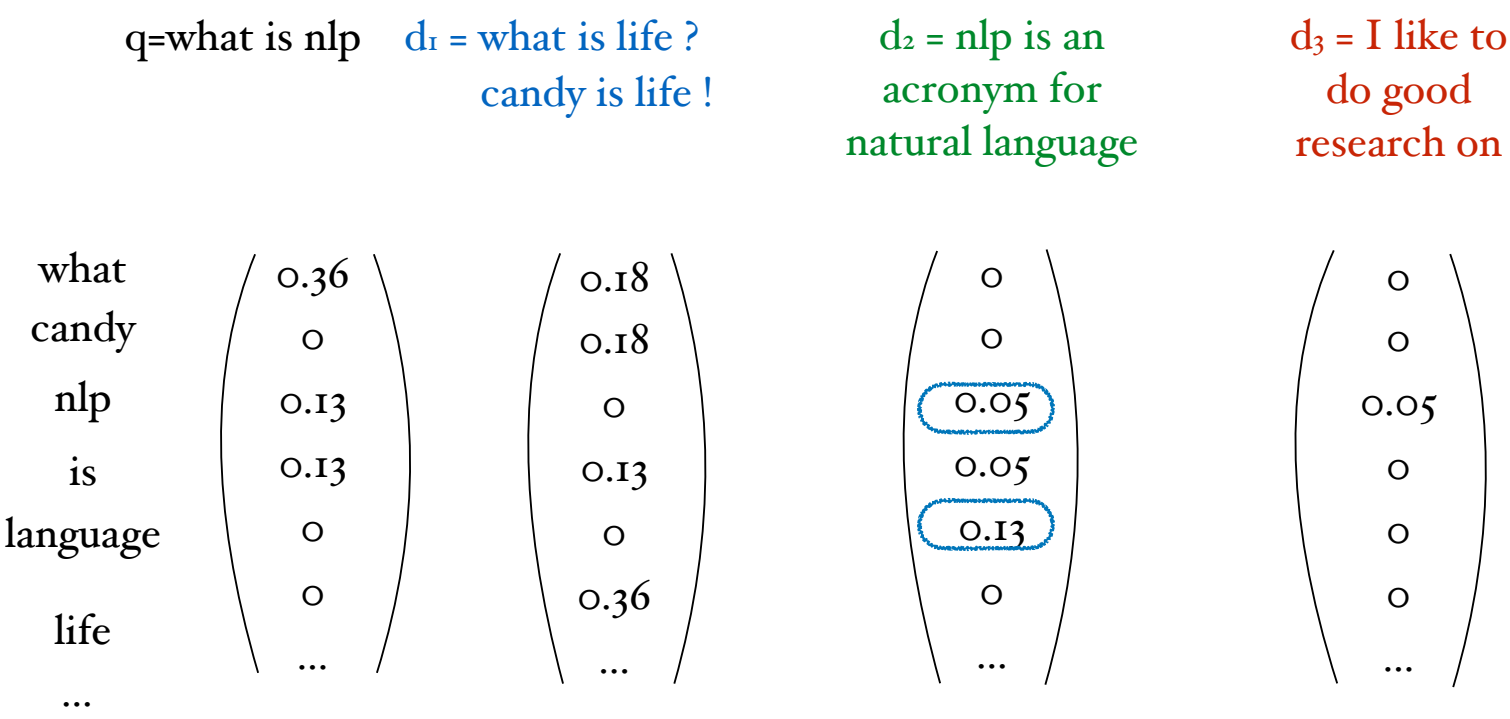
Compute TF-IDF weights to build weighted vectors

Weighted-term Vectors



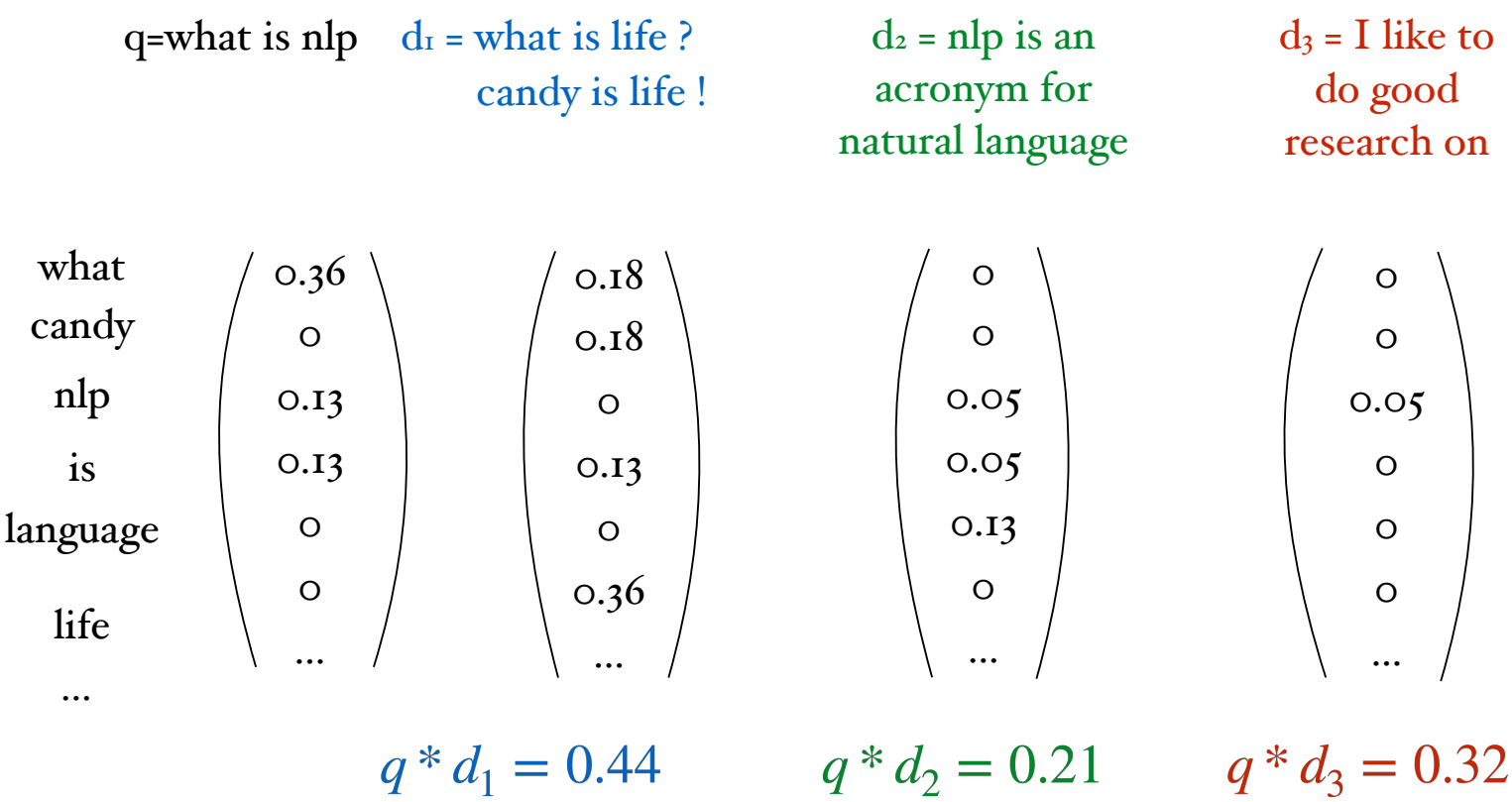
Compute TF-IDF weights to build weighted vectors

Weighted-term Vectors



Compute TF-IDF weights to build weighted vectors

Compute cosine similarity



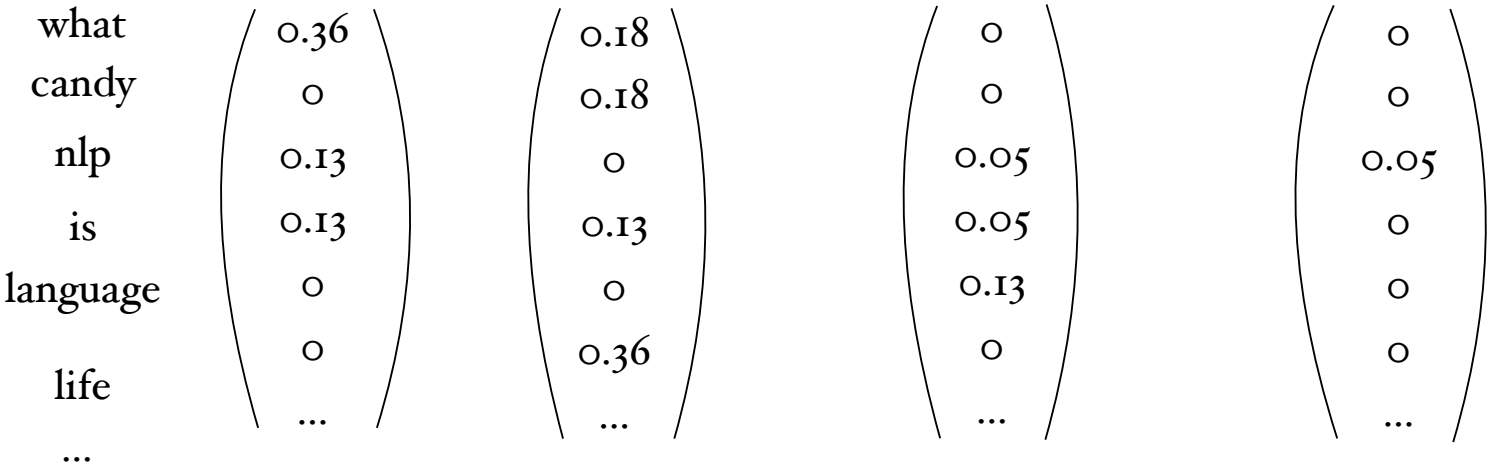
Compute cosine similarity

q=what is nlp

d_1 = what is life ? candy is life !

d_2 = nlp is an acronym for natural language

d_3 = I like to do good research on

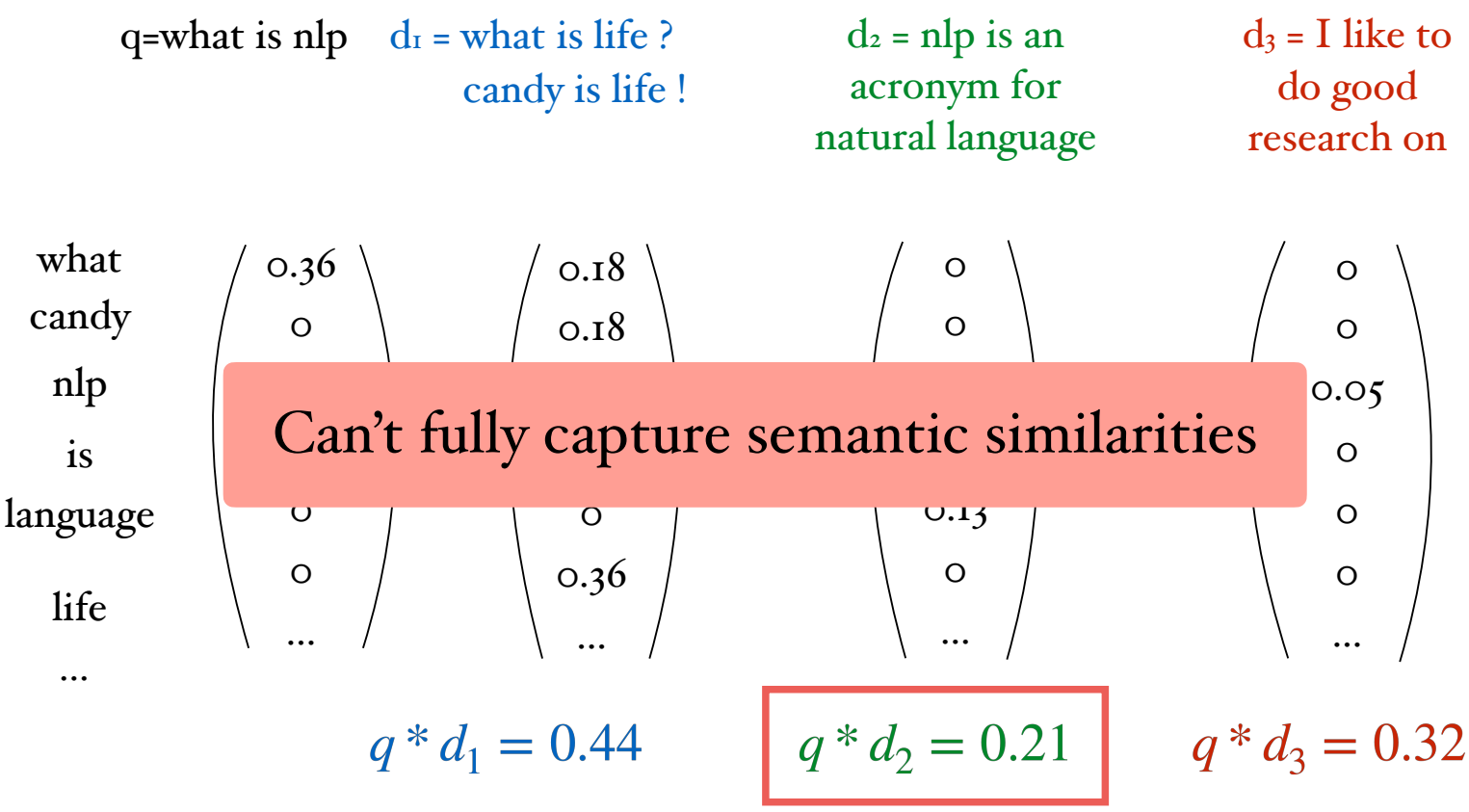


$q * d_1 = 0.44$

$q * d_2 = 0.21$

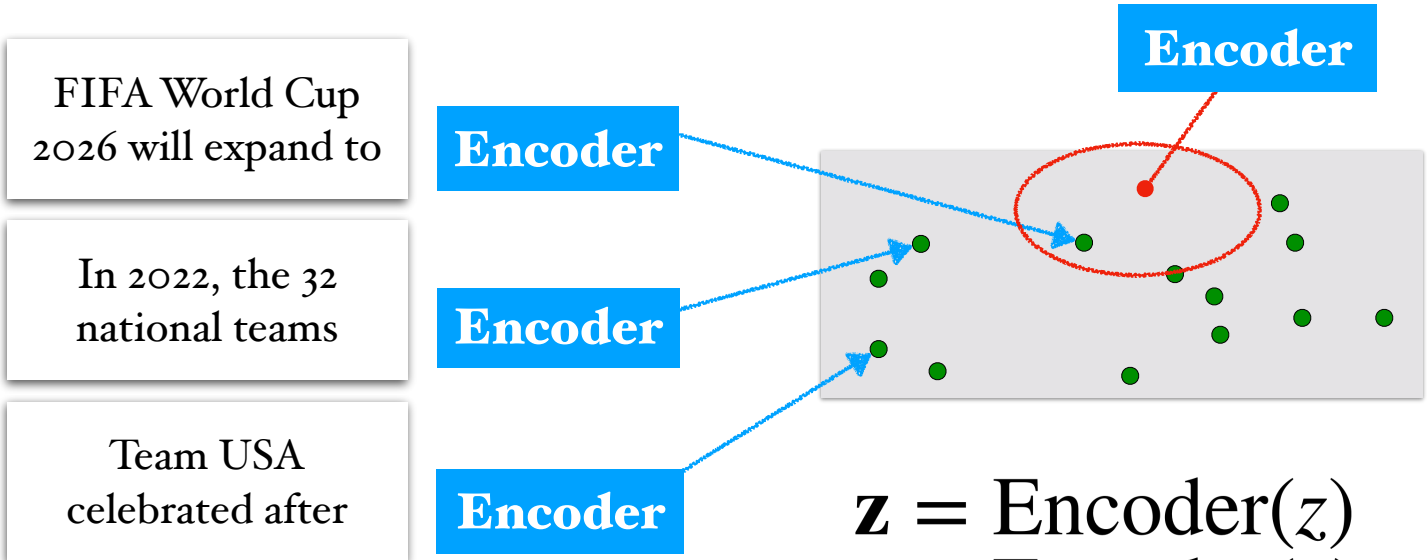
$q * d_3 = 0.32$

Compute cosine similarity



Dense Retrievers: Overview

$\boldsymbol{x}$ = How many teams will participate in FIFA World



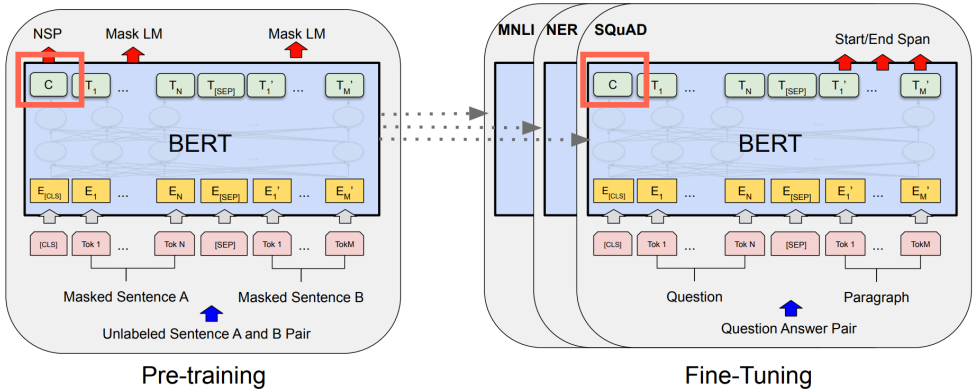
$$\mathbf{z} = \text{Encoder}(z)$$
$$\mathbf{x} = \text{Encoder}(x)$$

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

Dense Retrievers: Embeddings

- Use output vector of [CLS] in masked LMs $\mathbb{R}^d$

  e.g., DPR

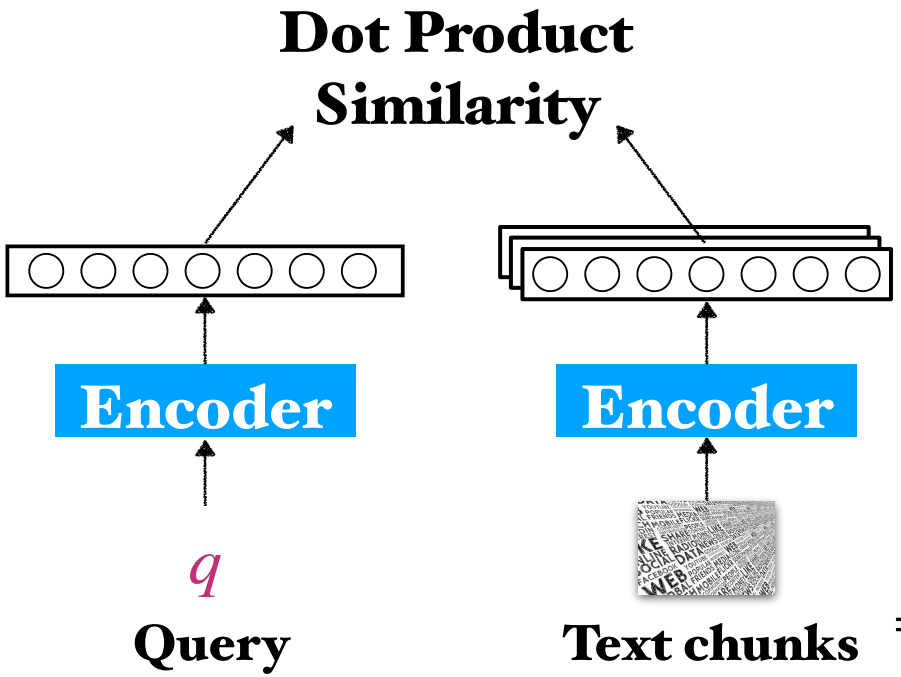


- Mean / Max pooling of output vectors (can be applied to autoregressive LMs) $\mathbb{R}^{N \times d}$

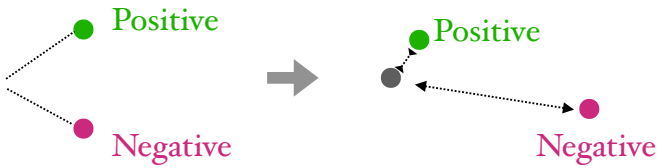
  e.g., SBERT, SGPT, Qwen Embeddings

Karpukhin et al 2020. Dense Passage Retrieval for Open-Domain Question Answering.
Muennighoff 2022. SGPT: GPT Sentence Embeddings for Semantic Search.

Training Dense Retrievers



Contrastive learning



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

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

Fast Nearest Neighbor Search

Method	Class name	index_factory	Main parameters	Bytes/vector	Exhausti
Exact Search for L2	IndexFlatL2	"Flat"	d	4*d	yes
Exact Search for Inner Product	IndexFlatIP	"Flat"	d	4*d	yes
Hierarchical Navigable Small World graph exploration	IndexHNSWFlat	"HNSW,Flat"	d, M	4*d + x * M * 2 * 4	no
Inverted file with exact post-verification	IndexIVFFlat	"IVFx,Flat"	quantizer, d, nlists, metric	4*d + 8	no
Locality-Sensitive Hashing (binary flat index)	IndexLSH	-	d, nbits	ceil(nbits/8)	yes
Scalar quantizer (SQ) in flat mode	IndexScalarQuantizer	"SQ8"	d	d	yes
Product quantizer (PQ) in flat mode	IndexPQ	"PQx", "PQ"M"x"nbits	d, M, nbits	ceil(M * nbits / 8)	yes
IVF and scalar quantizer	IndexIVFScalarQuantizer	"IVFx,SQ4" "IVFx,SQ8"	quantizer, d, nlists, qtype	SQfp16: 2 * d + 8, SQ8: d + 8 or SQ4: d/2 + 8	no
IVFADC (coarse quantizer+PQ on residuals)	IndexIVFPQ	"IVFx,PQ"y"x"nbits	quantizer, d, nlists, M, nbits	ceil(M * nbits/8)+8	no

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

Exact search (still fast for 10^6-10^7 scale)

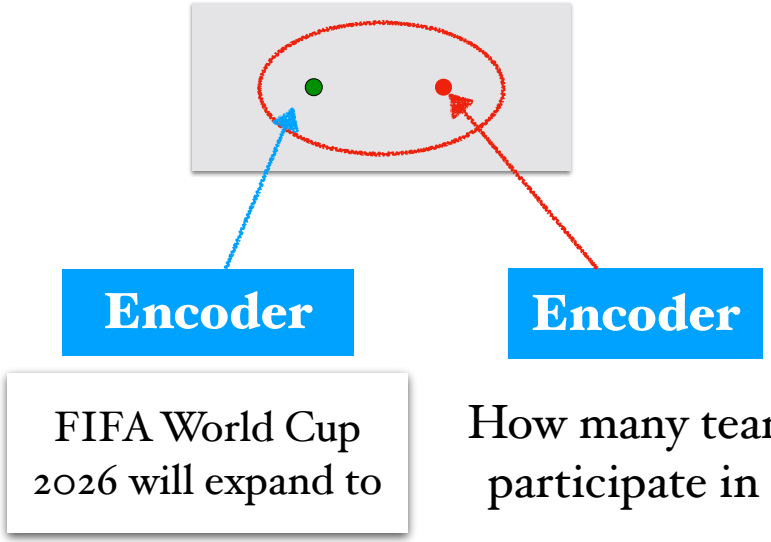
Approximate search (faster but more memory)

Reduce index size with quantization

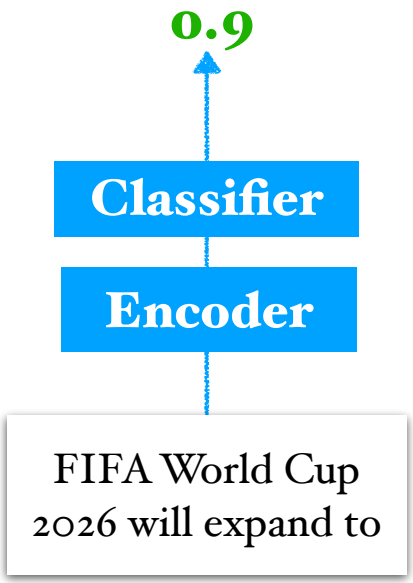
https://speakerdeck.com/matsui_528/cvpr20-tutorial-billion-scale-approximate-nearest-neighbor-search (CVPR 2020 Tutorial)

Reranking with Cross Encoders

Bi-Encoder



Cross-Encoder



Evaluation Metrics

Evaluation of **unranked** retrieval sets

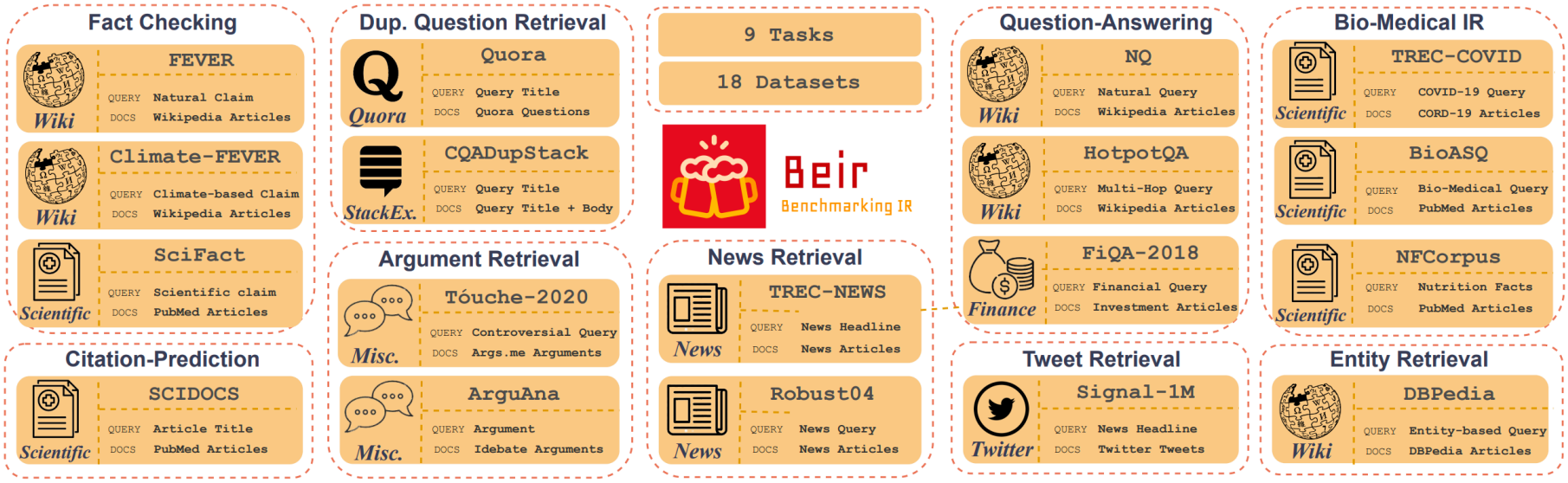
$$\text{Precision} = \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} \qquad \text{Recall} = \frac{\#(\text{relevant items retrieved})}{\#(\text{relevant items})}$$

Evaluation of **ranked** retrieval sets

$$\text{MAP}(Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_j}\sum_{k=1}^{m_j}\text{Precision}(R_{jk}) \qquad \text{NDCG}(Q,k) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}Z_{kj}\sum_{m=1}^{k}\frac{2^{R(j,m)}-1}{\log_2(1+m)}$$

nDCG@10 is widely used (e.g., BEIR)

Retrieval Benchmark: MTEB



Thakur et al. 2021. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models.

BEIR Results

	BM25	BM25+CE	DPR	Ours	Ours+CE
MS MARCO	22.8	41.3	17.7	40.7	**47.0**
Trec-COVID	65.6	**75.7**	33.2	59.6	70.1
NFCorpus	32.5	**35.0**	18.9	32.8	34.4
NQ	32.9	53.3	47.4	49.8	**57.7**
HotpotQA	60.3	70.7	39.1	63.8	**71.5**
FiQA	23.6	34.7	11.2	32.9	**36.7**
ArguAna	31.5	31.1	17.5	44.6	41.3
Touche-2020	**36.7**	27.1	13.1	23.0	29.8
CQADupStack	29.9	37.0.	15.3	34.5	**37.7**
Quora	78.9	82.5	24.8	**86.5**	82.4
DBPedia	31.3	40.9	26.3	41.3	**47.1**
Scidocs	15.8	16.6	7.7	16.5	**17.1**
FEVER	75.3	**81.9**	56.2	75.8	**81.9**
Climate-FEVER	21.3	25.3	14.8	23.7	**25.8**
Scifact	66.5	68.8	31.8	67.7	69.2
Avg. w/o CQA	44.0	49.5	26.3	47.5	51.2
Avg.	43.0	48.6	25.5	46.6	50.2
Best on	1	3	0	1	9

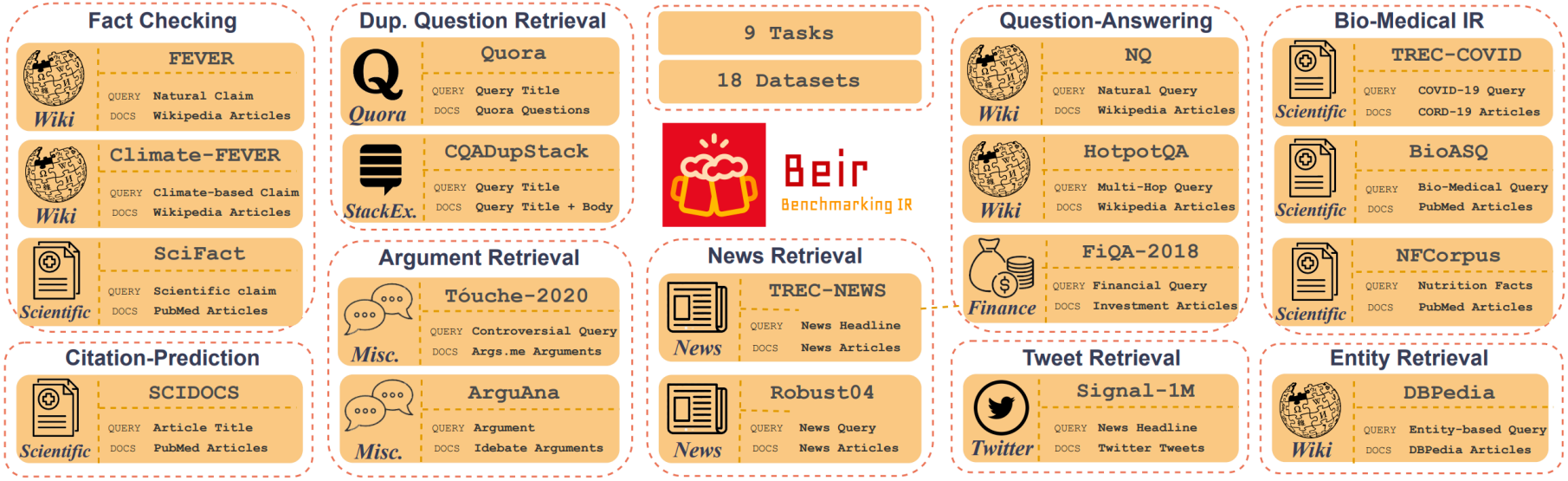
Adding CE (cross-encoder) helps

Dense retrievers could struggle in OOD

Unsupervised training helps in OOD

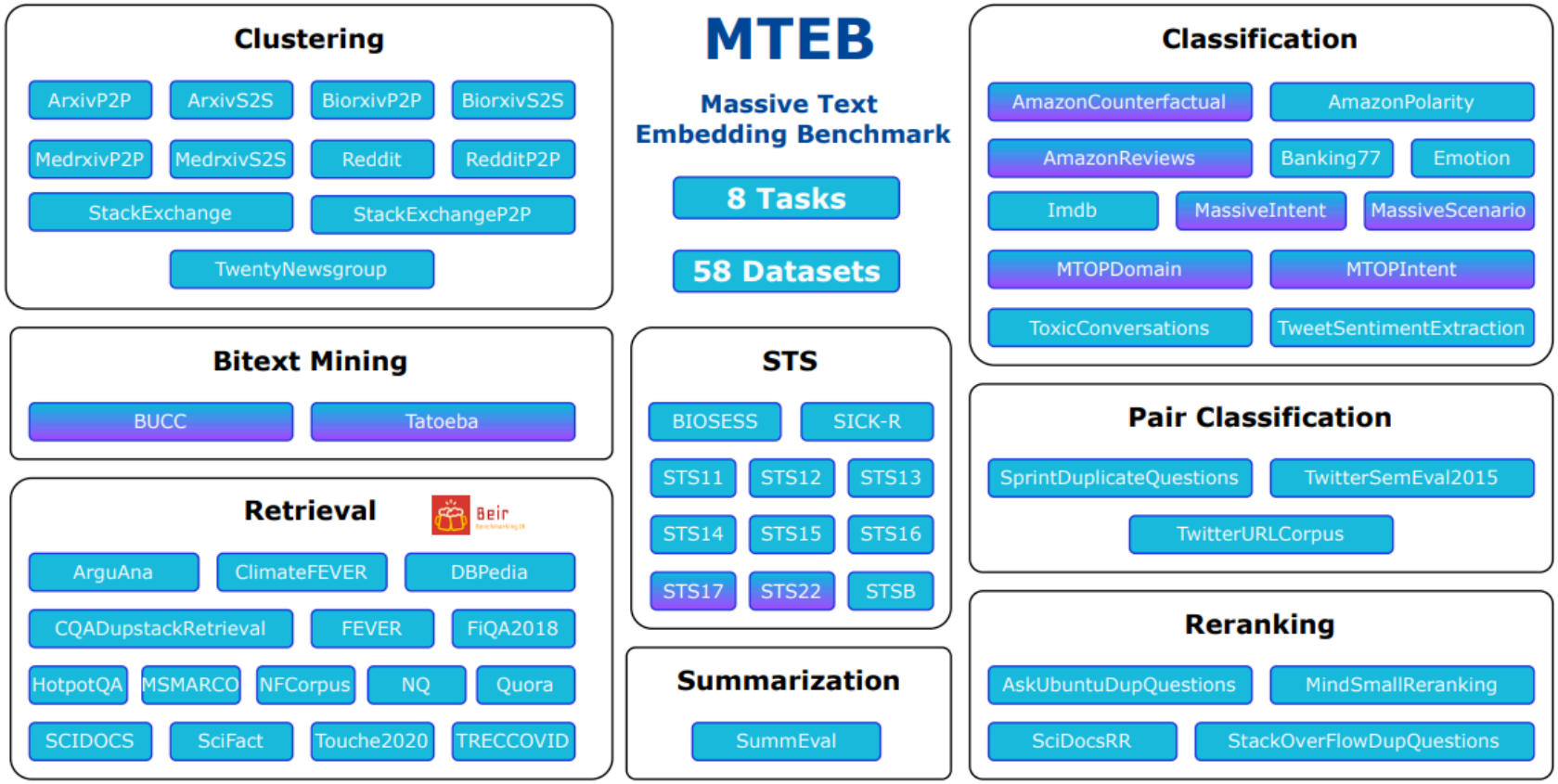
Izacard et al. TMLR 2022. Unsupervised Dense Information Retrieval with Contrastive Learning.

Retrieval Benchmark: MTEB



Thakur et al. 2021. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models.
Muennighoff et al. 2022. MTEB: Massive Text Embedding Benchmark.

Retrieval Benchmark: MTEB



Thakur et al. 2021. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models.
Muennighoff et al. 2022. MTEB: Massive Text Embedding Benchmark.

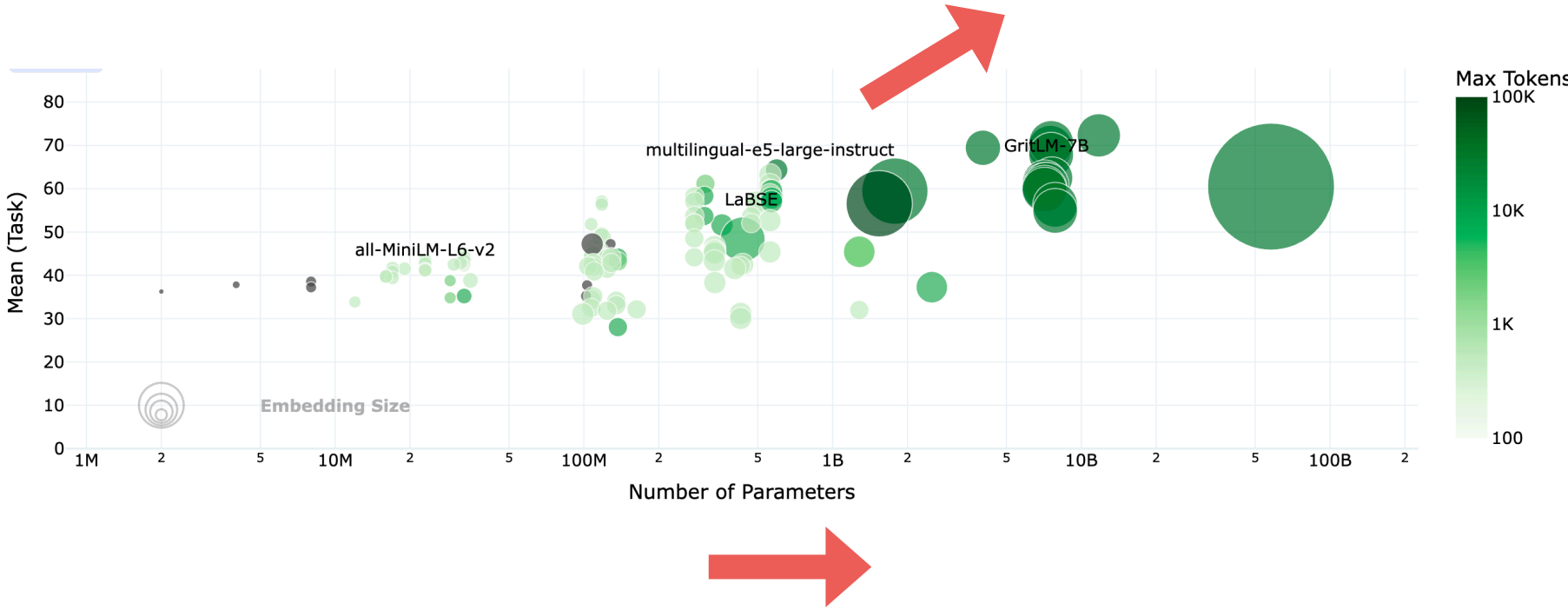
MTEB Leaderboard

Rank (Bor…	Model	Zero-shot	Memory Us…	Number of P…	Embedding D…	Max Tokens
1	KaLM-Embedding-Gemma3-12B-2511	73%	44884	11.8	3840	32768
2	llama-embed-nemotron-8b	99%	28629	7.5	4096	32768
3	Qwen3-Embedding-8B	99%	14433	7.6	4096	32768
4	gemini-embedding-001	99%			3072	2048
5	Qwen3-Embedding-4B	99%	7671	4.0	2560	32768
6	Octen-Embedding-8B	99%	14433	7.6	4096	32768
7	Seed1.6-embedding-1215	89%			2048	32768
8	Qwen3-Embedding-0.6B	99%	1136	0.596	1024	32768
9	gte-Qwen2-7B-instruct	⚠ NA	29040	7.6	3584	32768
10	Linq-Embed-Mistral	99%	13563	7.1	4096	32768
11	multilingual-e5-large-instruct	99%	1068	0.560	1024	514

https://huggingface.co/spaces/mteb/leaderboard

Muennighoff et al. 2022. MTEB: Massive Text Embedding Benchmark.

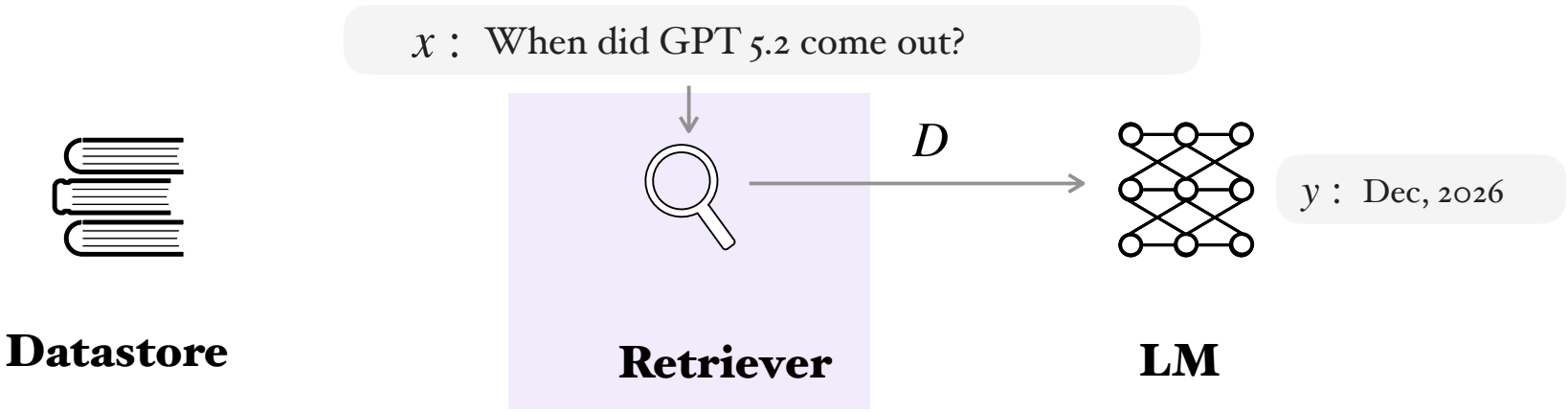
MTEB Leaderboard



https://huggingface.co/spaces/mteb/leaderboard

Muennighoff et al. 2022. MTEB: Massive Text Embedding Benchmark.

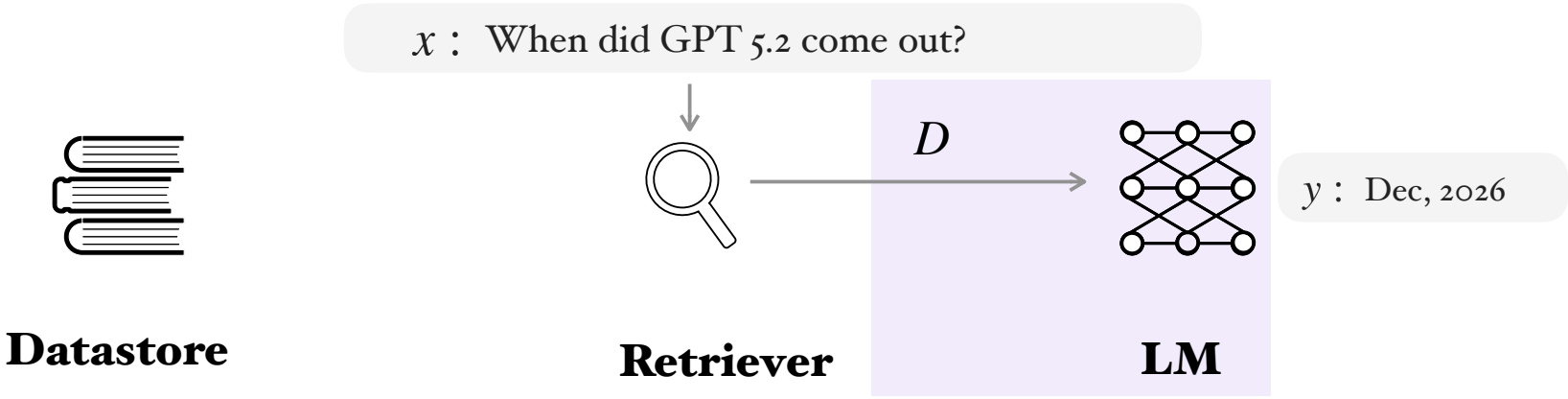
Summary of Part 2



- ✓ Types of retrievers
- ✓ Training
- ✓ Evaluations

- Different types pf retrievers
- Training with contrastive loss
- Common metrics: NDCG@10 ... etc
- Performance v.s. cost trade off

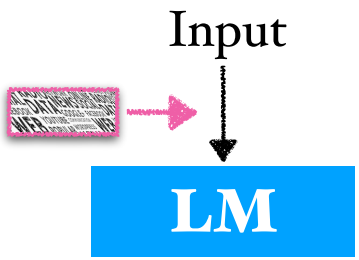
Key Factors & Design Choices



- ✓ Architectures
- ✓ Training
- ✓ Inference

How to Use Retrieval

Input Augmentation



- Augment input of LMs
- Easy to apply (w/o training) & effective
- Difficulty of using many D

e.g., RAG

Intermediate Fusion

- Modify LMs to incorporate D in intermediate layers
- Scalable to many passages
- Requires retraining

e.g., RETRO, InstructRETRO

Output Interpolation

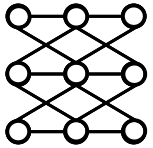
- Directly manipulate output token distributions
- No training required*
- Limited effectiveness on tasks

e.g., kNNLM

RAG (Lewis et al., 2020)

x : When did GPT 5.2 come out?





Datastore

Retriever

LM

$D \in \text{Top}_k \text{Sim}(\cdot \mid x)$

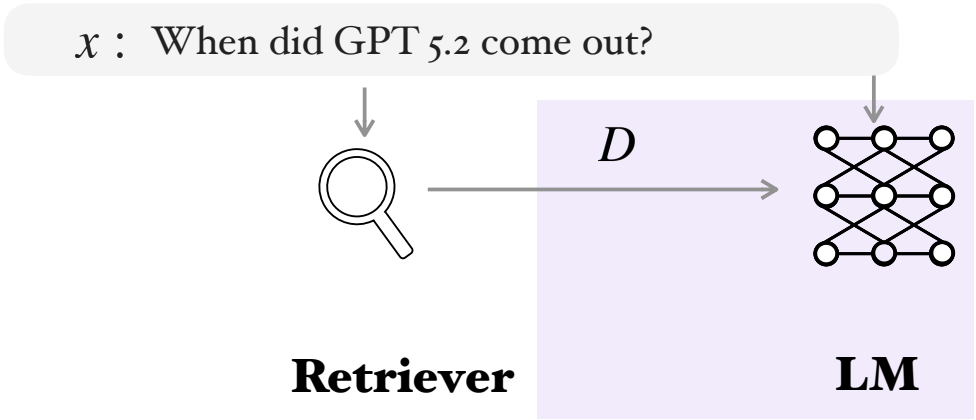
Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for

0.9

GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was

0.7

RAG (Lewis et al., 2020)



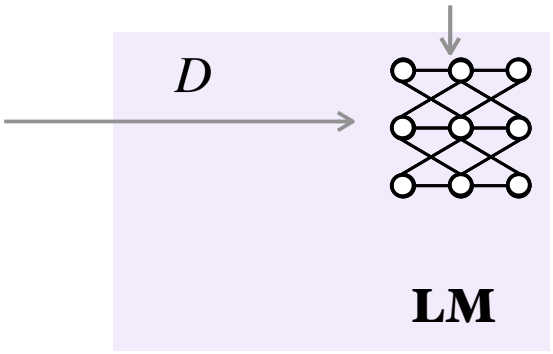
$D \in \text{Top}_k\text{Sim}(\cdot \mid x)$

Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for — 0.9

GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was — 0.7

RAG (Lewis et al., 2020)

x : When did GPT 5.2 come out?



$D \in \text{Top}_k \text{Sim}(\cdot \mid x)$

Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for

GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was

Question: When did GPT 5.2 come out?

References:

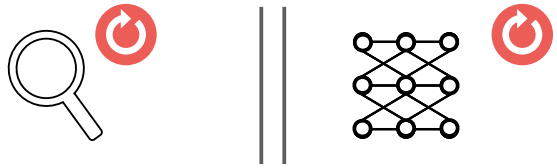
Dec 11, 2025 - Open AIL GPT-5.2 Instant is a fast, capable workhorse for

GPT-5 (Wikipedia) Preceded in the series by GPT-4, it was

y : Dec, 2026

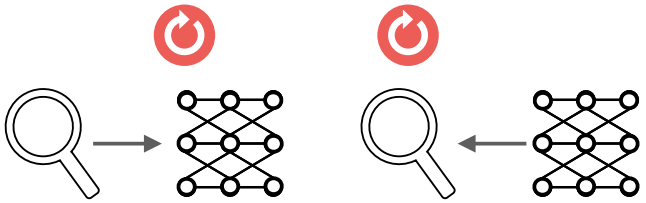
Lewis et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.

Training RAG



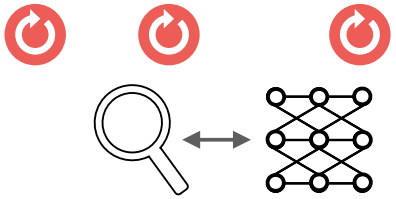
Independent training

- DPR (Karpukhin et al., 2020)
- DRQA (Chen et al., 2017)



Sequential training

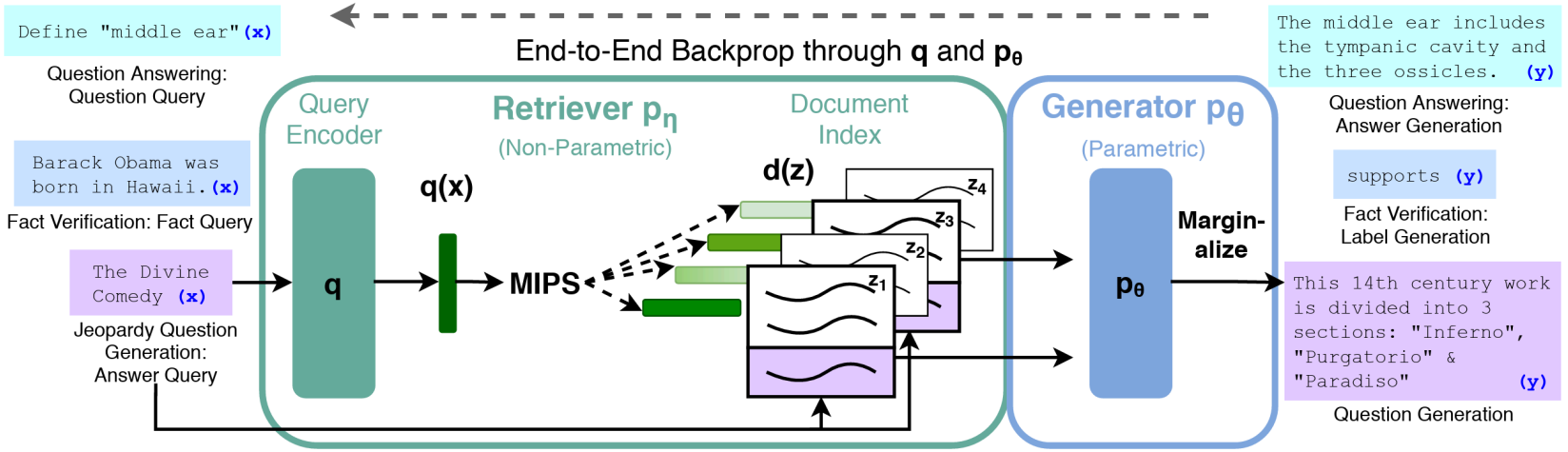
- Evidentiality Generator (Asai et al., 2023)
- REPLUG (Shi et al., 2023)



Joint training

- RAG (Lewis et al., 2021)
- REALM (Guu et al., 2021)

End-to-end Training for RAG

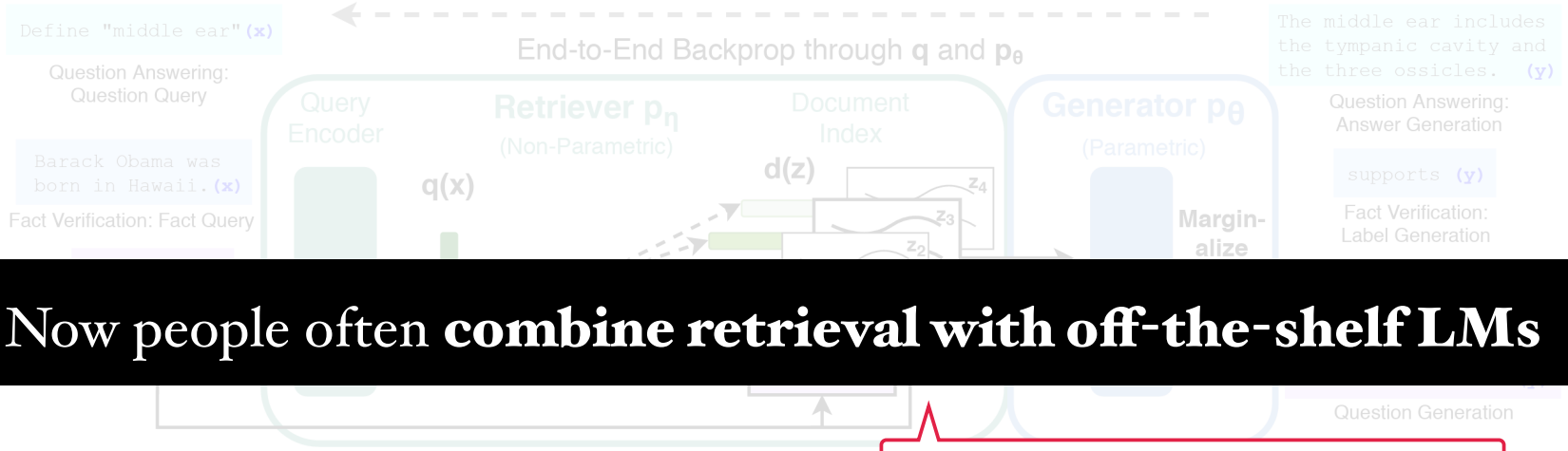


$$p_{\text{RAG}} \approx \prod_{i} \sum_{z \in \text{top-k}(p(\cdot|x))} p_\eta(z|x) p_\theta(y_i|x, z, y_{1:i-1})$$

Retriever score Generator score

Lewis et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.

End-to-end Training for RAG



Now people often combine retrieval with off-the-shelf LMs

Widely referred to as ***RAG***

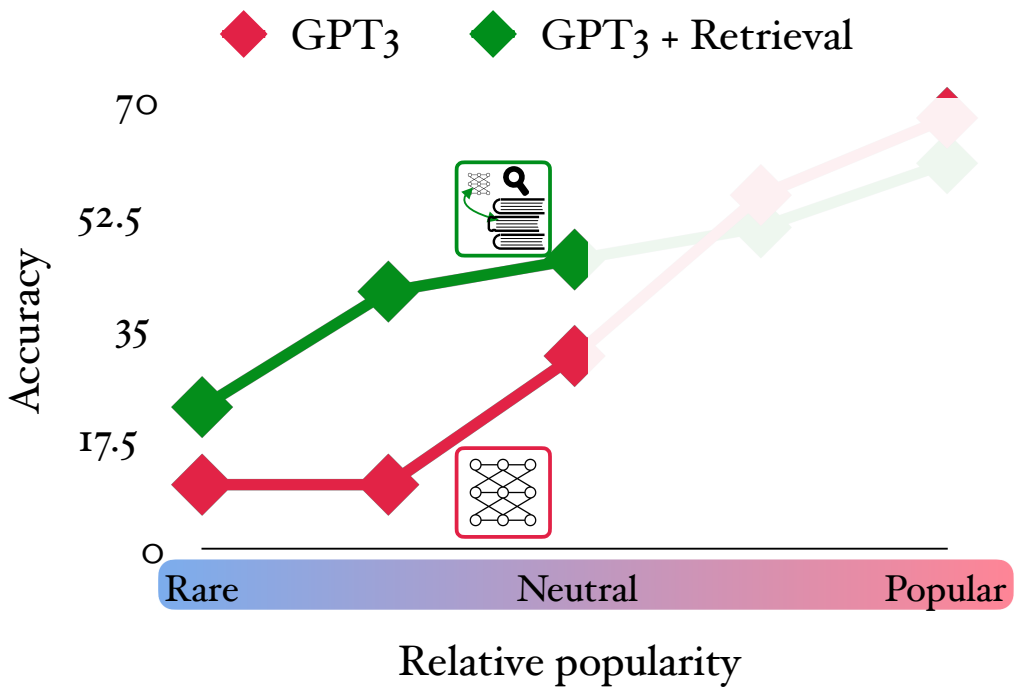
$$p_{\text{RAG}} \approx \prod_i \sum_{z \in \text{top-k}(p(\cdot|x))} p_\eta(z|x) p_\theta(y_i|x, z, y_{1:i-1})$$

Retriever score Generator score

Lewis et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.

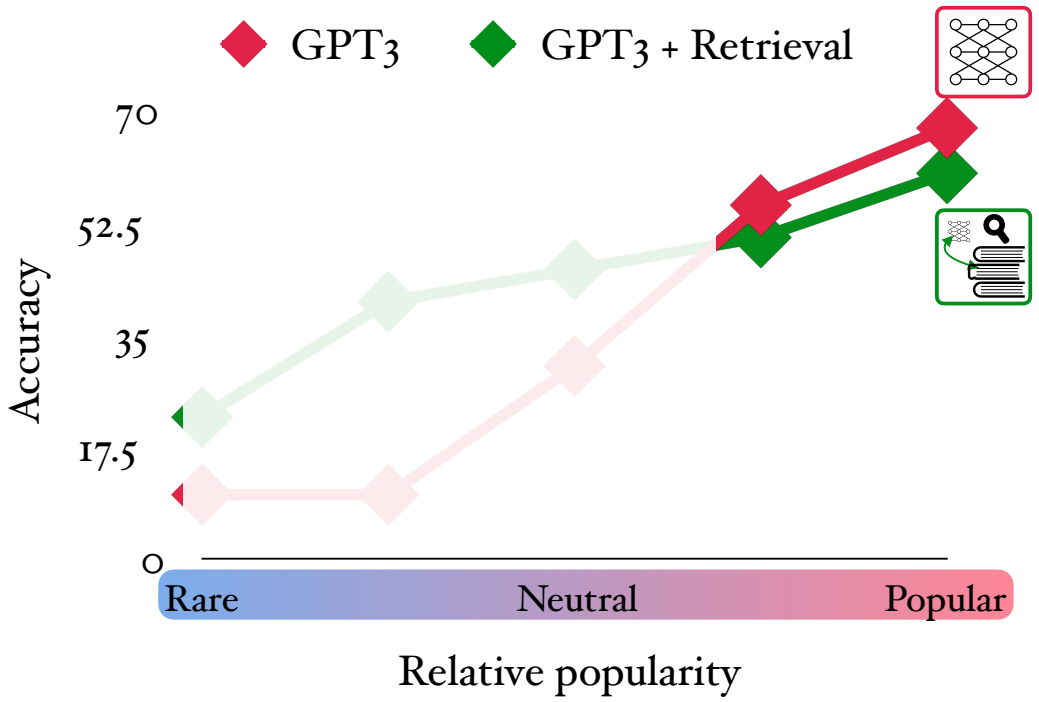
Effectiveness of In-context RAG

RAG constantly gives performance improvements in long-tail



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

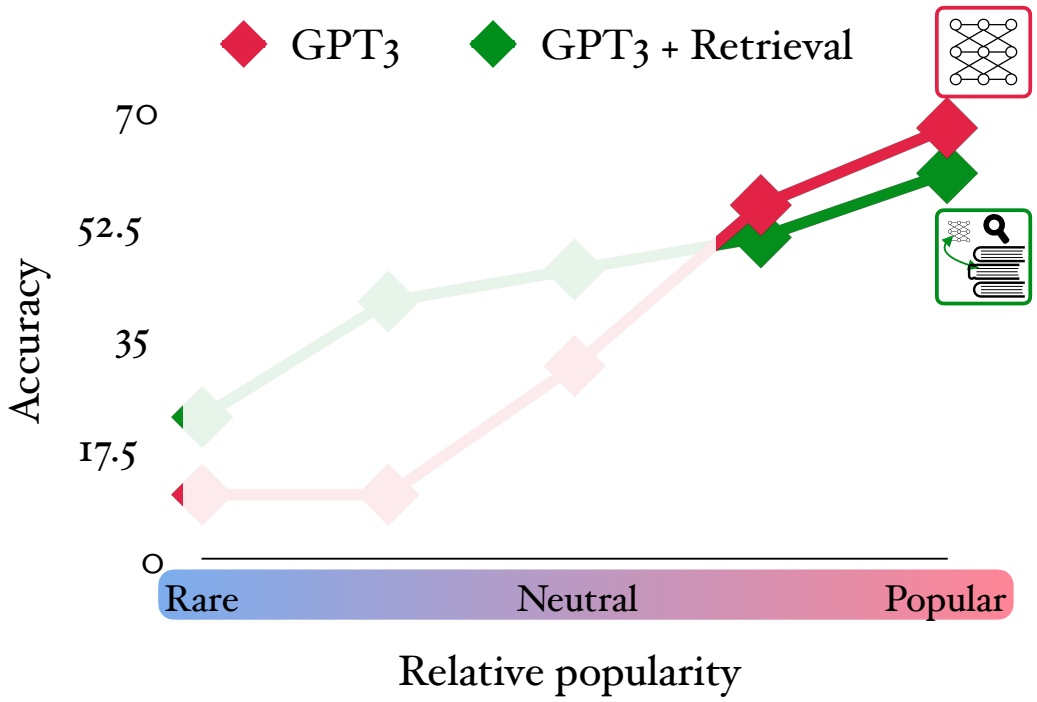
Limitations of In-context RAG

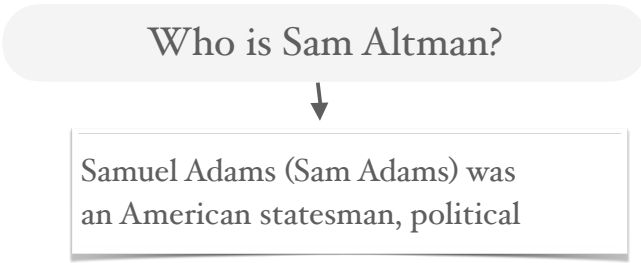


Who is Sam Altman?

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

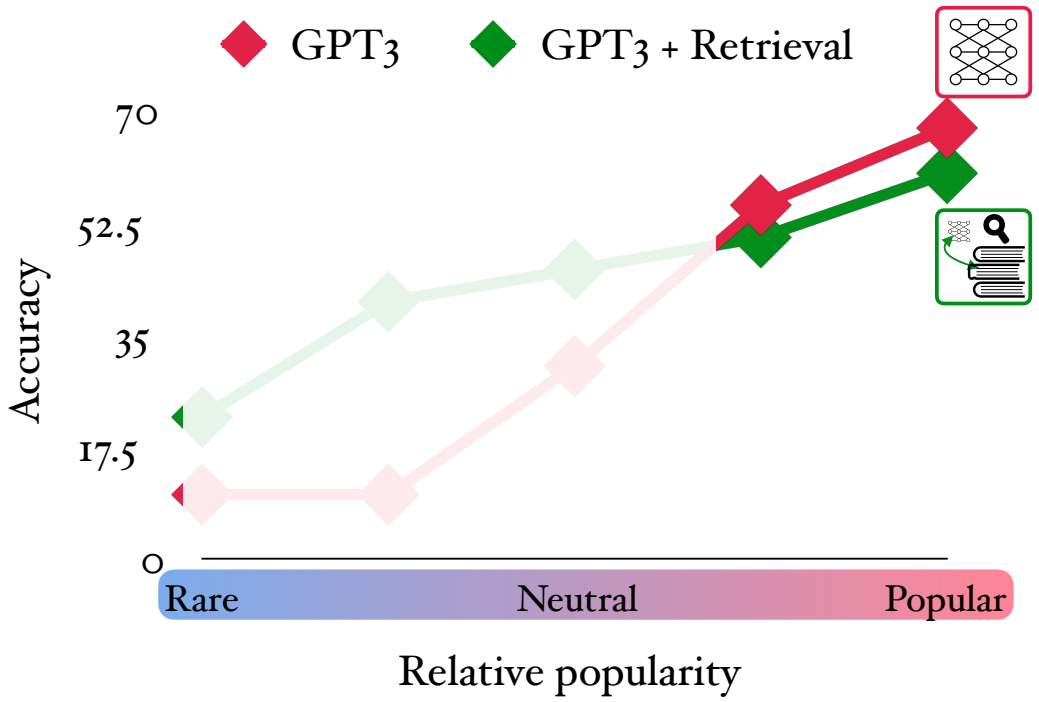
Limitations of In-context RAG

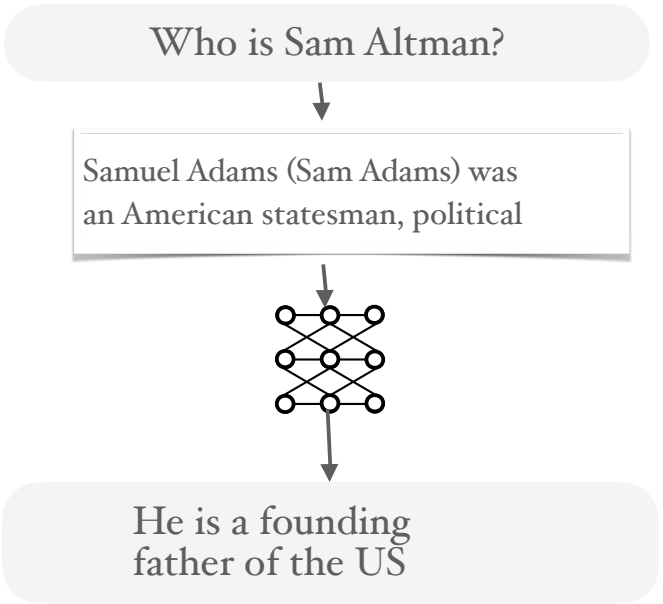




Mallen*, Asai* (contributed equally) et al. 2023. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories

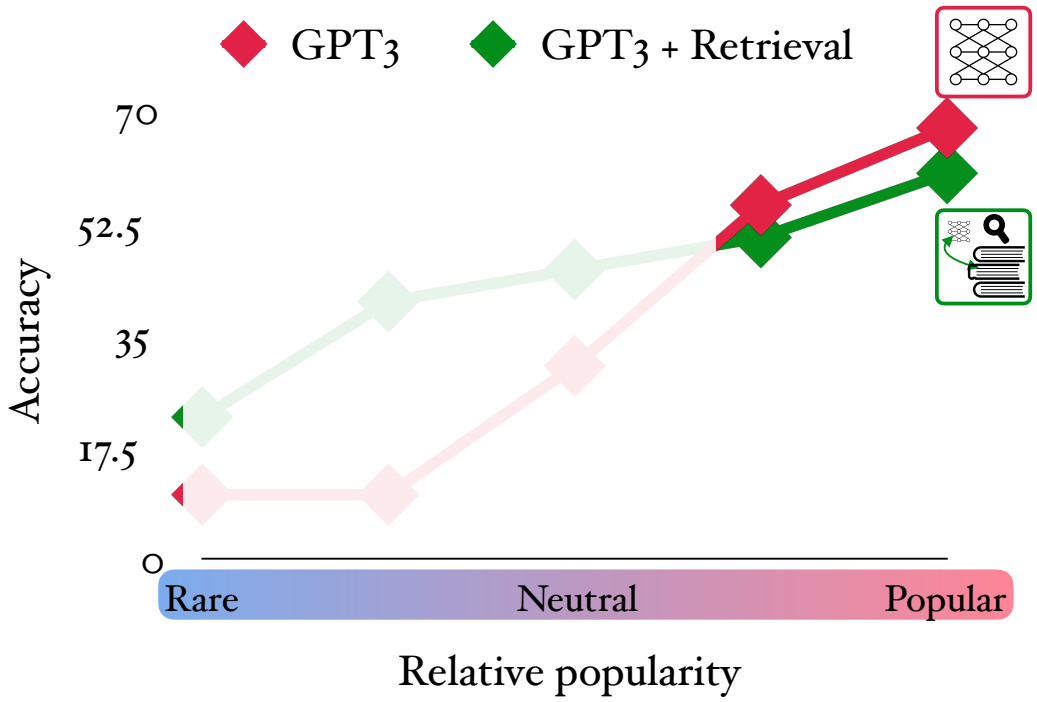
Limitations of In-context RAG

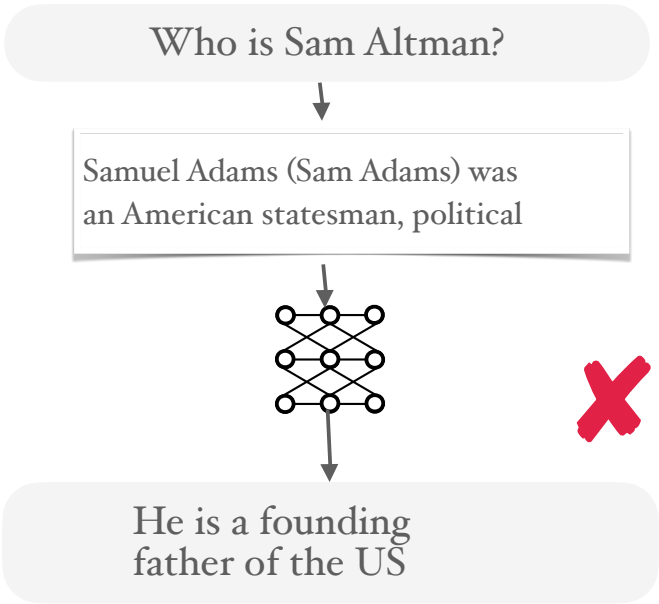




Mallen*, Asai* (contributed equally) et al. 2023. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories

Limitations of In-context RAG

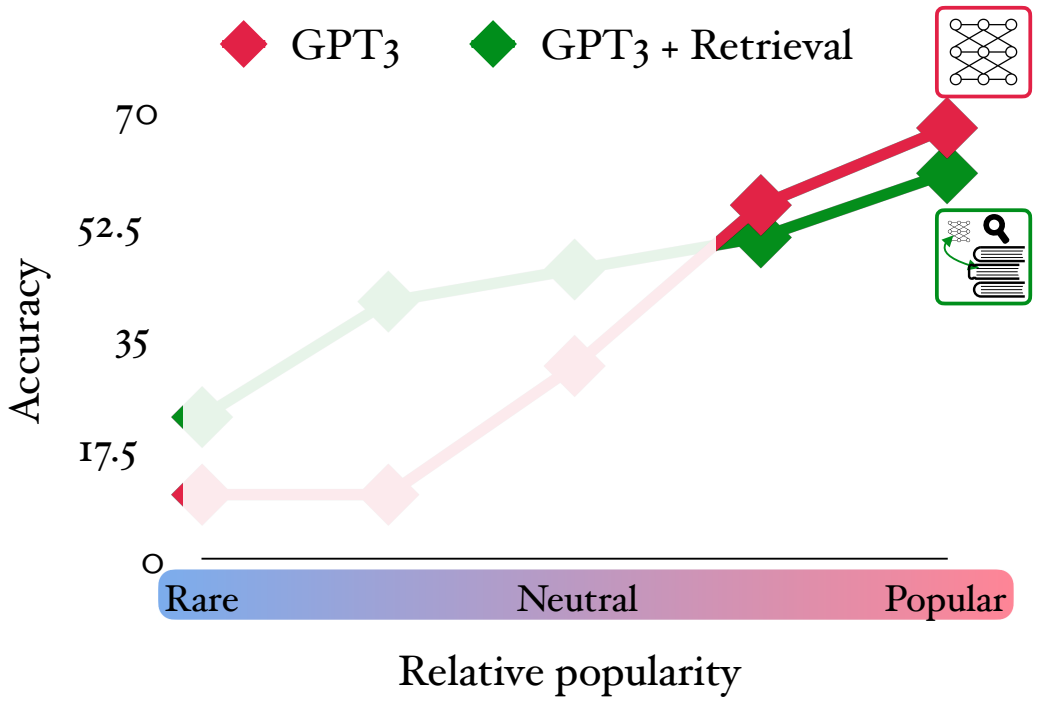


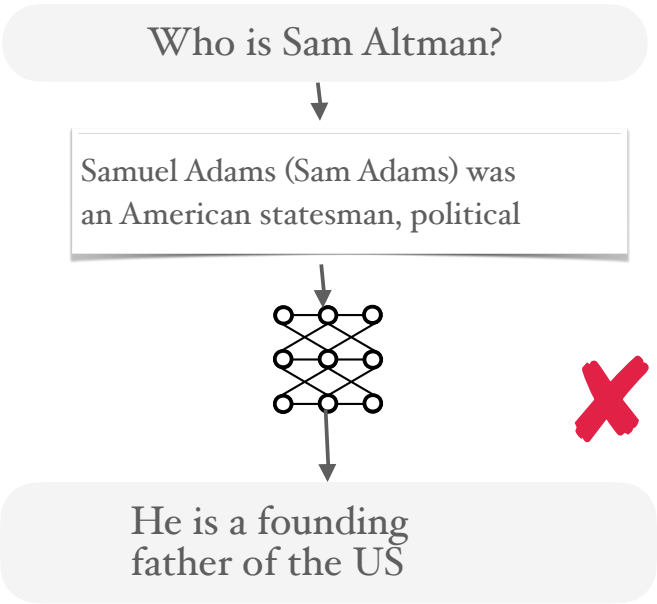


Mallen*, Asai* (contributed equally) et al. 2023. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories

Limitations of In-context RAG

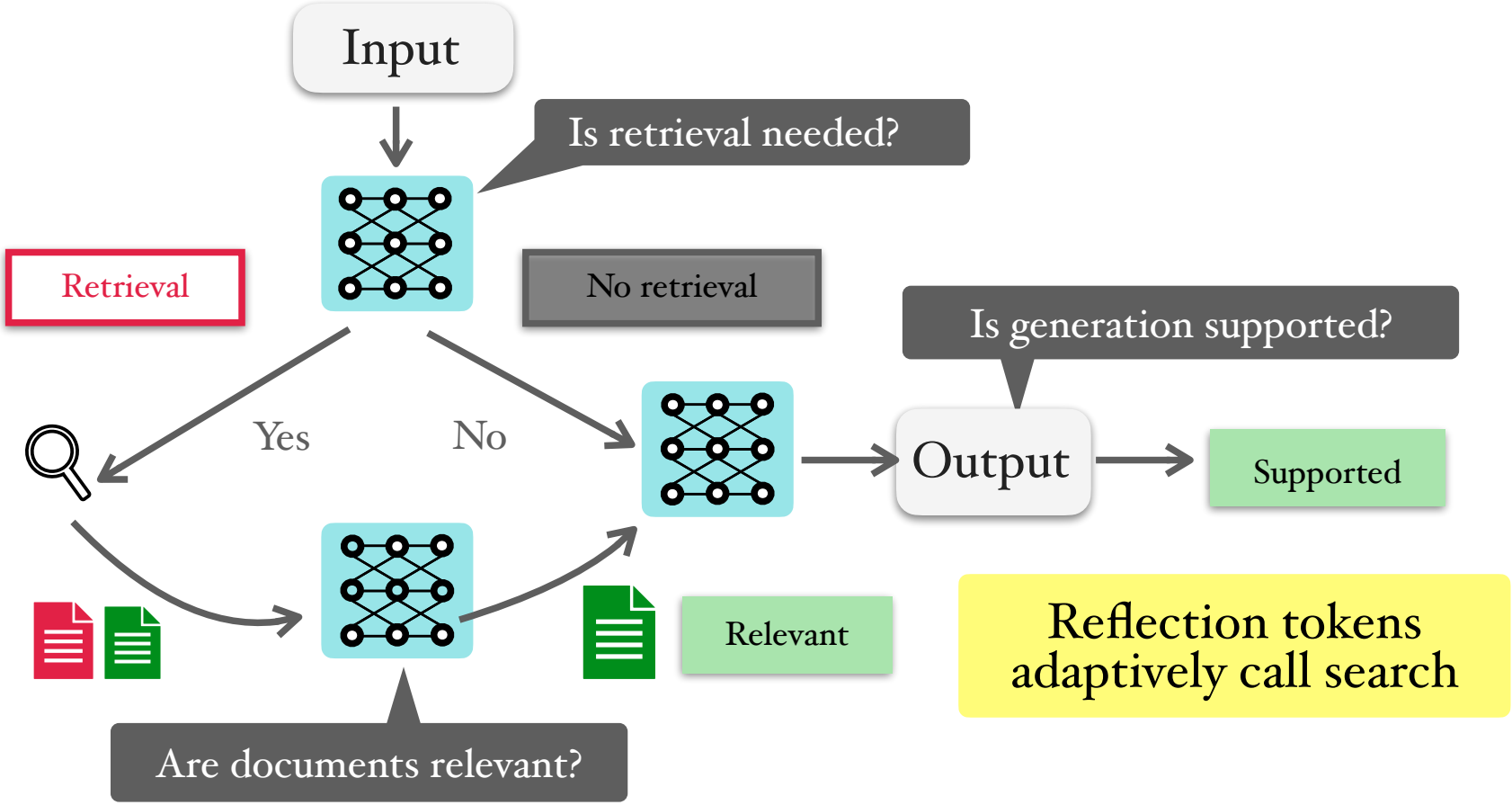
Inflexibility and lack of robustness to unhelpful docs





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

Self-RAG: Adaptive Retrieval



Asai et al. 2024..Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection.

Tool-augmented LMs

- Training LMs to *adaptively* and *iteratively* use external tools at inference time
- LMs can use diverse set of tools, not just retrieval

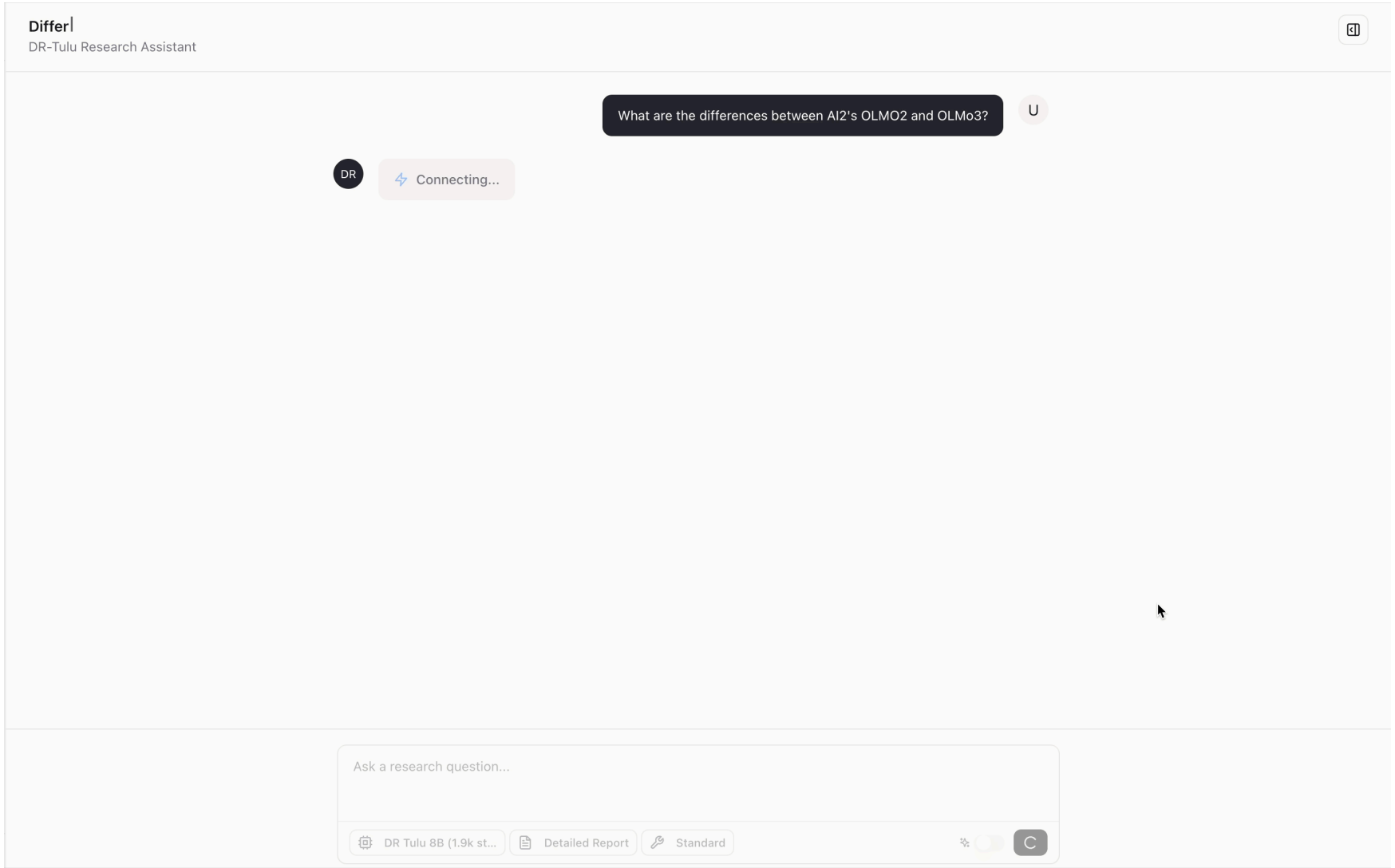
The Brown Act is California's law **[WikiSearch("Brown Act") → The Ralph M. Brown Act is an act of the California State Legislature that guarantees the public's right to attend and participate in meetings of local legislative bodies.]** that requires legislative bodies, like city councils, to hold their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

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

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

Schick et al. 2023. Toolformer: Language Models Can Teach Themselves to Use Tools.

Deep Research (DR) Agents

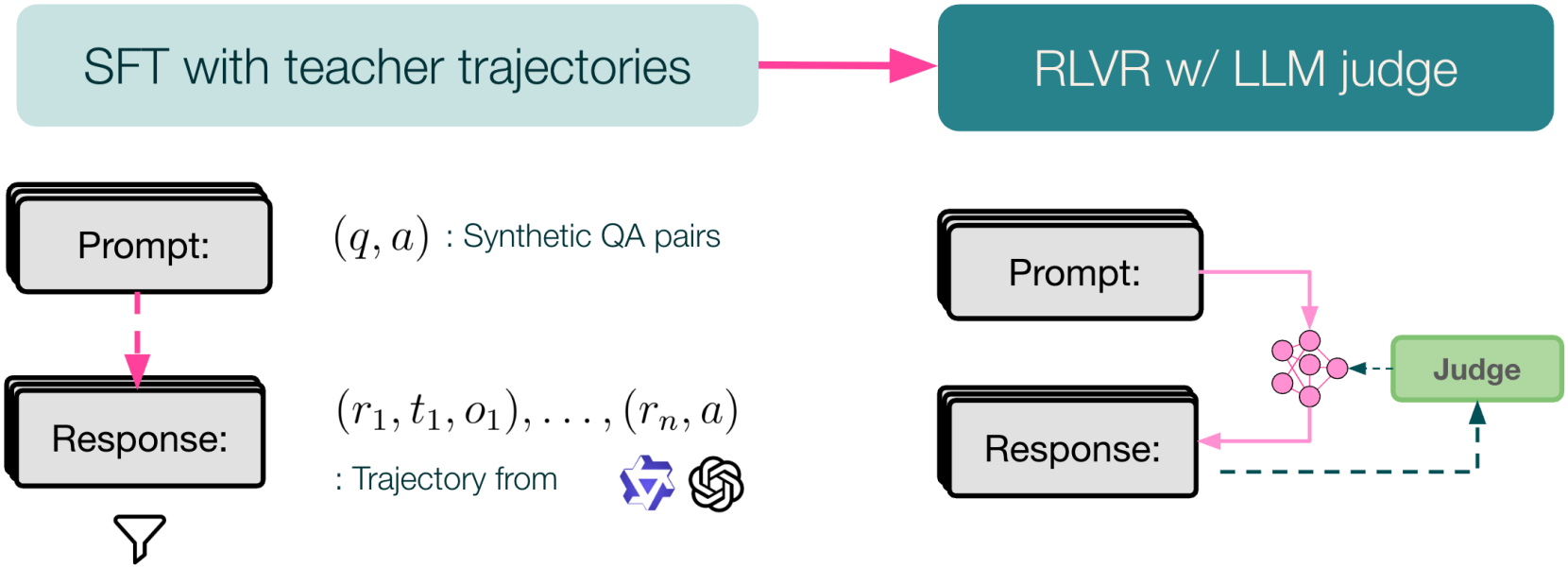


https://www.dr-tulu.org/chat

Shao*, Asai* et al. 2025. DR Tulu: Reinforcement Learning with Evolving Rubrics for Deep Research.

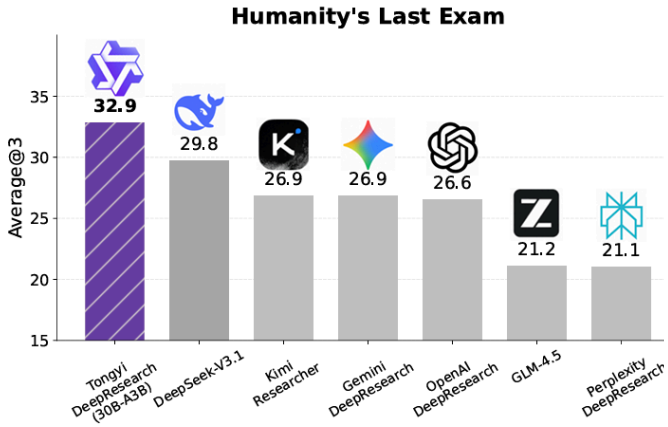
Training for DR Agents

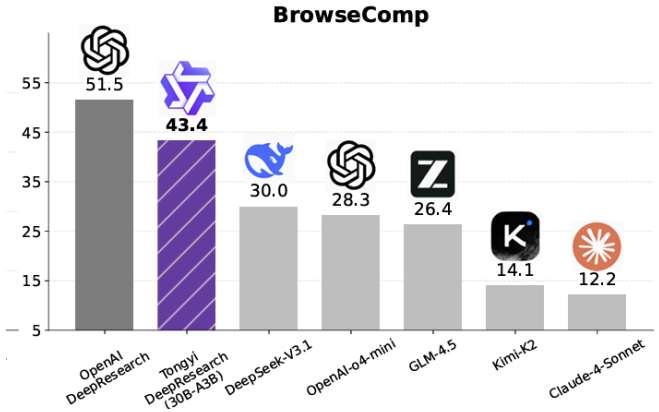
- Large-scale SFT (w. Rejection sampling) followed by RLVR using answer matching as reward



Tongyi Research. 2025. Tongyi Deep Research Technical Report.

Training for DR Agents





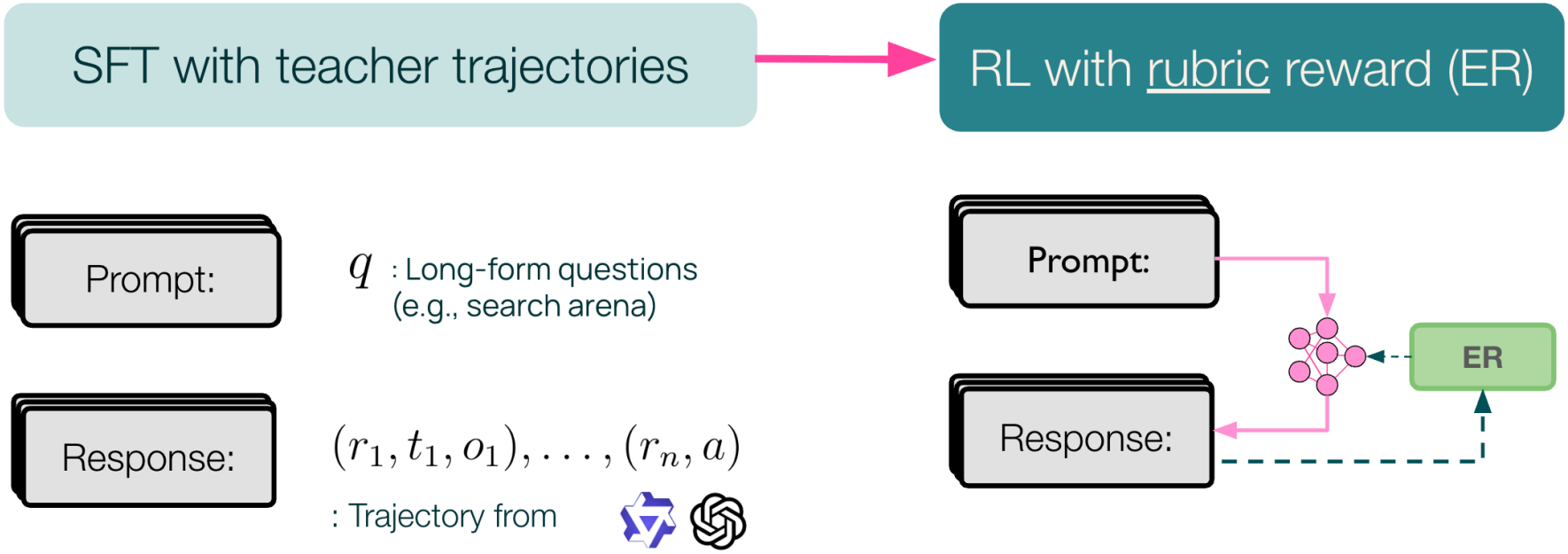
Deep Research Bench (Long-form DR)

Rank	model	overall
1	🚀 cellcog	54.54
2	🚀 Qianfan-DeepResearch Pro	54.22
3	🚀 Qianfan-DeepResearch	53.02
4	🚀 tavily-research	52.44
5	🚀 thinkdepthai-deepresearch	52.43
6	🚀 salesforce-air-deep-research	50.65
7	🚀 langchain-open-deep-research(GPT-5,with gensee search)	50.6
8	🚀 gemini-2.5-pro-deepresearch	49.71
9	🚀 langchain-open-deep-research(GPT-5,with Tavily)	49.33
10	🚀 openai-deepresearch	46.45
...		
17	🚀 tongyi-deepresearch-30B-A3B	40.46

Tongyi Research. 2025. Tongyi Deep Research Technical Report.

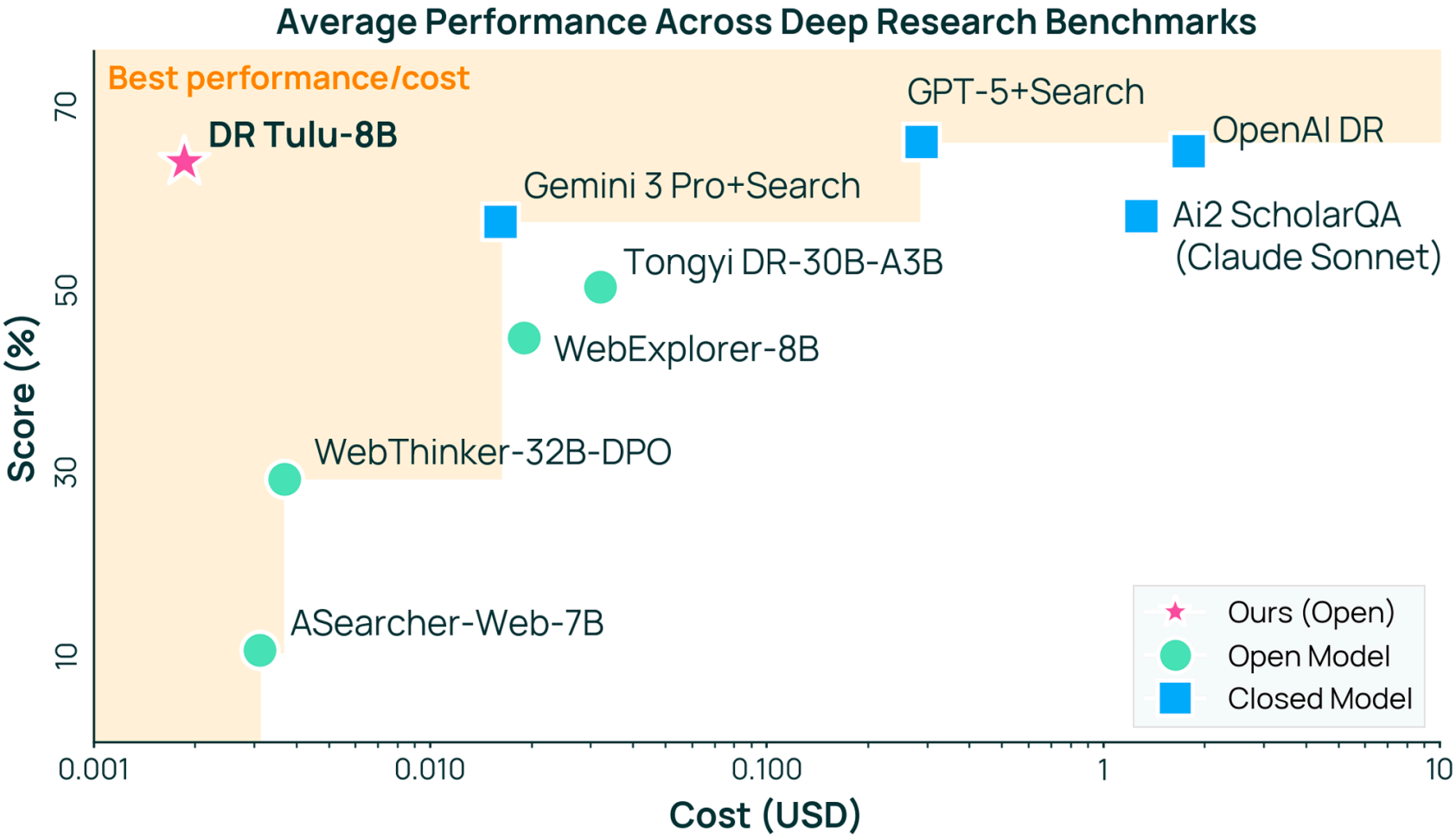
Training for DR Agents with Rubrics

- Long-form responses are "hard-to-verify"
- Rubric-reward based RL for DR agent training



Shao*, Asai* et al. 2025. DR Tulu: Reinforcement Learning with Evolving Rubrics for Deep Research.

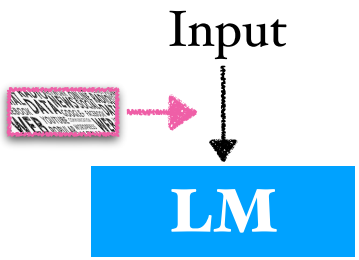
Training for DR Agents with Rubrics



Shao*, Asai* et al. 2025. DR Tulu: Reinforcement Learning with Evolving Rubrics for Deep Research.

How to Use Retrieval

Input Augmentation



Not scalable to many documents (needs context engineering)

Not strictly grounded

- Augment input of LMs
- Easy to apply (w/o training) & effective
- Difficulty of using many D

e.g., RAG

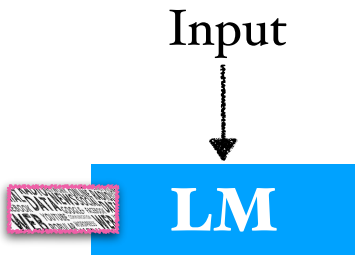
How to Use Retrieval

Input Augmentation

- Augment input of LMs
- Easy to apply (w/o training) & effective
- Difficulty of using many D

e.g., RAG

Intermediate Fusion



- Modify LMs to incorporate D in intermediate layers
- Scalable to many passages
- Requires retraining

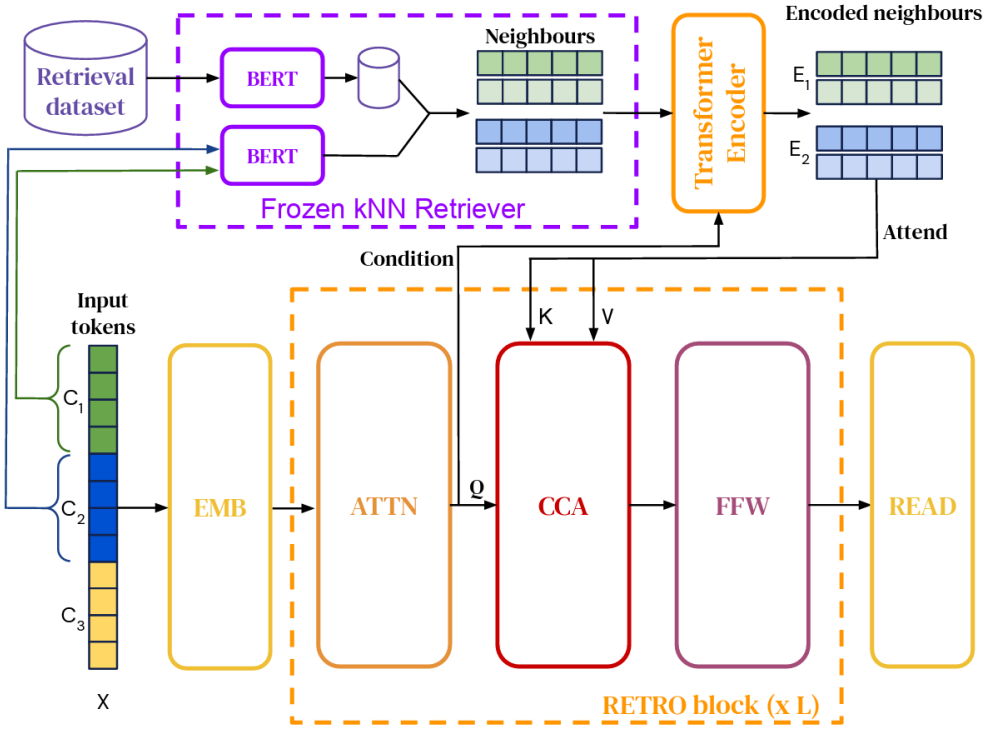
e.g., RETRO, InstructRETRO

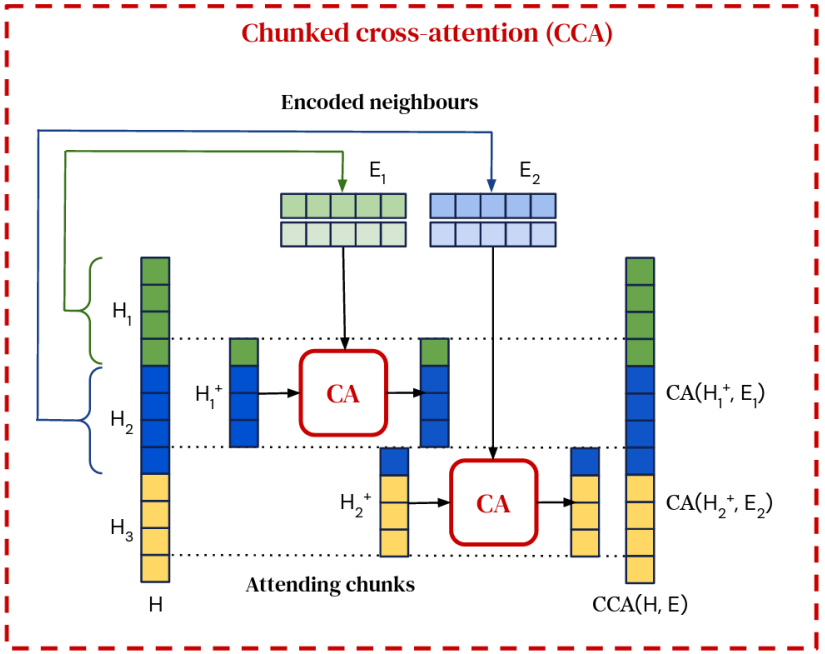
Output Interpolation

- Directly manipulate output token distributions
- No training required*
- Limited effectiveness on tasks

e.g., kNNLM

RETRO





Borgeaud et al. 2022. Improving language models by retrieving from trillions of tokens.

How to Use Retrieval

Input Augmentation

- Augment input of LMs
- Easy to apply (w/o training) & effective
- Difficulty of using many D

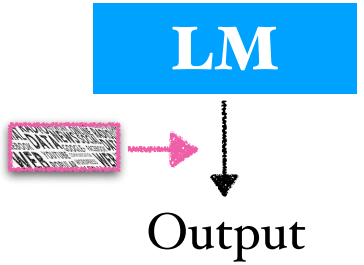
e.g., RAG

Intermediate Fusion

- Modify LMs to incorporate D in intermediate layers
- Scalable to many passages
- Requires retraining

e.g., RETRO, InstructRETRO

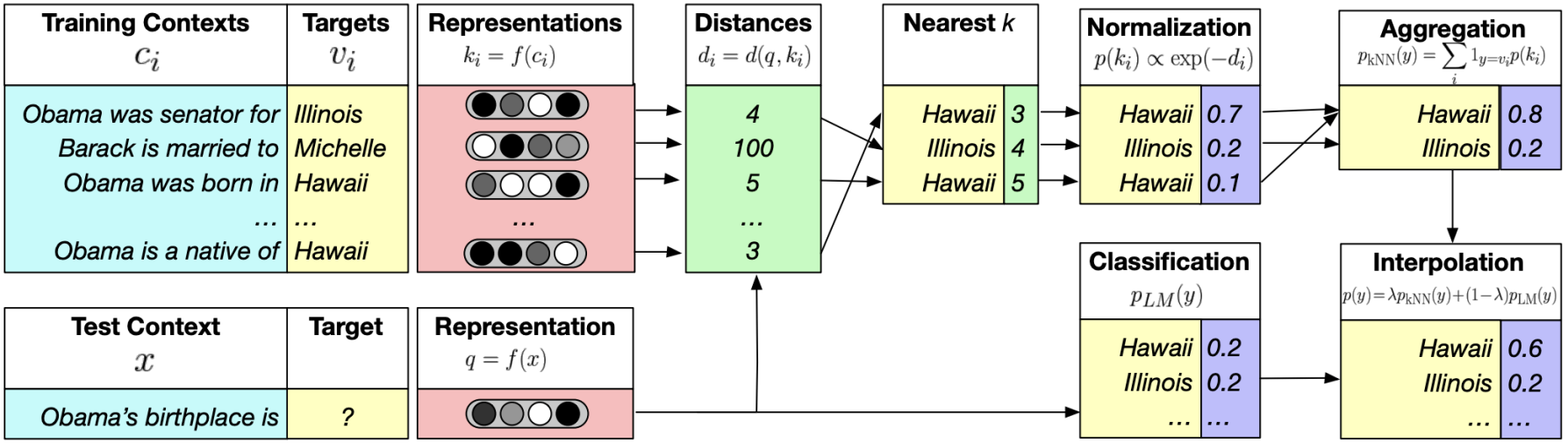
Output Interpolation



- Directly manipulate output token distributions
- No training required*
- Limited effectiveness on tasks

e.g., kNNLM

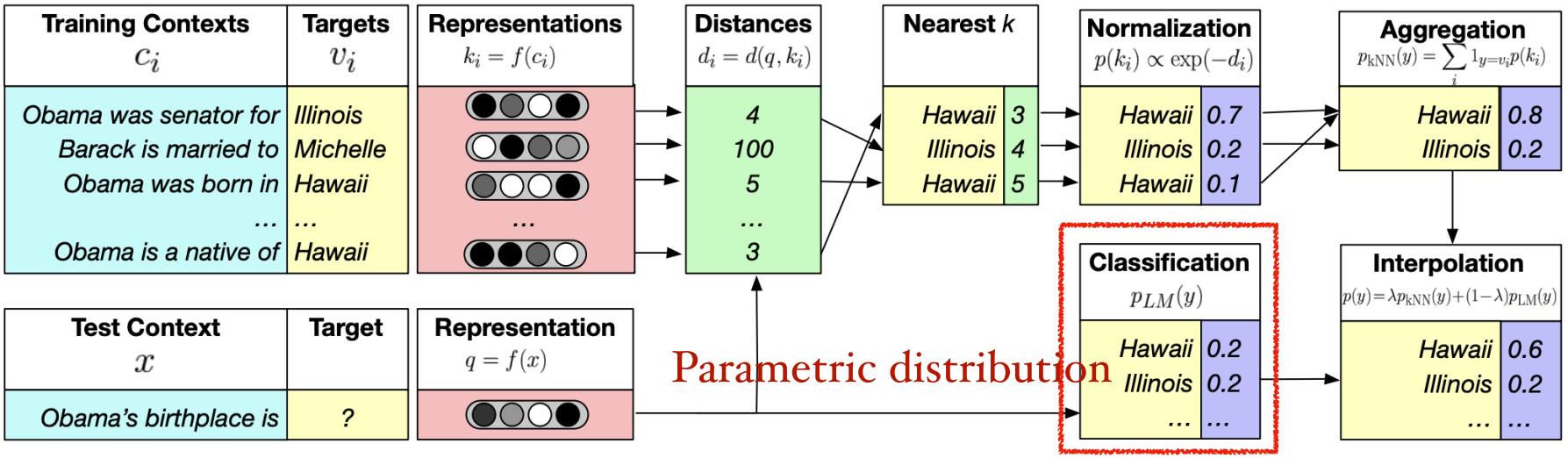
kNN-LM



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

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

kNN-LM

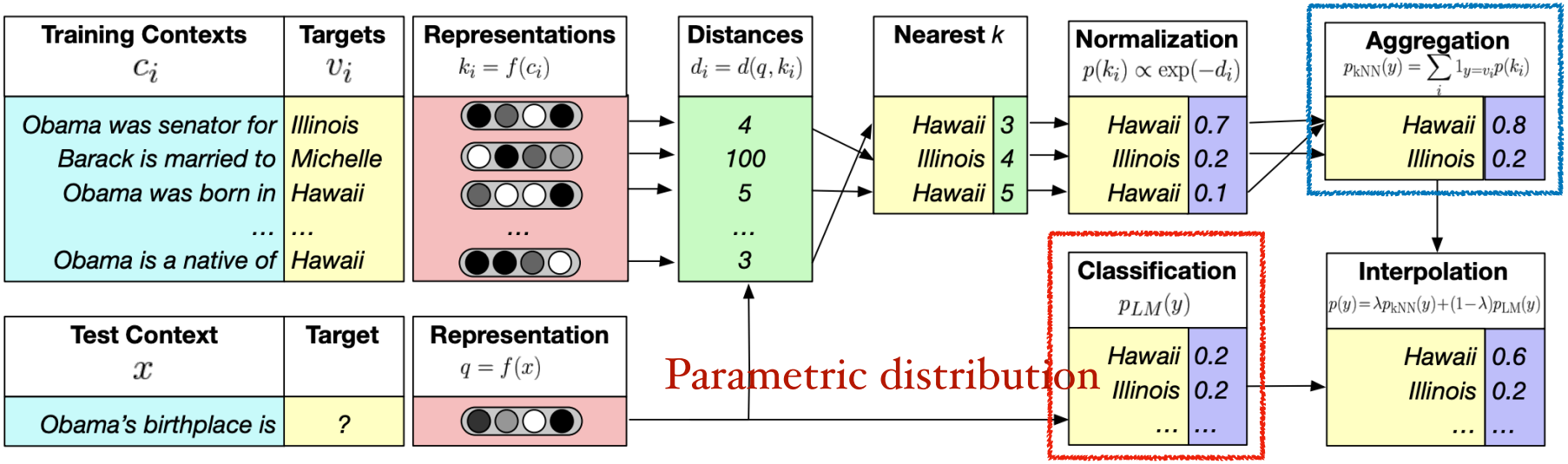


$$P_{k\text{NN-LM}}(y \mid x) = (1 - \lambda) P_{\text{LM}}(y \mid x) + \lambda P_{k\text{NN}}(y \mid x)$$

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

kNN-LM

Nonparametric distributio

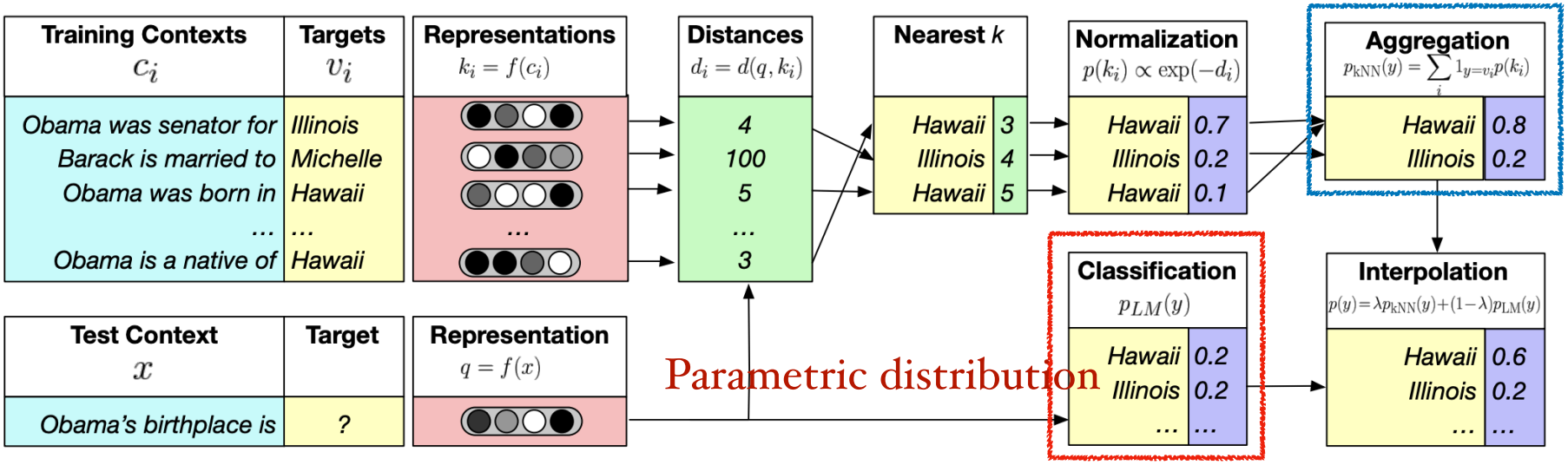


$$P_{k\text{NN-LM}}(y \mid x) = (1 - \lambda) P_{\text{LM}}(y \mid x) + \lambda P_{k\text{NN}}(y \mid x)$$

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

kNN-LM

Nonparametric distributio

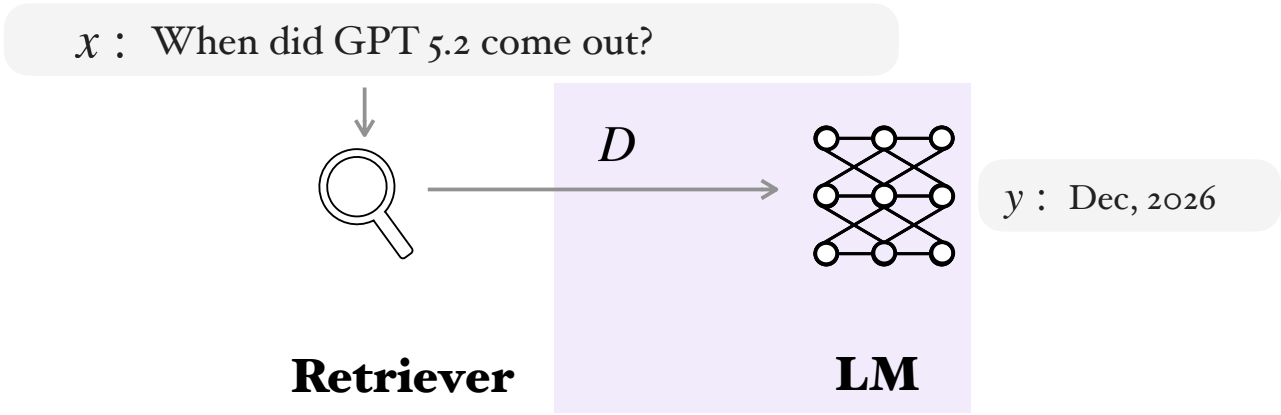


λ: hyperparameter

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

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

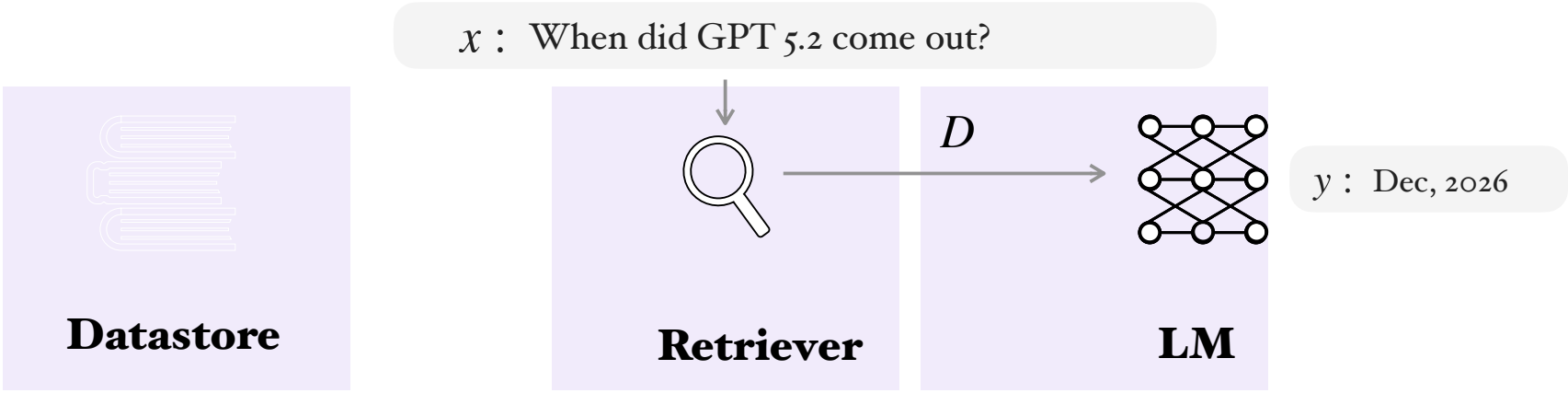
Summary of Part 3



- ✓ Architectures
- ✓ Training
- ✓ Inference

- RAG is widely used but several limitations
- Recent progress to overcome such shortcomings e.g., Deep Research
- Other architectures: intermediate incorporation or output interpolation gain while adding challenges

Retrieval & RAG



- ✓ Sources of datastore
- ✓ Processing
- ✓ Scaling

- ✓ Types of retrievers
- ✓ Training
- ✓ Evaluations

- ✓ Architectures
- ✓ Training
- ✓ Inference