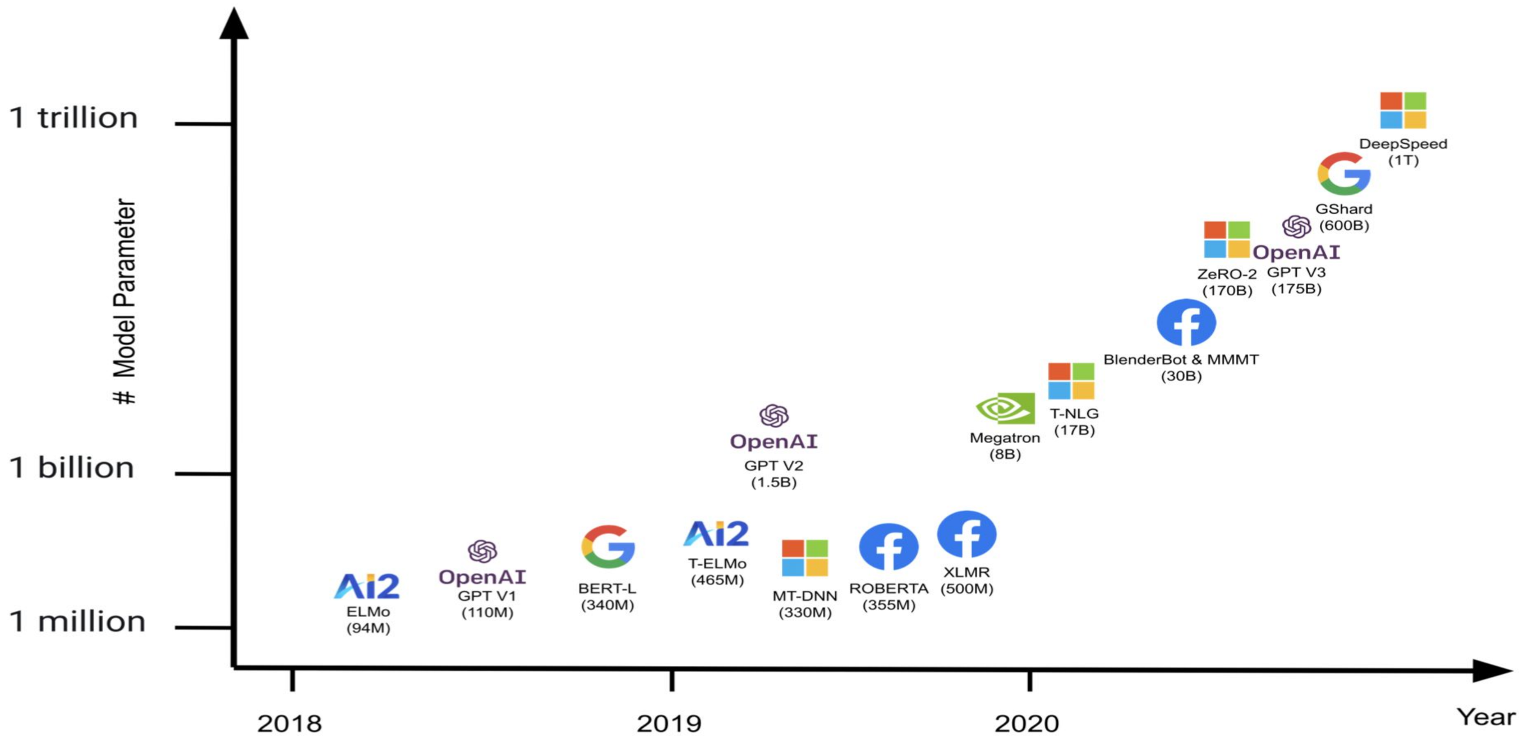
Efficient & Scalable NLP through Retrieval-Augmented Language Models

Scott Wen-tau Yih

Meta AI - FAIR

Language Model Pre-training: Ever bigger scale!



(slide from Luke Zettlemoyer, adapted from Myle Ott)

What is a language model (LM)?

$$p(w_1, \dots, w_n) = \prod_{i=1..n} p(w_i | w_1, \dots, w_{i-1})$$

e.g. *p(ELMo and BERT are Sesame Street characters) =*
p(ELMo) x p(and | ELMo) x p(BERT | ELMo, and) x ...

$p(w_i | w_1, \dots, w_{i-1})$ is often a (recurrent) neural network

(slide from Luke Zettlemoyer)

Not just bigger… Zero-shot learner!

Classification

Classification

Classify items into categories via example.

Prompt

The following is a list of companies and the categories they fall into

Facebook: Social media, Technology
LinkedIn: Social media, Technology, Enterprise, Careers
Uber: Transportation, Technology, Marketplace
Unilever: Conglomerate, Consumer Goods
Mcdonalds: Food, Fast Food, Logistics, Restaurants
FedEx:

Sample response

Logistics, Transportation

TL;DR summarization

Transformation Generation

This prompt summarizes text by adding a 'tl;dr:' to the end of a text passage. It shows that the API understands how to perform a number of tasks with no instructions.

Prompt

A neutron star is the collapsed core of a massive supergiant star, which had a total mass of between 10 and 25 solar masses, possibly more if the star was especially metal-rich.[1] Neutron stars are the smallest and densest stellar objects, excluding black holes and hypothetical white holes, quark stars, and strange stars.[2] Neutron stars have a radius on the order of 10 kilometres (6.2 mi) and a mass of about 1.4 solar masses.[3] They result from the supernova explosion of a massive star, combined with gravitational collapse, that compresses the core past white dwarf star density to that of atomic nuclei.

tl;dr:

Sample response

A neutron star is a star that is so dense that it has collapsed into a sphere the size of a city.

Q&A

Answers Generation Conversation

This prompt creates a question + answer structure for answering questions based on existing knowledge.

Prompt

I am a highly intelligent question answering bot. If you ask me a question that is rooted in truth, I will give you the answer. If you ask me a question that is nonsense, trickery, or has no clear answer, I will respond with "Unknown".

Q: What is human life expectancy in the United States?
A: Human life expectancy in the United States is 78 years.

Q: Who was president of the United States in 1955?
A: Dwight D. Eisenhower was president of the United States in 1955.

Q: Which party did he belong to?
A: He belonged to the Republican Party.

Q: What is the square root of banana?
A: Unknown

Q: How does a telescope work?
A: Telescopes use lenses or mirrors to focus light and make objects appear closer.

Q: Where were the 1992 Olympics held?
A: The 1992 Olympics were held in Barcelona, Spain.

Q: How many squigs are in a bonk?
A: Unknown

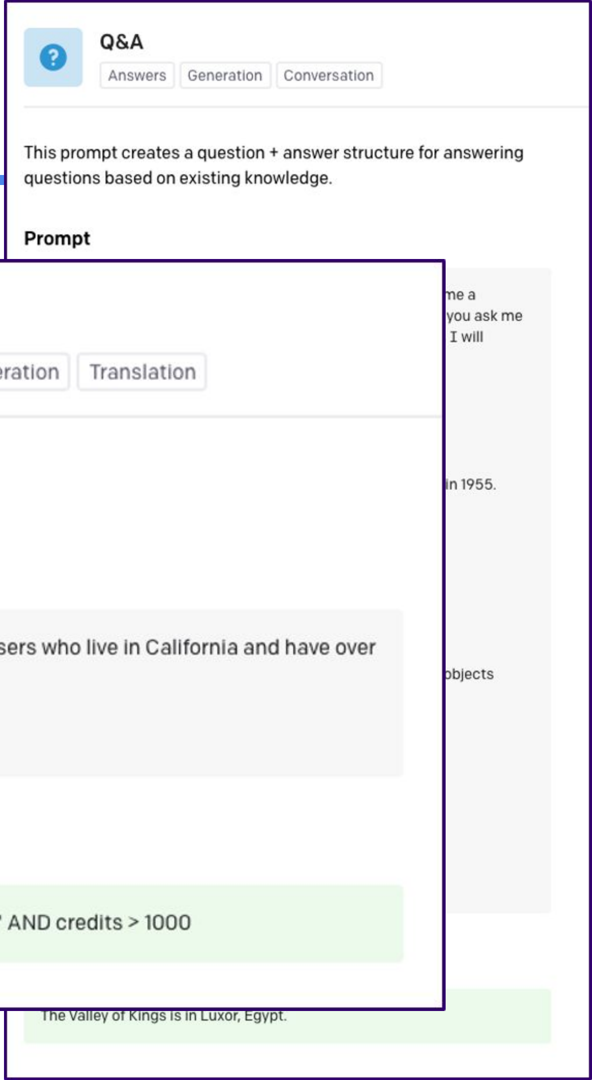
Q: Where is the Valley of Kings?
A:

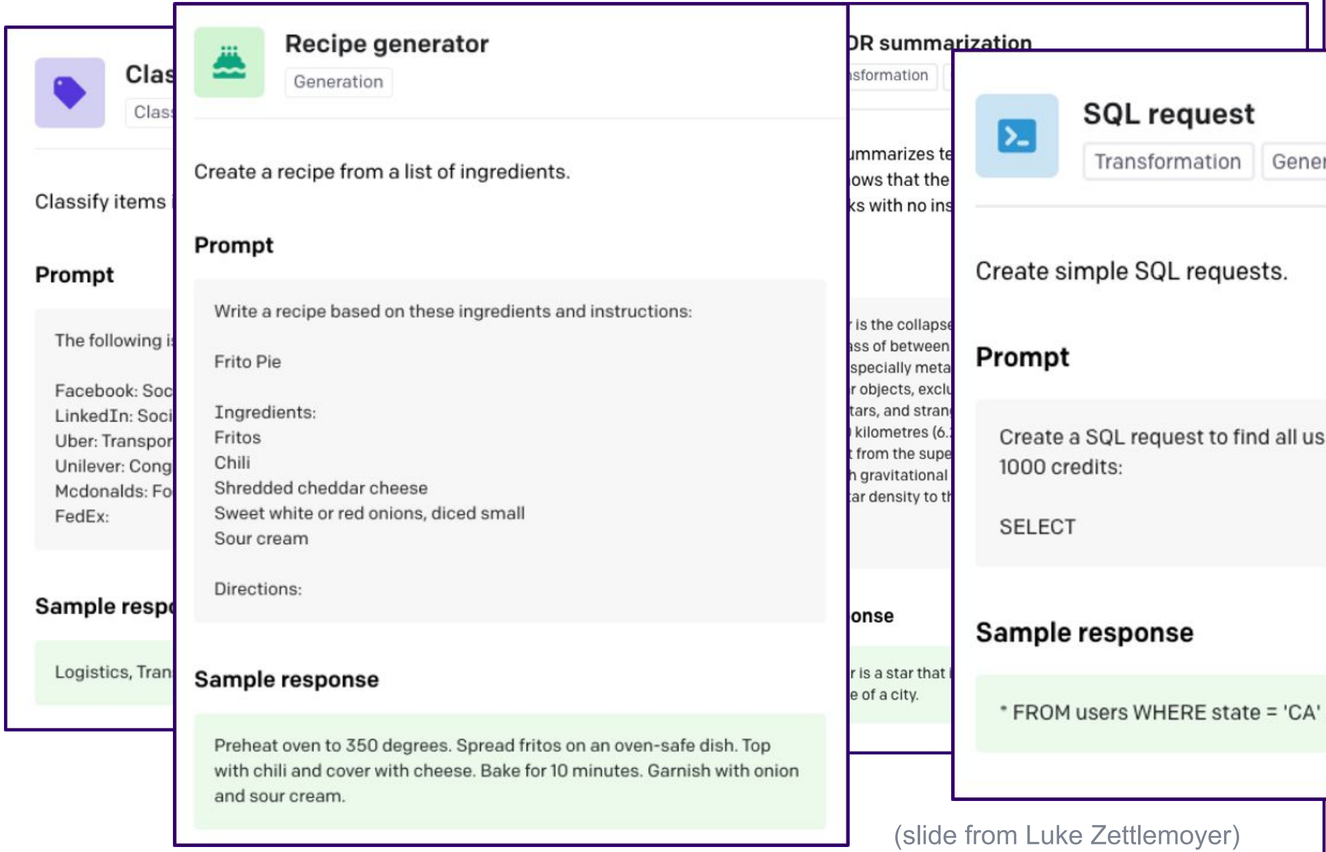
Sample response

The Valley of Kings is in Luxor, Egypt.

(slide from Luke Zettlemoyer)

Not just bigger... Zero-shot learner!

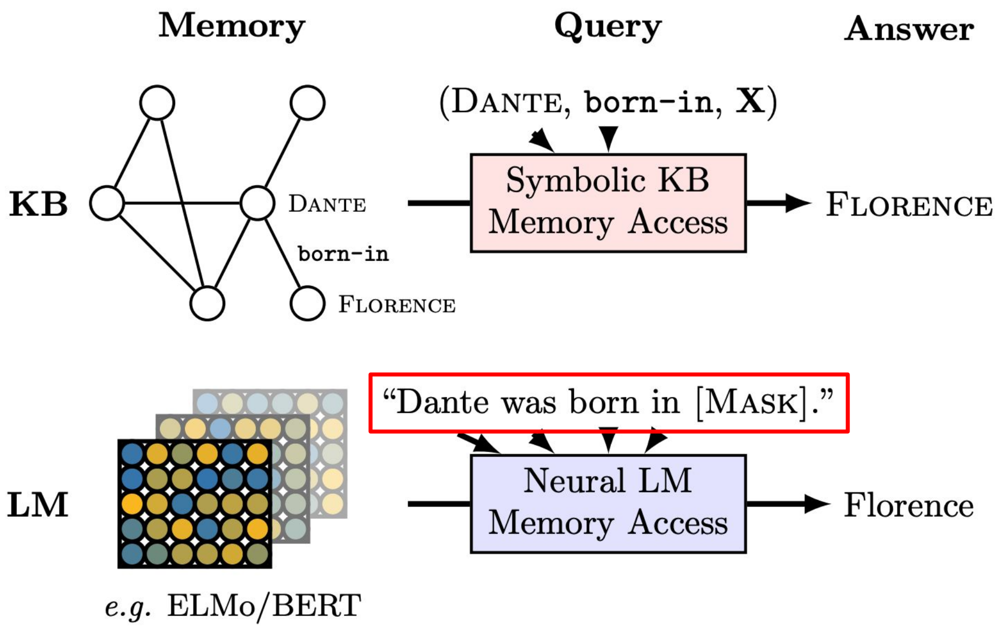




(slide from Luke Zettlemoyer)

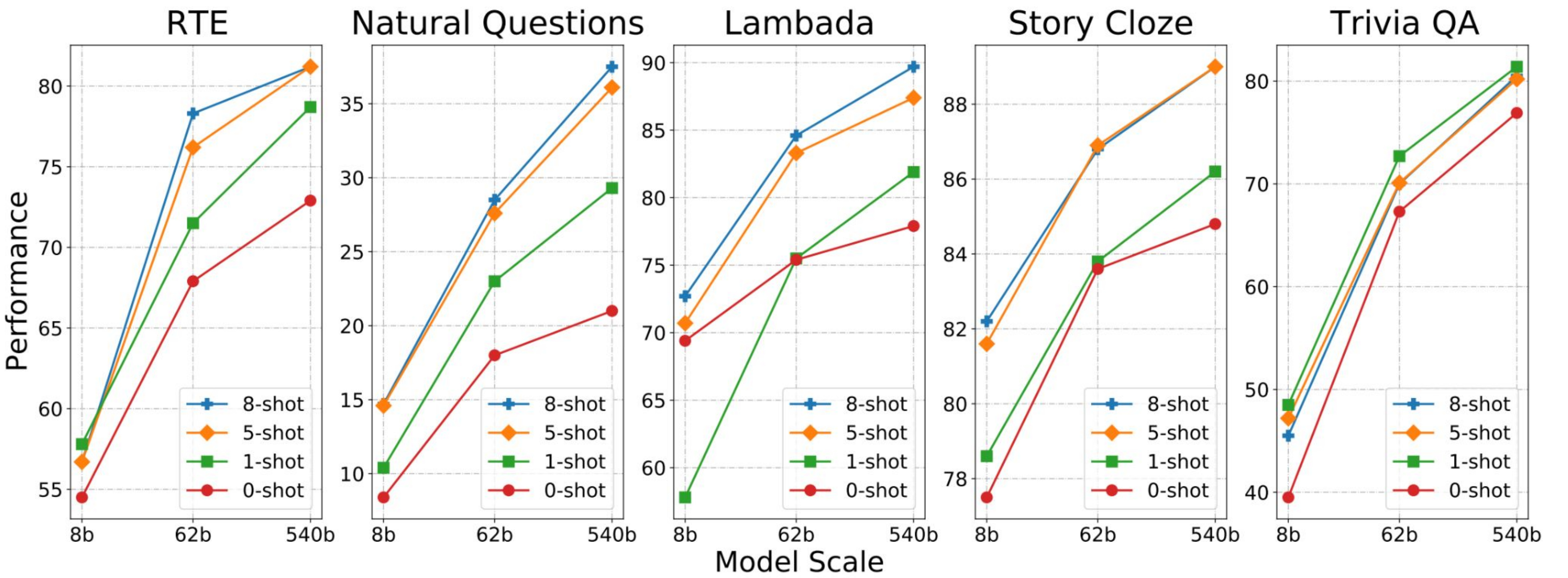
Knowledge in LM's Parameter Space

Petroni et al., *Language Models as Knowledge Bases?* In EMNLP-2019



(figure from Petroni et al., 2019)

More Parameters, More Knowledge?



(figure from PaLM; Chowdhery et al., 2022)

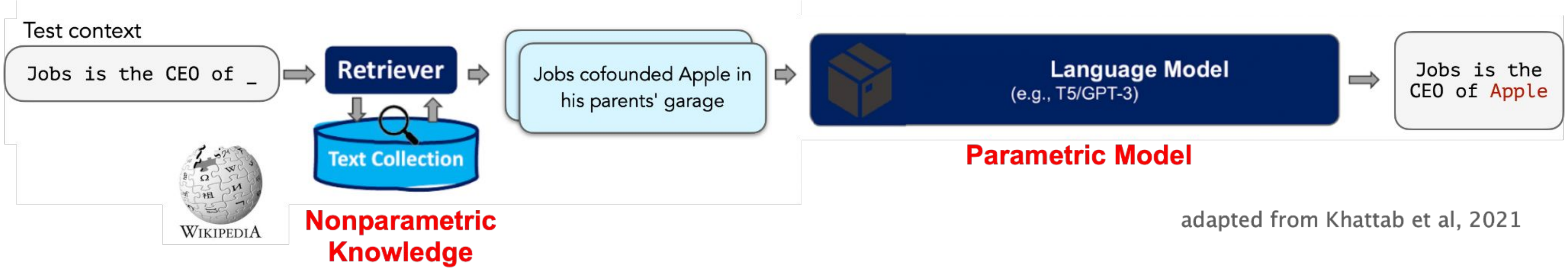
Better Large Language Model Training?

- Training large language models is expensive
  - GPT-3 175B: 355 years on one Tesla V100, ~$4.6M (2020 numbers; source)
  - PaLM 540B: 842 years on one TPU v4 chip, ~$17M (2022 numbers; source)

- Can we use an external knowledge store instead of cramming knowledge into parameters?
  - Smaller, core model has the basic capabilities
  - Retrieval component provides relevant knowledge at inference time

Retrieval-Augmented Language Models

Retrieval-augmented Language Models



adapted from Khattab et al, 2021

Separating world knowledge information from LMs' parameters

(slide from Weijia Shi)

Advantages of Retrieval-augmented LMs

- Parameter efficient
  - Knowledge is explicitly encoded in the datastore
  - Fewer model parameters are needed for memorization

Advantages of Retrieval-augmented LMs

- Parameter efficient
  - Knowledge is explicitly encoded in the datastore
  - Fewer model parameters are needed for memorization
- Less opaque; more interpretable
  - Easier to trace the knowledge source of the predictions

Advantages of Retrieval-augmented LMs

- Parameter efficient
  - Knowledge is explicitly encoded in the datastore
  - Fewer model parameters are needed for memorization
- Less opaque; more interpretable
  - Easier to trace the knowledge source of the predictions
- Easy to update knowledge
  - The datastore can be updated and expanded easily
  - No model retraining is needed

This Talk

- REPLUG: Retrieval-Augmented Black-Box Language Models (arXiv Link)
    - Retrieval can help improve language models even in the “black-box” setting
- RA-CM3: Retrieval-Augmented Multimodal Language Modeling (arXiv Link)
    - Works on multimodal (text / image) as well

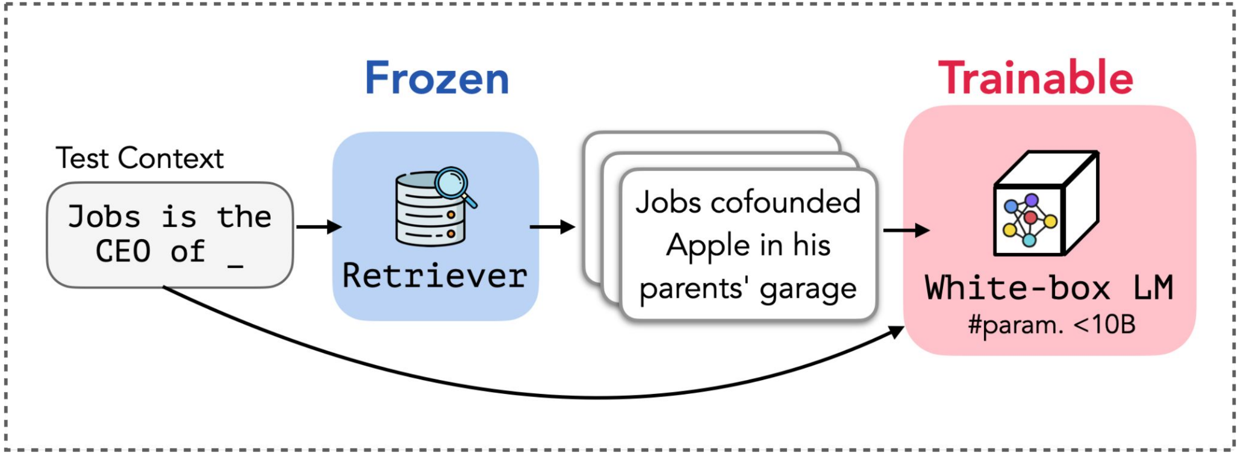
REPLUG: Retrieval-Augmented Black-Box Language Models

Weijia Shi, Sewon Min, Minjoon Seo, Michihiro Yasunaga, Rich James, Mike Lewis, Luke Zettlemoyer, Scott Yih

(Slides from Weijia Shi)

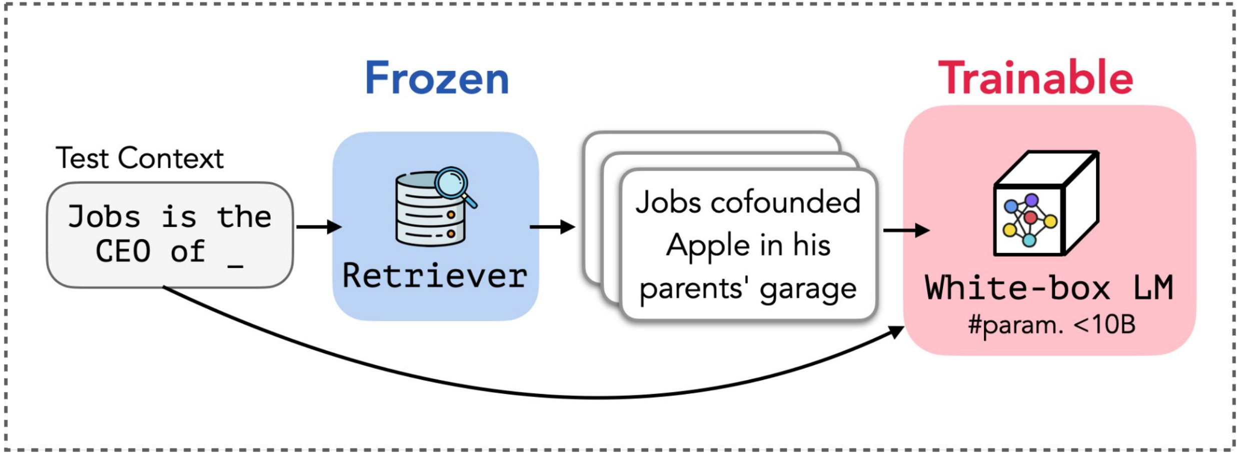
Previous Retrieval-Augmented LMs

Previous retrieval-augmented LMs
(e.g., RETRO (Borgeaud, et al. 2022), RAG (Lewis, et al. 2020))



Previous Retrieval-Augmented LMs

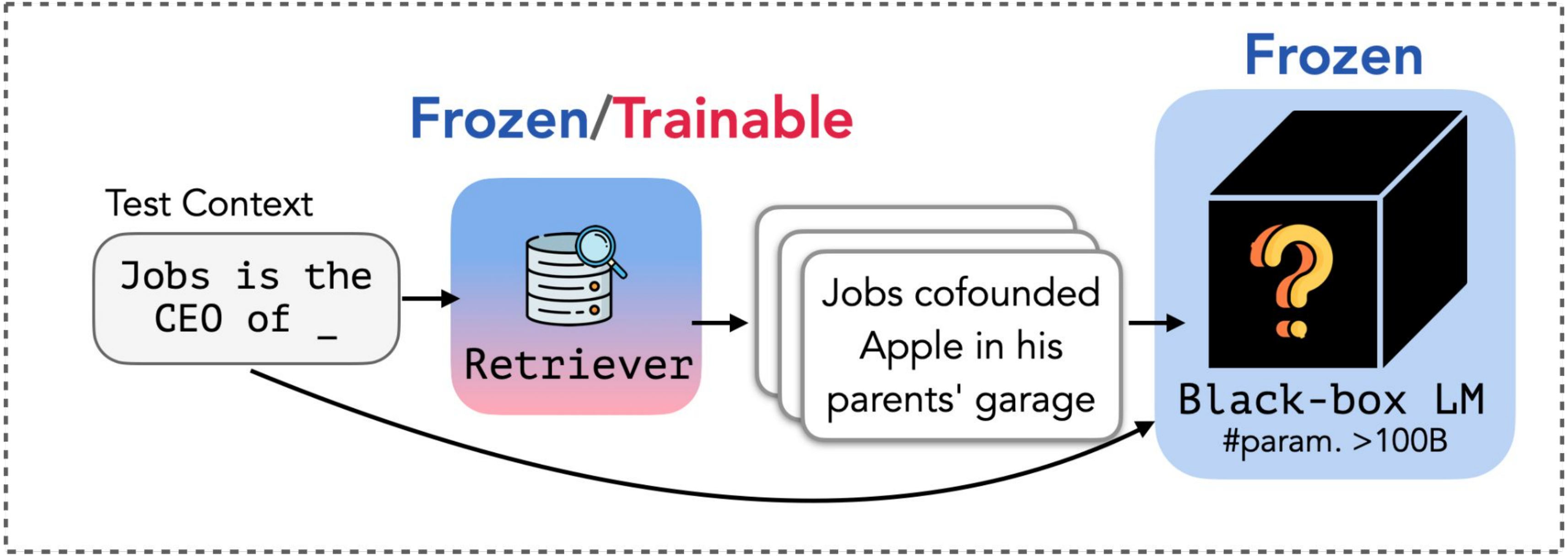
Previous retrieval-augmented LMs
(e.g., RETRO (Borgeaud, et al. 2022), RAG (Lewis, et al. 2020))



Not suitable for large language models

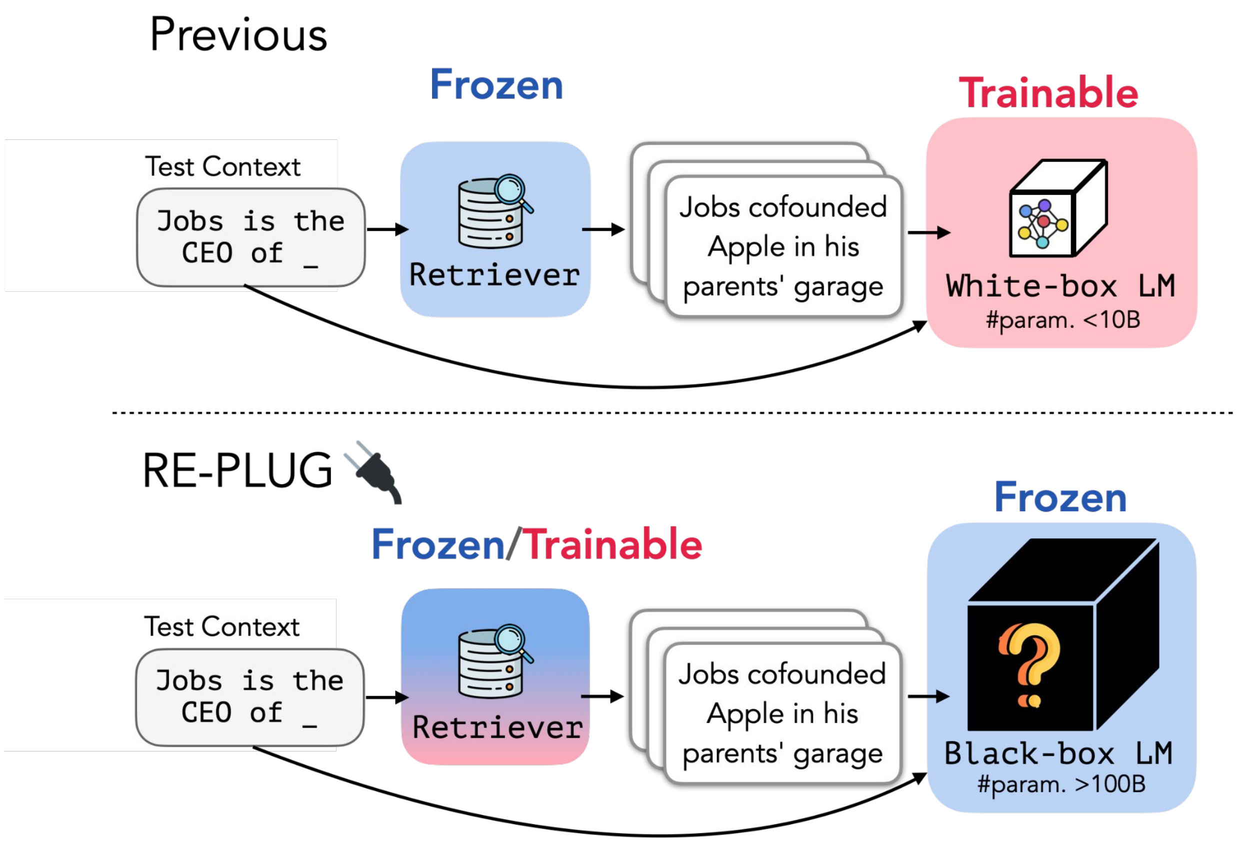
- e.g., expensive to finetune or only accessible by APIs

Our Framework: REPLUG



- How to incorporate retrieved texts?
- How to train a better dense retriever for language modeling and downstream tasks?

Comparison: Previous vs. REPLUG



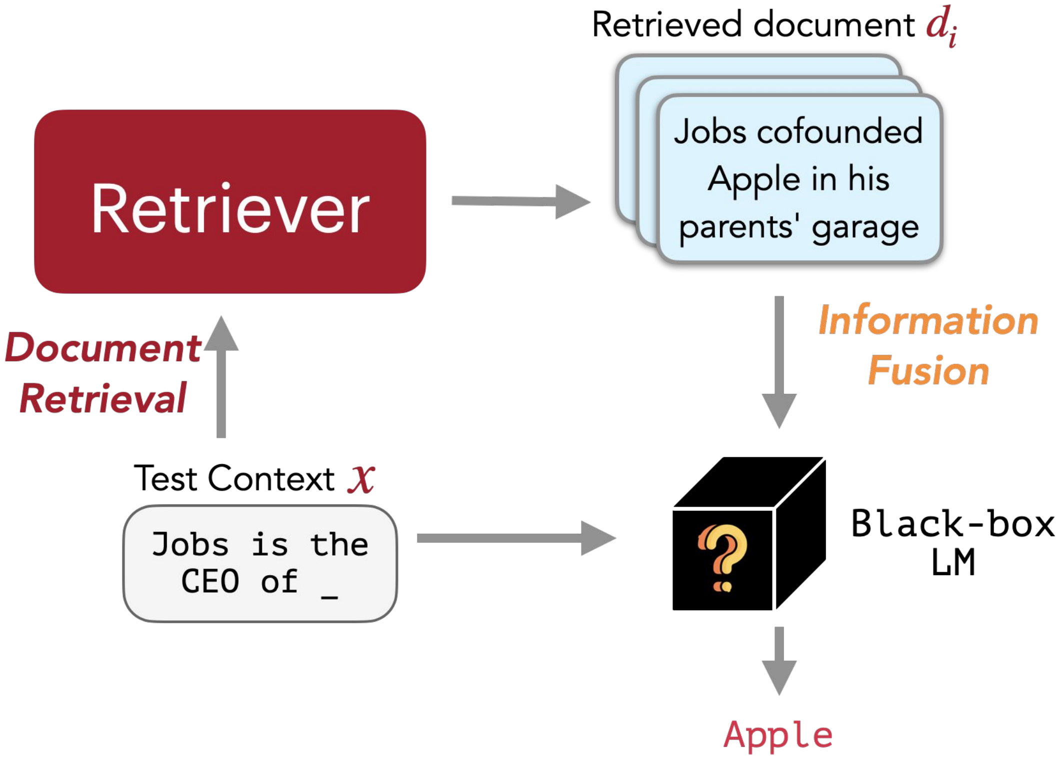
Our Method

- REPLUG Inference
    1. Retrieves a small set of relevant documents from an external corpus
    2. Prepends each document separately to the input context
    3. Ensembles LM output probabilities

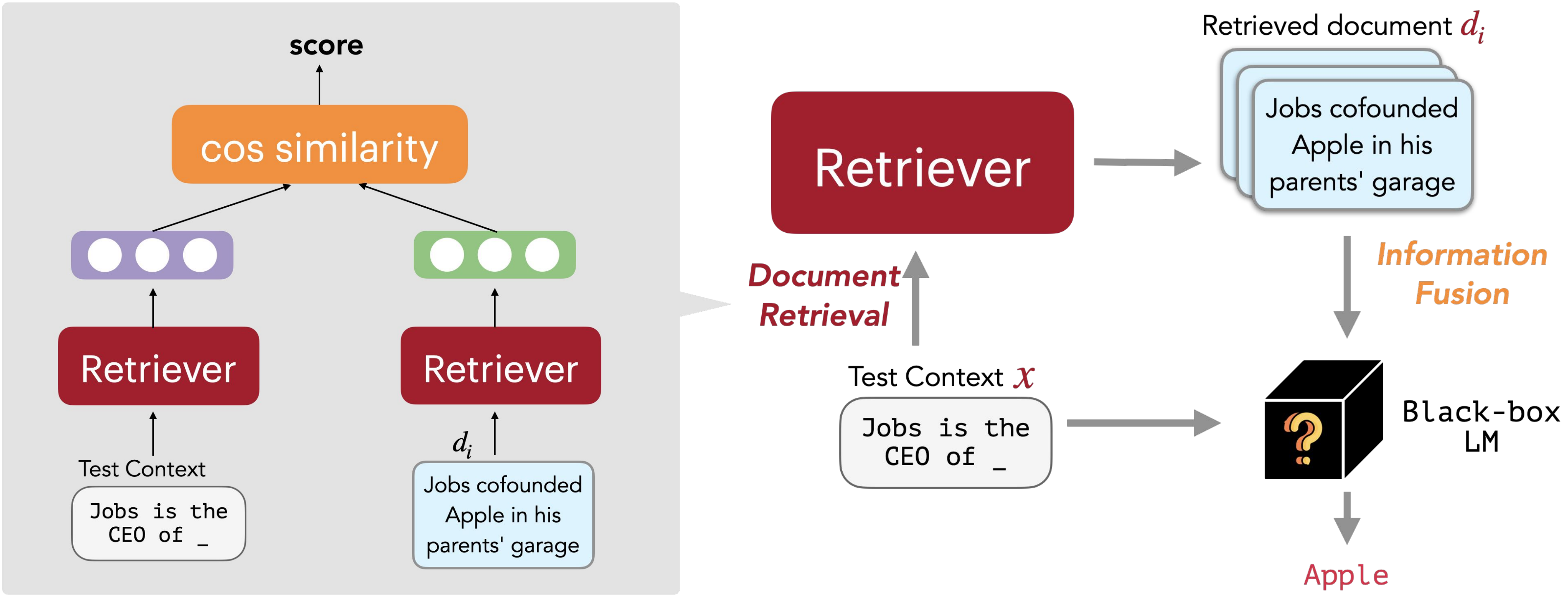
- REPLUG LSR (LM-Supervised Retrieval): Training the dense retriever

REPLUG Inference

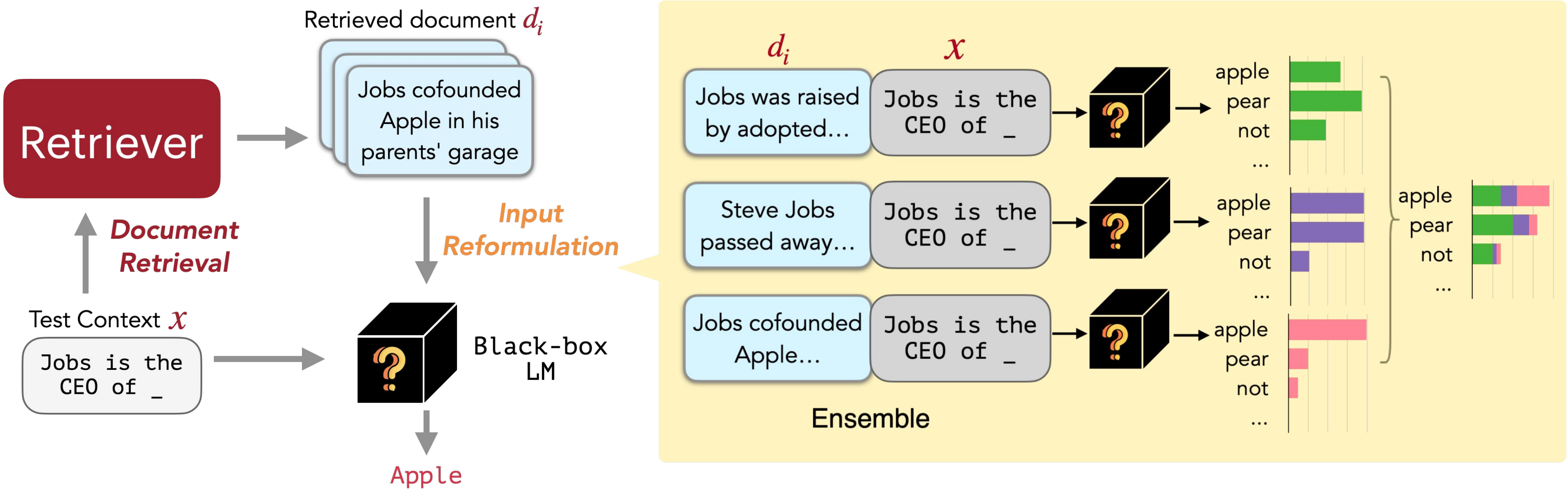
1. Document retrieval
2. Information fusion



REPLUG **Inference** - Document Retrieval



RePlug **Inference** - Information Fusion



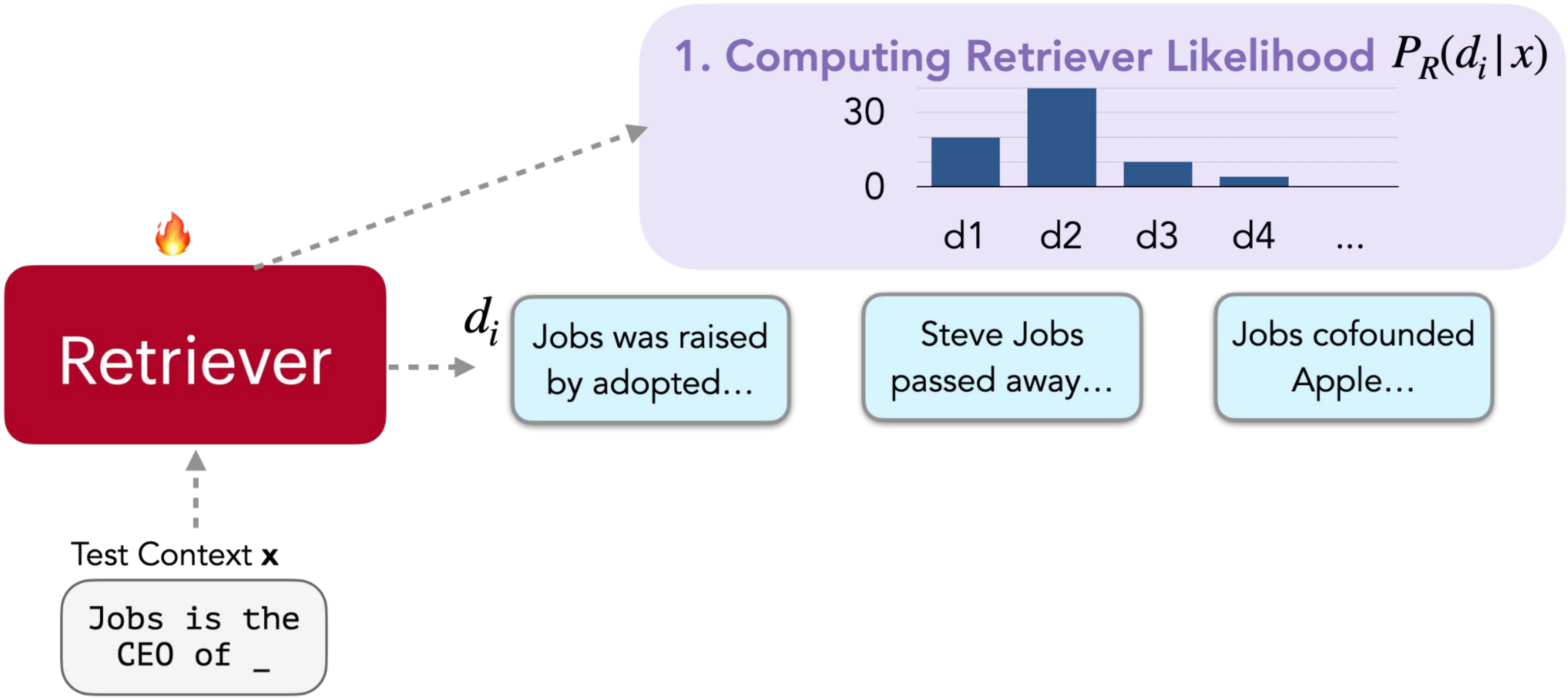
Our Method

- REPLUG Inference

- REPLUG LSR (LM-Supervised Retrieval): Training the dense retriever
    - Use the LM output as supervision signals for different inputs (context + retrieved document)
    - Train the retriever using the “labeled” data

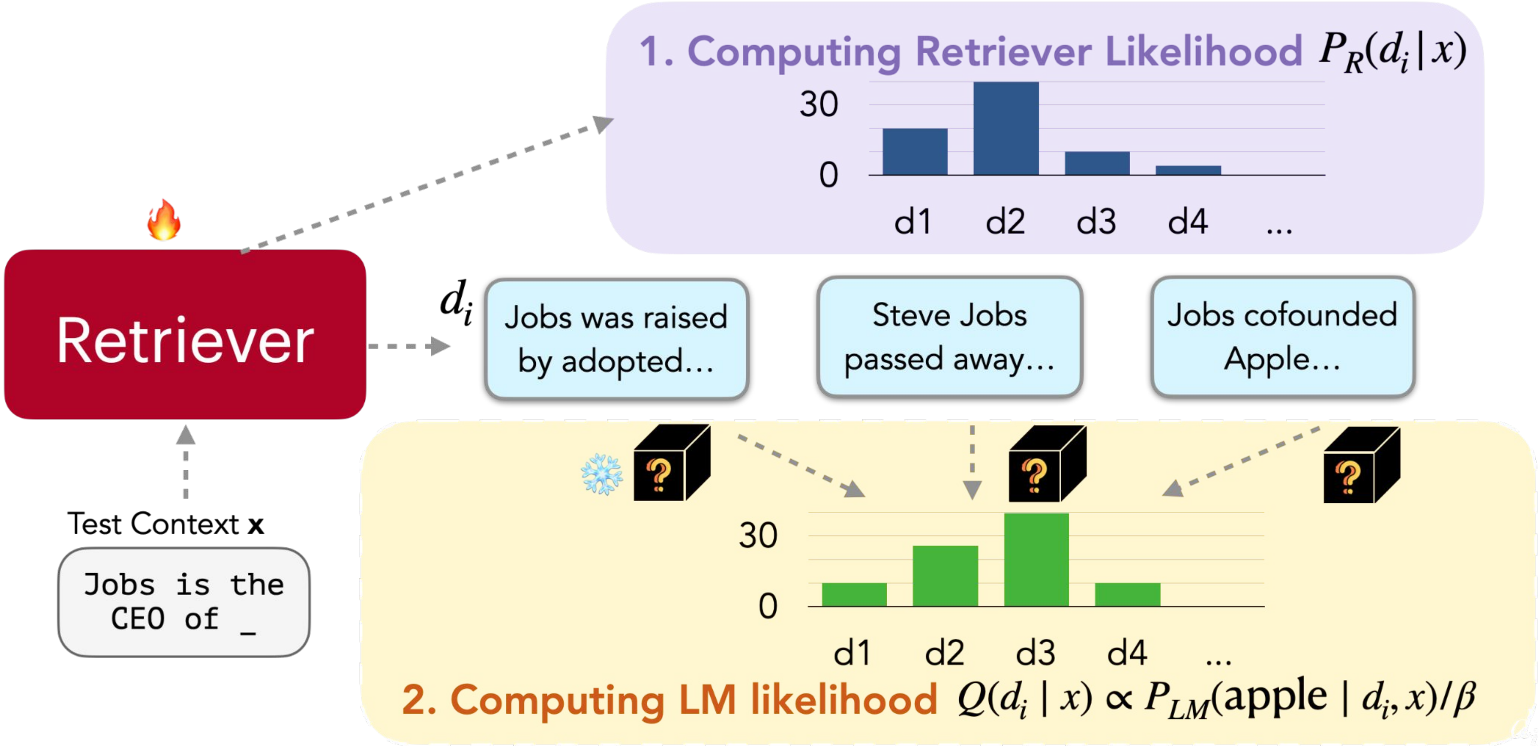
REPLUG LSR Training

Step 1: Compute retriever likelihood



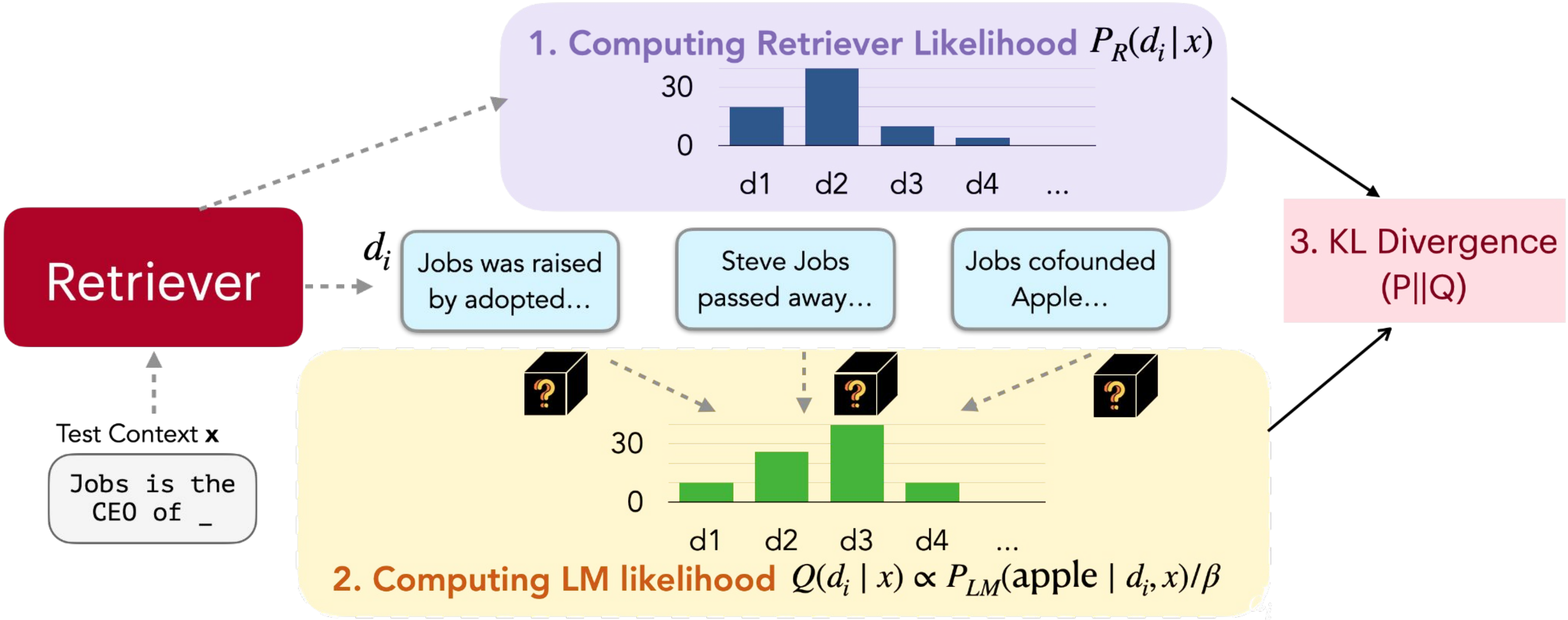
RePlug LSR Training

Step 2: Compute LM likelihood



REPLUG LSR Training

Step 3: Minimize KL divergence (Update only the retriever parameters)



REPLUG: **Retriever and Training Setup**

- REPLUG: using an off-the-shelf retriever, Contriever (Izacard et al., 2022)
- REPLUG LSR: a tuned retriever adapted to a black-box LM
  - Initialized with Contriever
  - Trained on the Pile with supervision signals provided by GPT-3 Curie

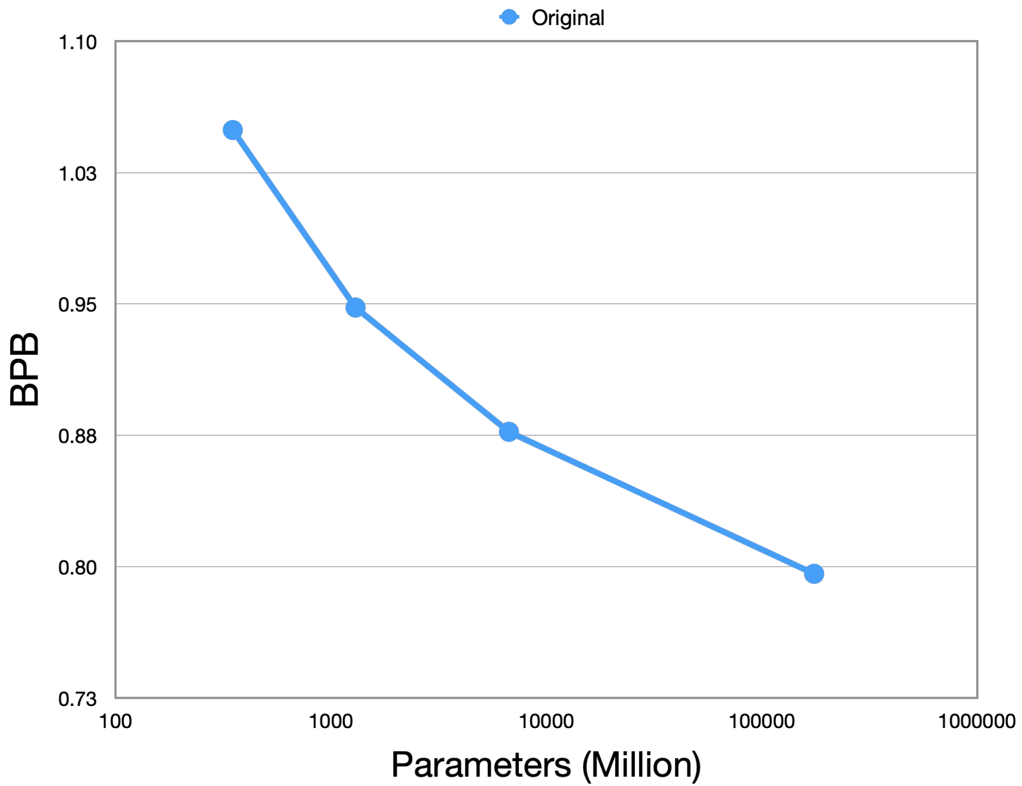
Experiments

- Language Modeling
- MMLU (Massive Multitask Language Understanding)
- Open-domain Question Answering

Language Modeling

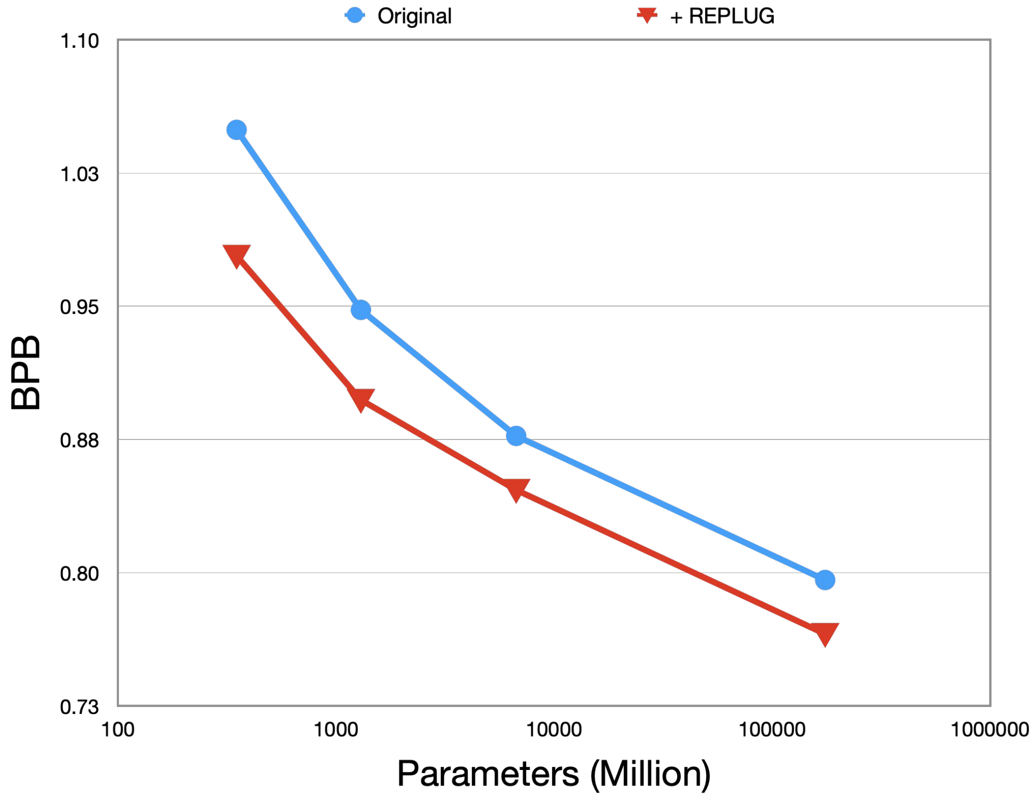
- Evaluation: the Pile test set
- Datastore: Subset of the Pile training data (367M documents of 128 tokens)

Language Modeling



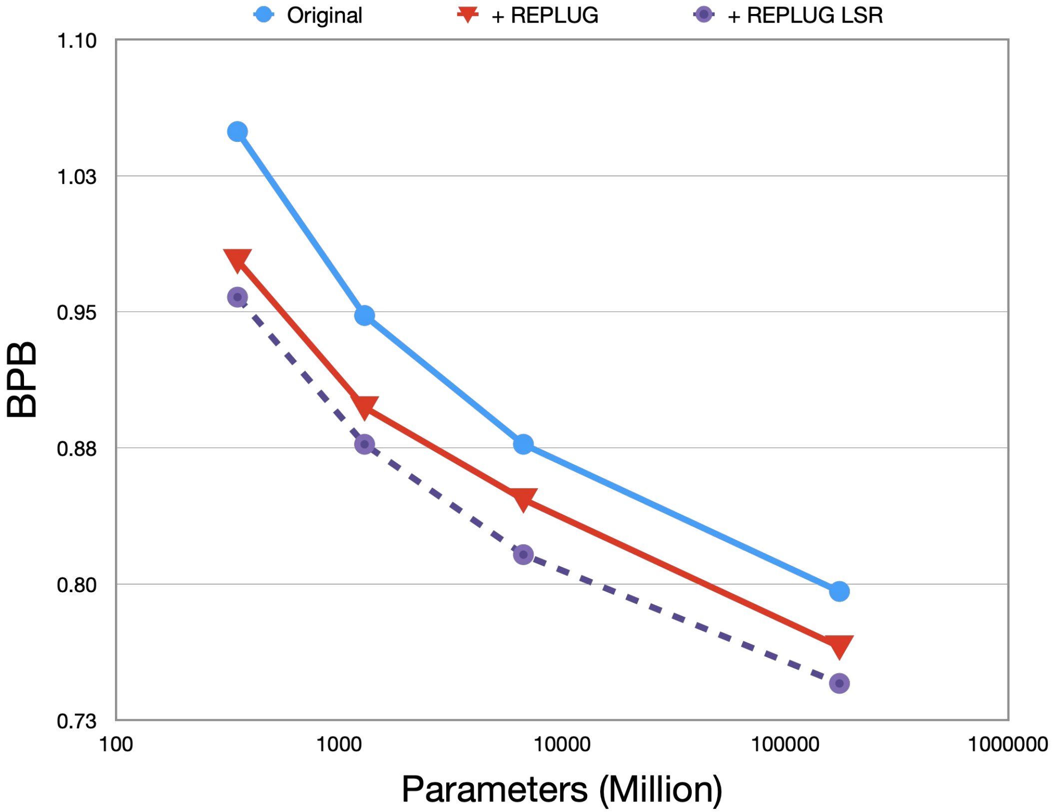
GPT-3

Language Modeling



GPT-3

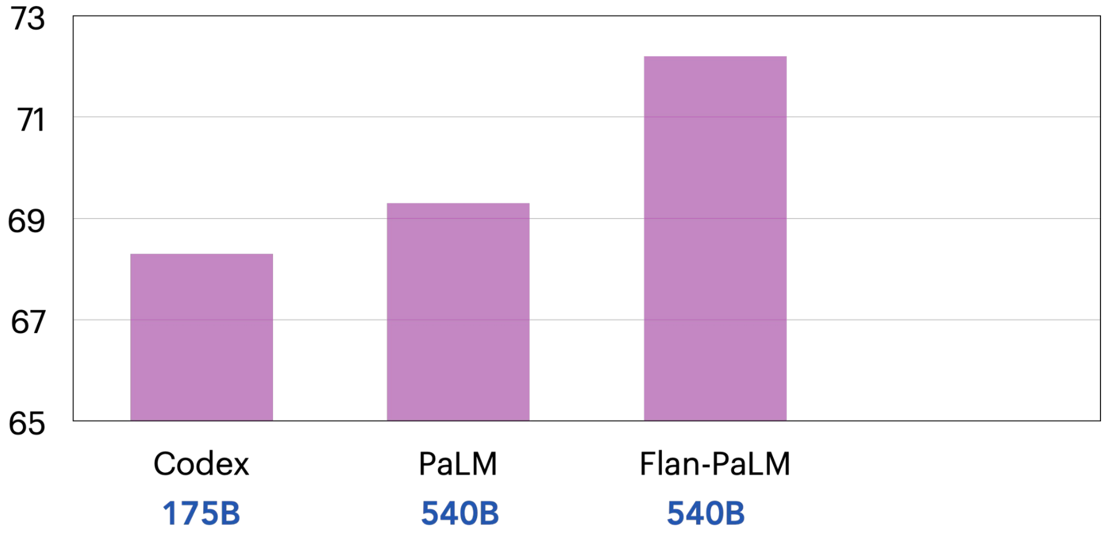
Language Modeling



GPT-3

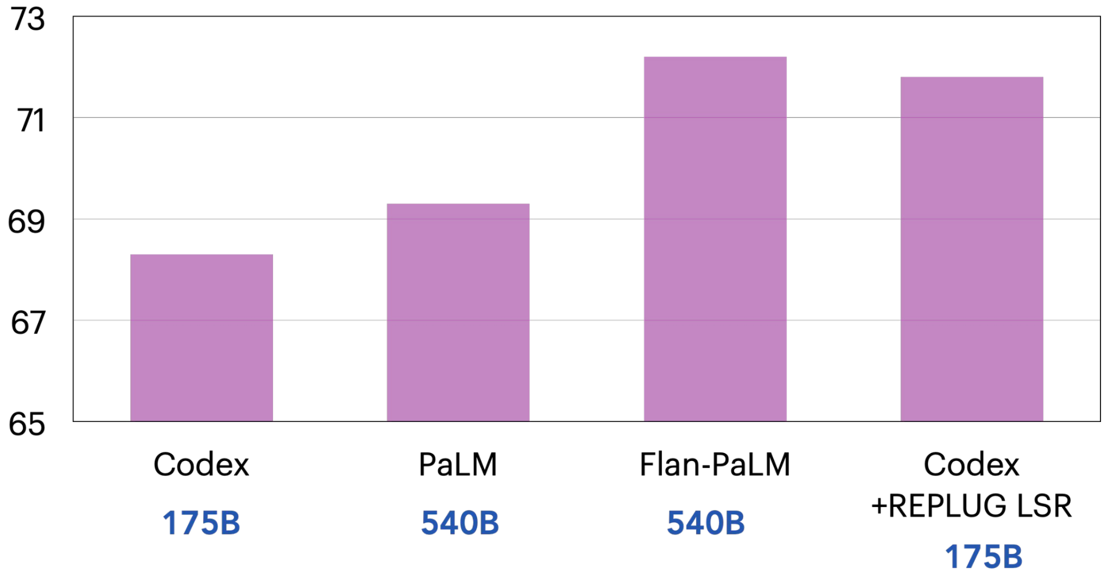
MMLU (Massive Multitask Language Understanding)

- A multiple-choice QA dataset covering exam questions from 57 tasks (e.g., Math, CS, Law, Psychology, etc.)

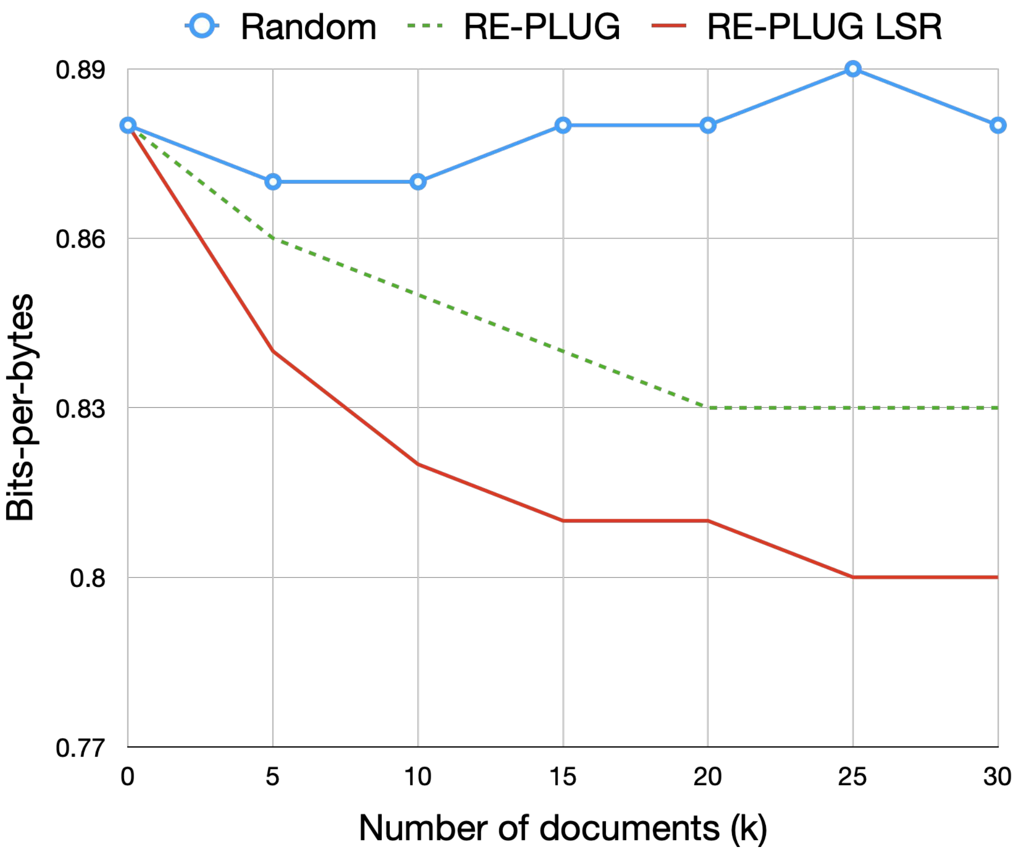


MMLU (Massive Multitask Language Understanding)

- A multiple-choice QA dataset covering exam questions from 57 tasks (e.g., Math, CS, Law, Psychology, etc.)

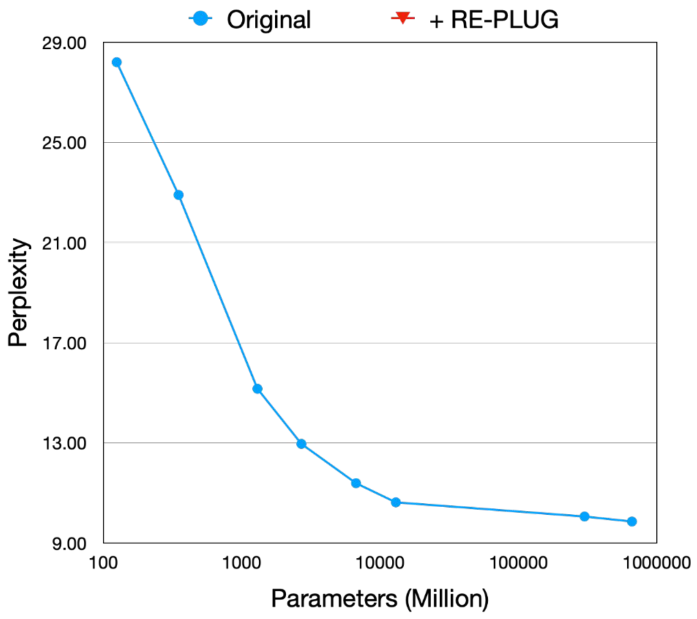


Observation: Random Documents Don’t Help

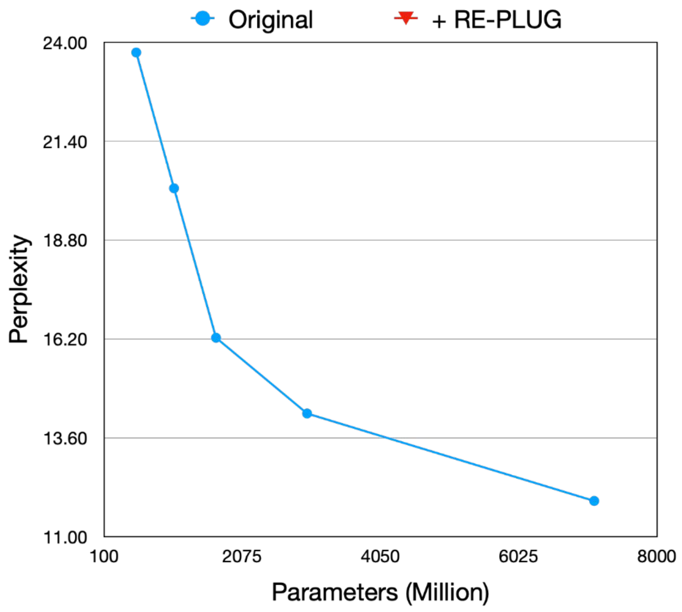


Observation: REPLUG Works on other LMs too

- Evaluation: Wikitext-103 test set
- Datastore: Wikitext-103 training set



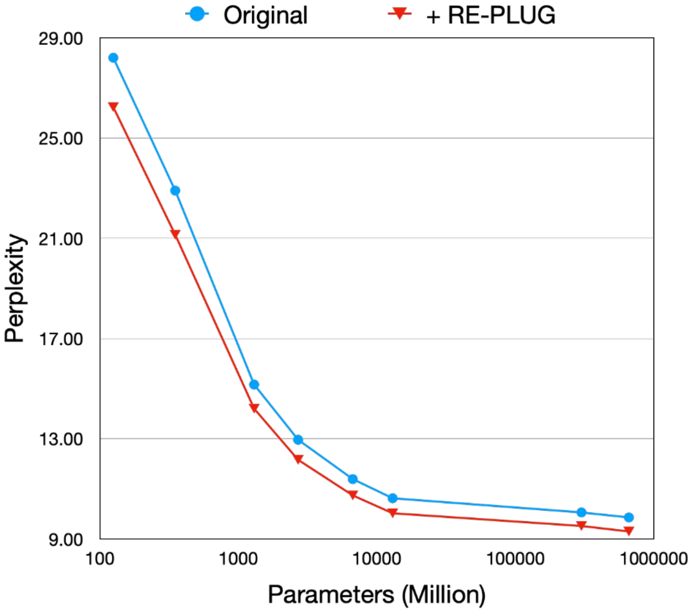
OPT



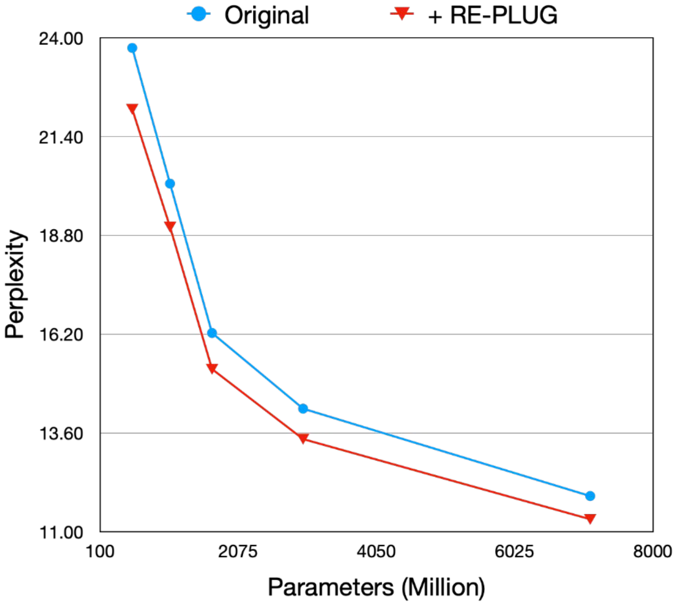
BLOOM

Observation: REPLUG Works on other LMs too

- Evaluation: Wikitext-103 test set
- Datastore: Wikitext-103 training set



OPT



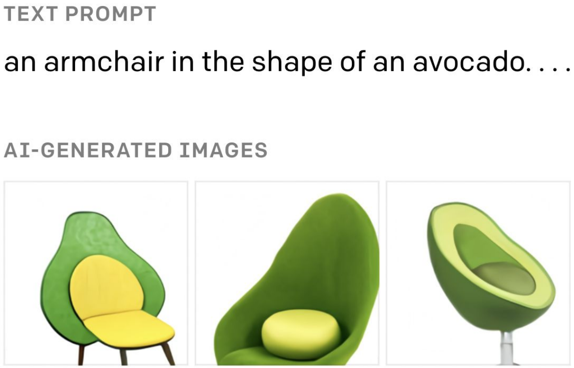
BLOOM

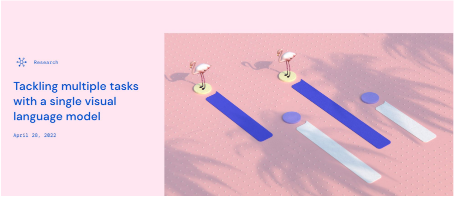
Retrieval Augmented Multimodal Model Training

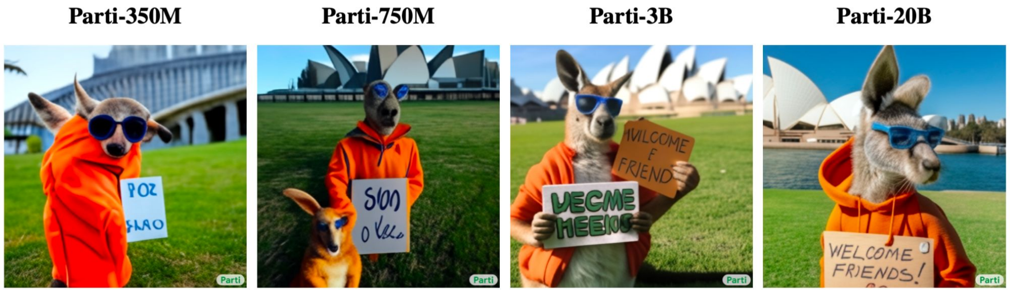
Michihiro Yasunaga, Armen Aghajanyan, Weijia Shi, Rich James, Jure Leskovec,
Percy Liang, Mike Lewis, Luke Zettlemoyer, Scott Yih

(Slides from Michihiro Yasunaga)

Multimodal Models







A portrait photo of a kangaroo wearing an orange hoodie and blue sunglasses standing on the grass in front of the Sydney Opera House holding a sign on the chest that says Welcome Friends!

https://openai.com/blog/dall-e/, https://parti.research.google/

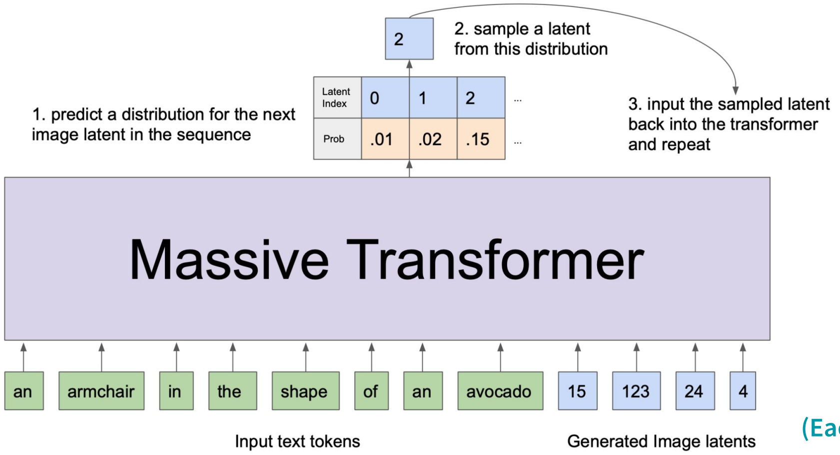
Multimodal Models

DALL·E, Parti (text → image; autoregressive)

DALL·E 2, StableDiffusion (text → image; diffusion)

Flamingo (image → text; autoregressive)

CM3 (text ⇄ image; autoregressive) ⟵ We focus on this direction



(Each image = 1024 tokens)

https://ml.berkeley.edu/blog/posts/dalle2/

Multimodal models need world knowledge

What does Bichon Frisé look like..?

A **Bichon Frisé** sitting on the bench.

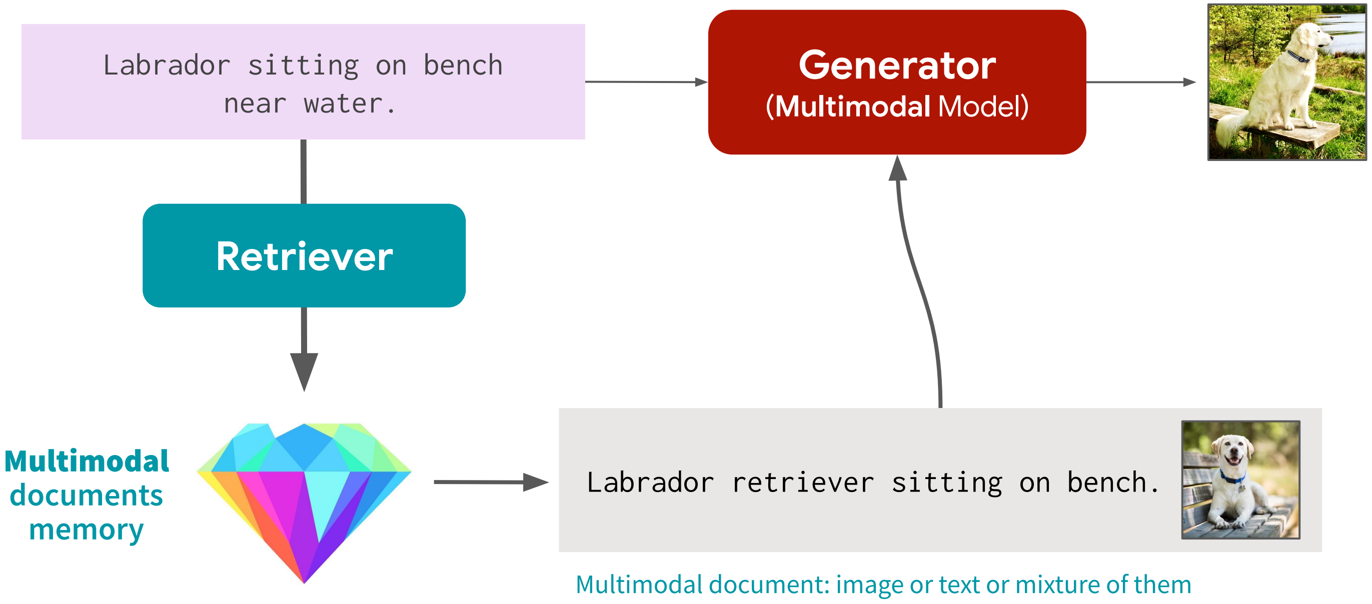
Text to image

What is the name of this place..?

Image to text

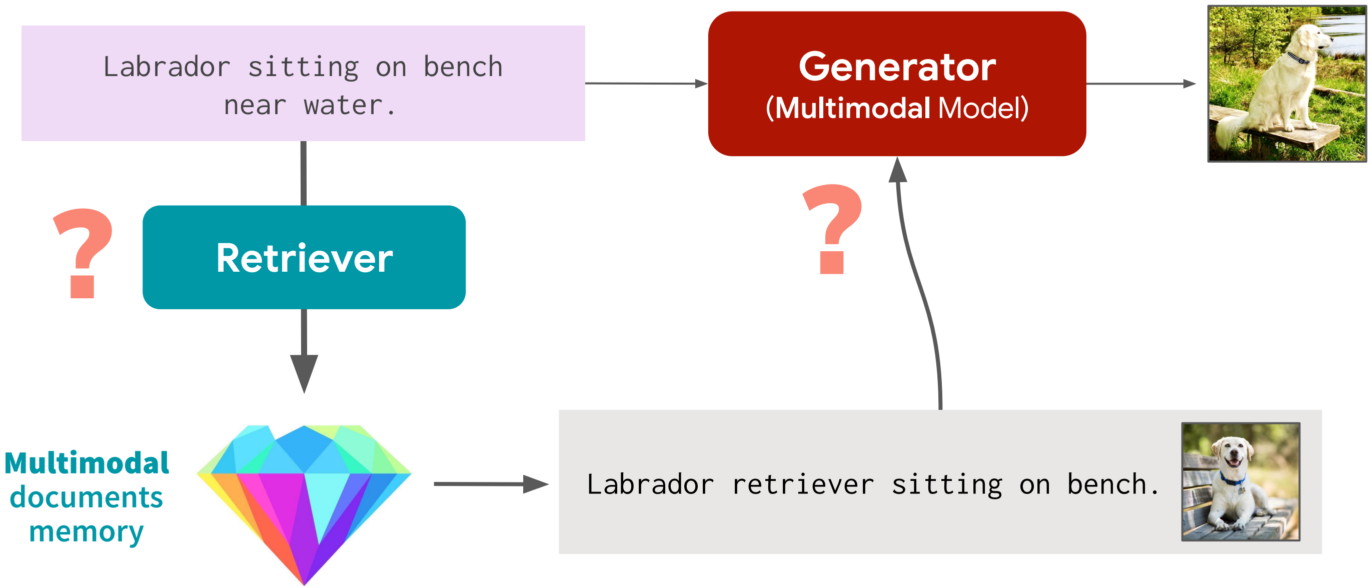
The Dragon and Tiger Pagodas next to fireworks.

Our Idea: Retrieval Augmented Multimodal Model



Challenges & Research Questions

- What is effective **retrieval** method in multimodal setting?
- How to incorporate multimodal docs into the **generator**?

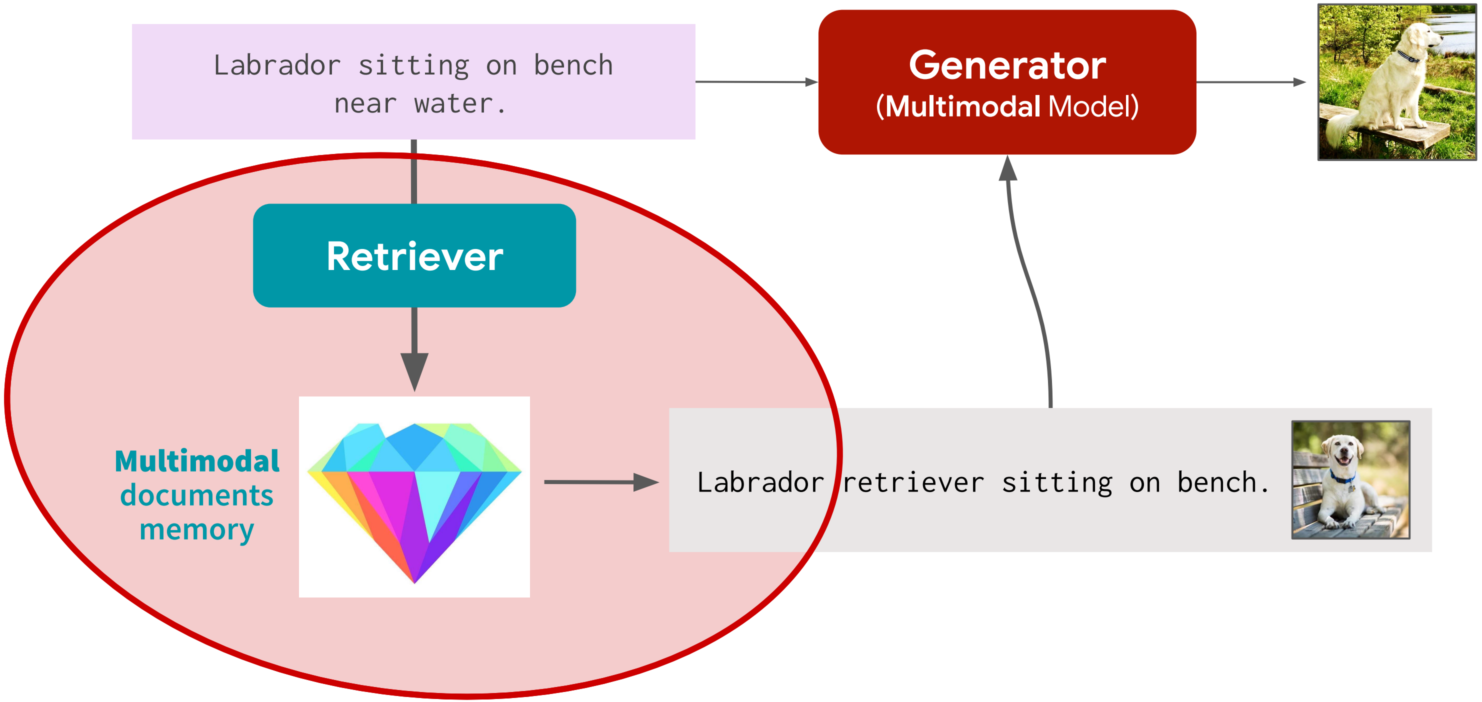


Retrieval Augmented Multimodal Model

Techniques

- Multimodal retrieval
  - Dense retriever with CLIP-based mix-modal encoder
  - Strategy for obtaining retrieved docs
- Multimodal retrieval-augmented generator
  - Prepend retrieved docs
  - Jointly train over retrieved docs and main doc

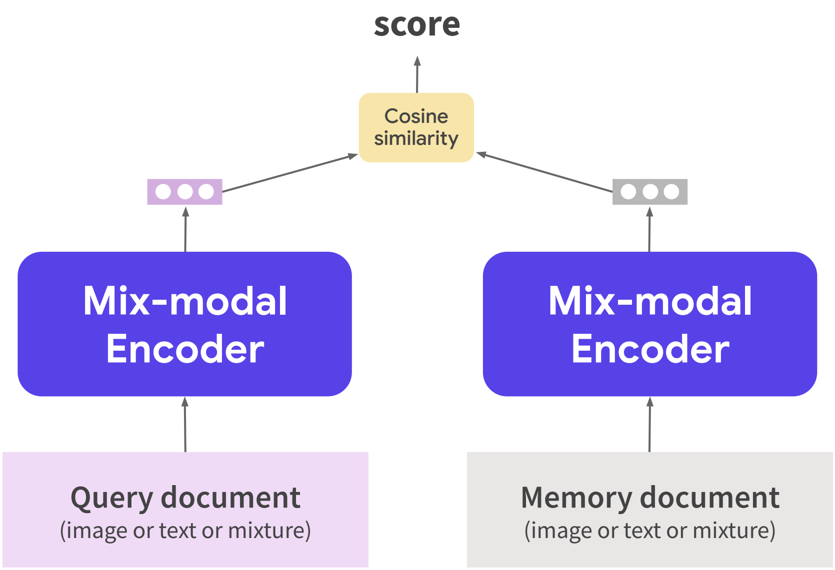
Multimodal Retrieval



Our Multimodal Retriever

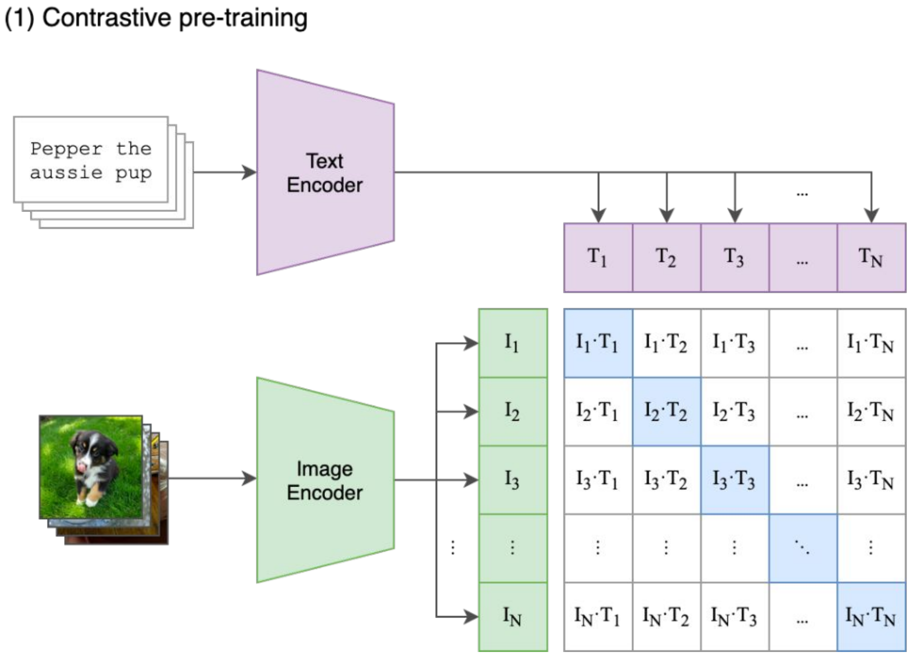
Dense Retriever with Mix-modal Encoder

```
f(query, memory) ⟶ score
```



Background: CLIP

CLIP produces text embeddings and image embeddings in shared vector space

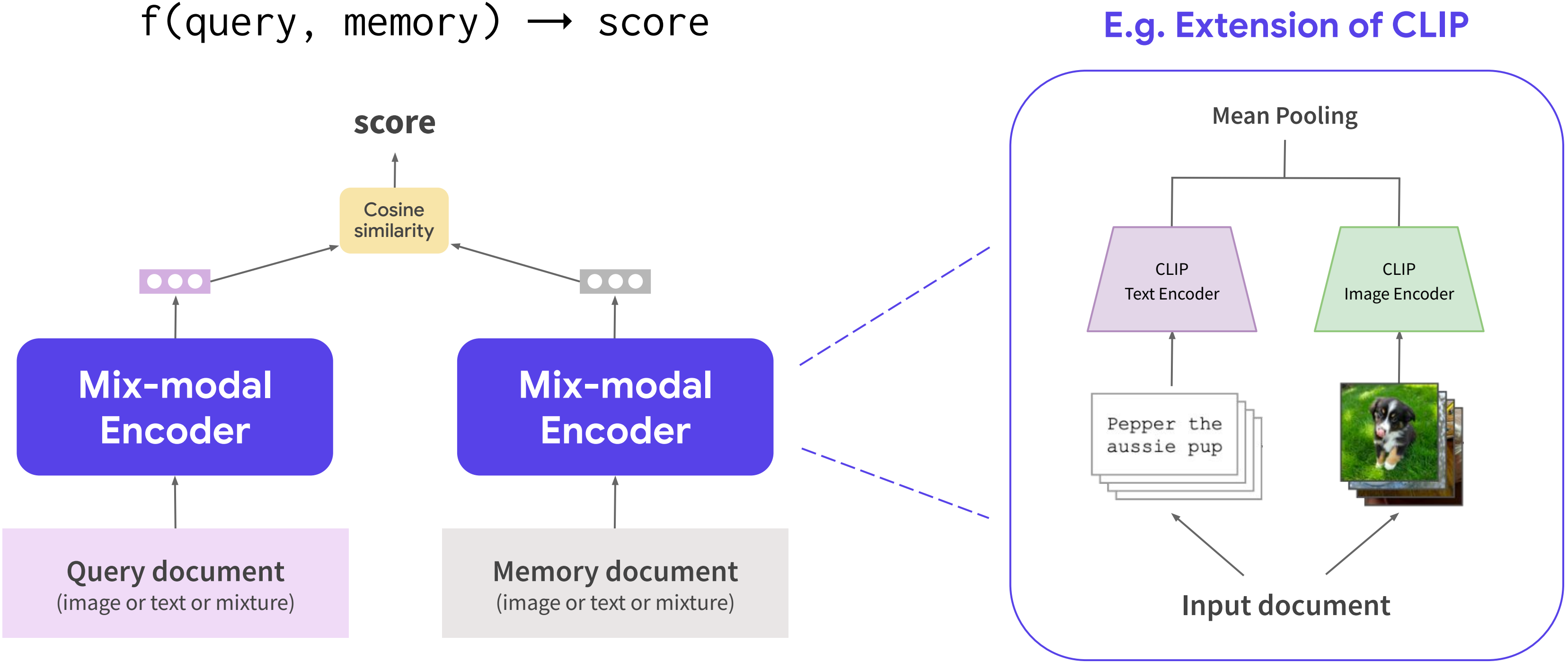


Radford+2021

Our Multimodal Retriever

Dense Retriever with Mix-modal Encoder

`f(query, memory) ⟶ score`



Our Multimodal Retriever

Example

Memory

Query

Labrador sitting on bench near water.

0.85

0.81

0.35

Labrador retriever sitting on bench.

Labrador retriever sitting by water.

⋮

Strategy for Obtaining Retrieved Documents

Relevance

The retrieved docs should be relevant to query

✅ **Cosine similarity score**

Diversity (for training)

If simply take docs of top scores, may include duplicate images/text
This can cause model to overfit or pick up repetitive decoding

💡 **Avoid redundant docs**

- Skip candidate doc if it is too similar to query or docs already retrieved

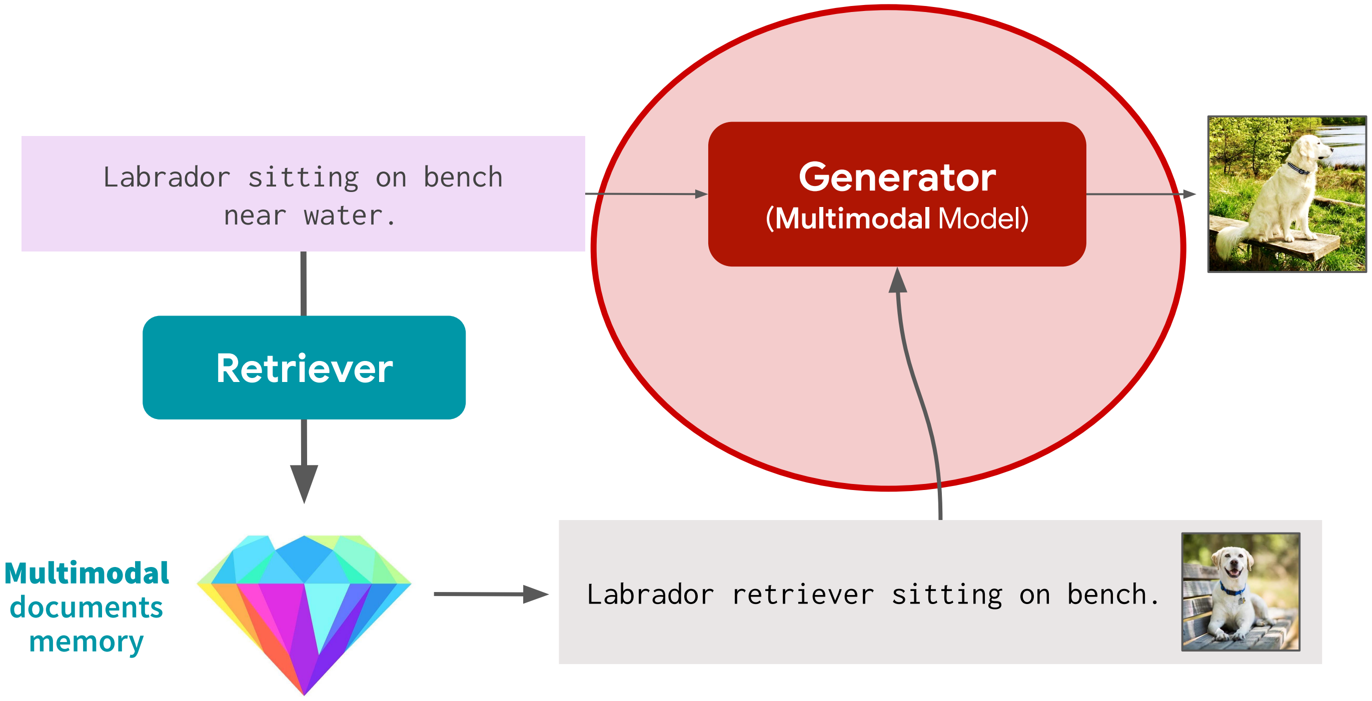
💡 **Query dropout**

- Drop some tokens of query used in retrieval (e.g., 20% of tokens)
- This further increases diversity and serves as regularization

Diversity is crucial in multimodal setting

- Multimodal dataset often contains duplicate images across docs
- Each image takes many tokens (1024), so can significantly hurt model training

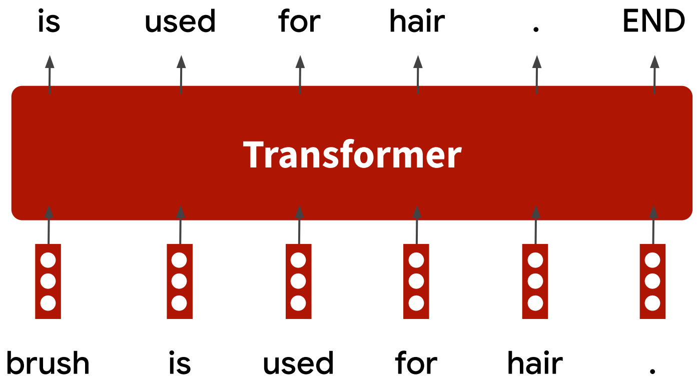
Multimodal Generator



Background: CLM and CM3

Causal language model (CLM)

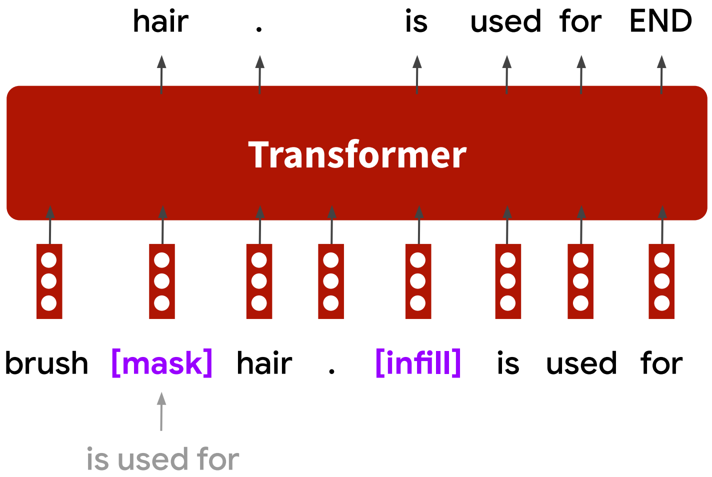
⇒ Inference: can do auto-regressive generation



Causal masked language model (CM3)

⇒ Inference: can also do in-filling

If don’t mask, same as CLM



Aghajanyan+2022

Background: CM3 can do text ⇄ image

Original doc

Labrador sitting on bench near water. 

Text to Image

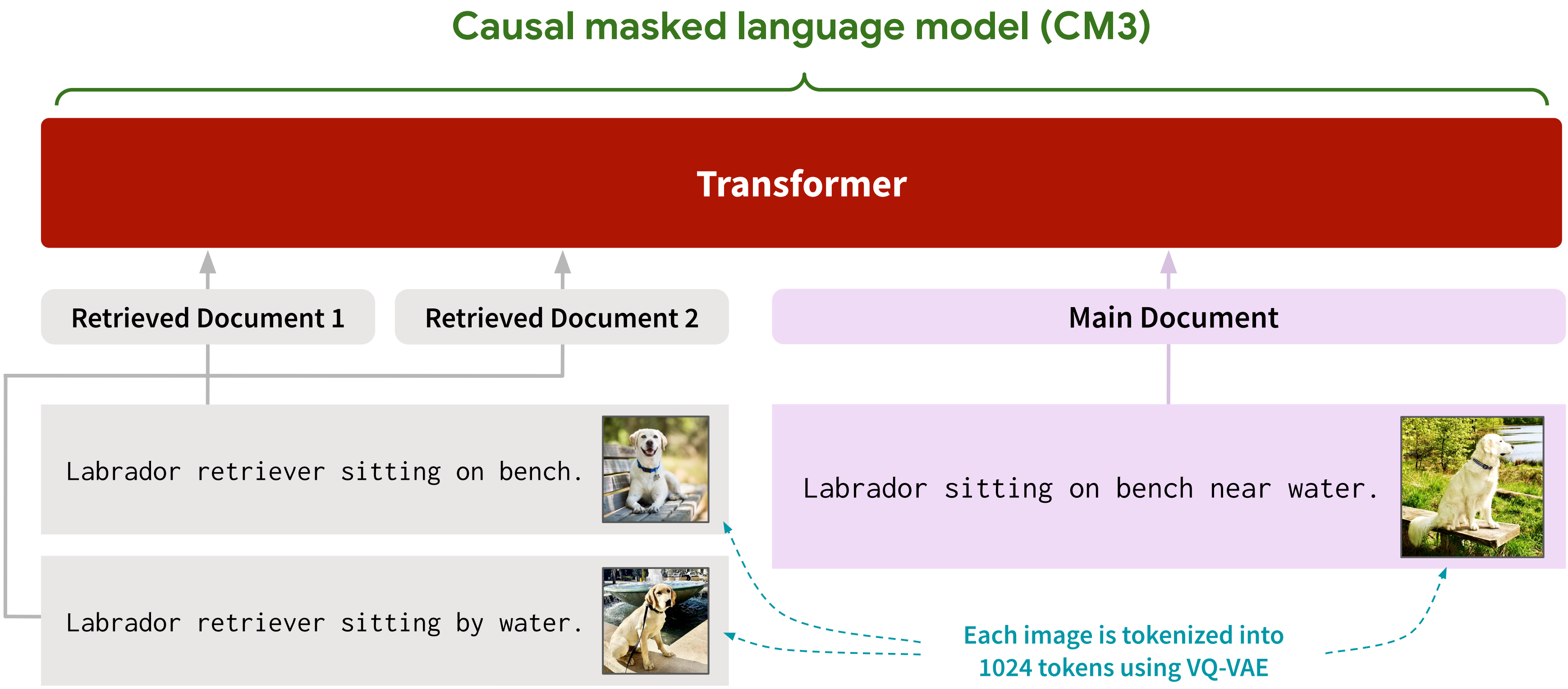
Labrador sitting on bench near water. → CM3 → 

Image to text

[mask]  [infill] → CM3 → Labrador sitting on bench near water.

Aghajanyan+2022

Our Generator: Retrieval Augmented CM3



How to Train the Generator

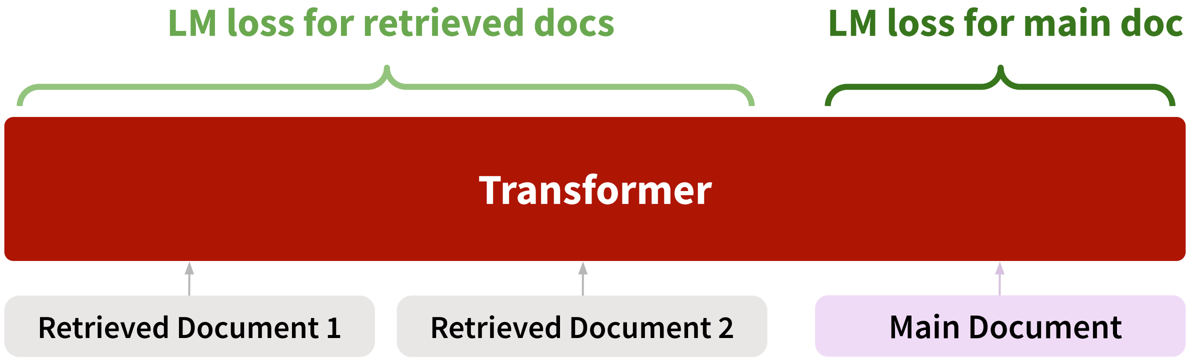
Loss = **(LM loss for main doc)** + α • **(LM loss for retrieved docs)**

- Existing retrieval augmented LMs: α = 0
- **Our method: α > 0 (α = 0.1 works the best)**

α > 0 has effect like increasing batch size without extra forward compute, increasing training efficiency

α > 0 is crucial in multimodal setting

- Each image takes many tokens (1024)
- If α = 0, we are throwing away a lot of compute



Text to Image

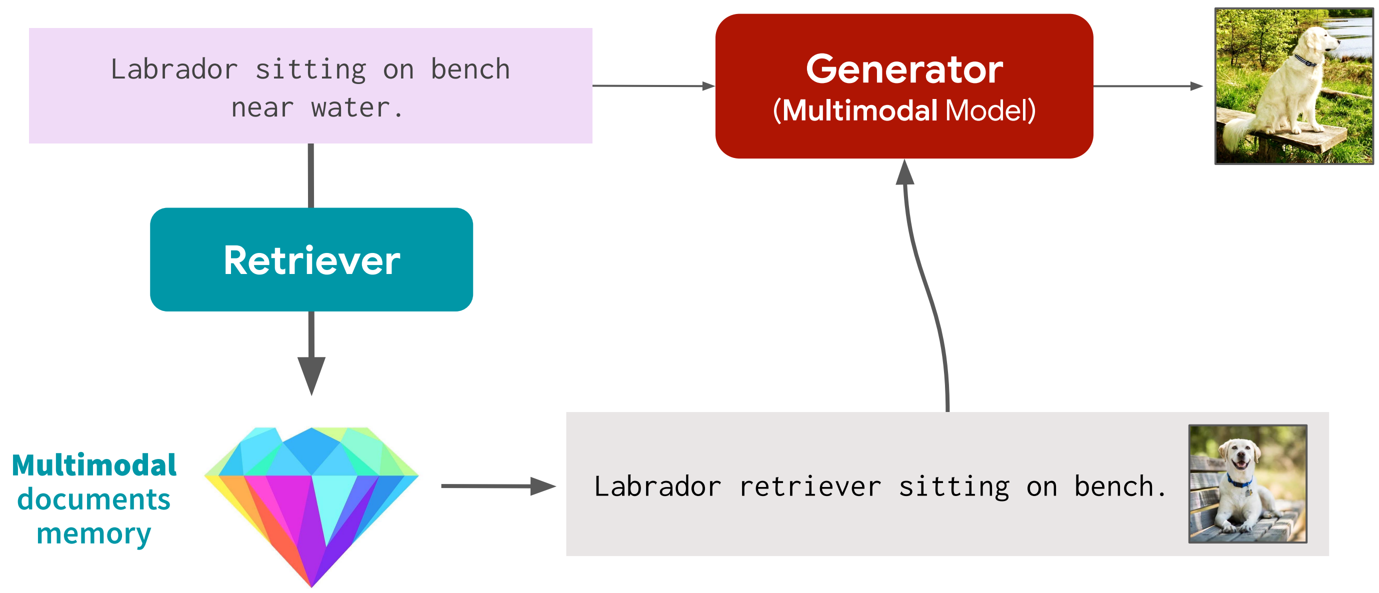
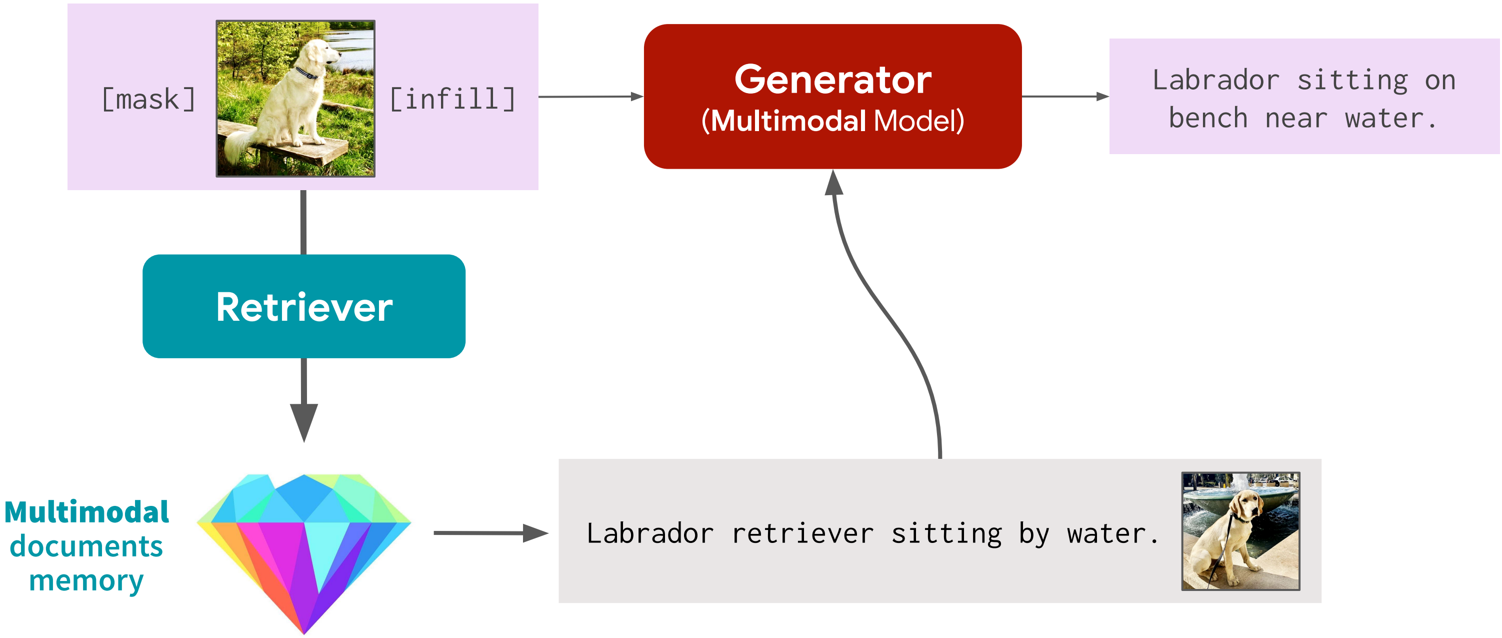
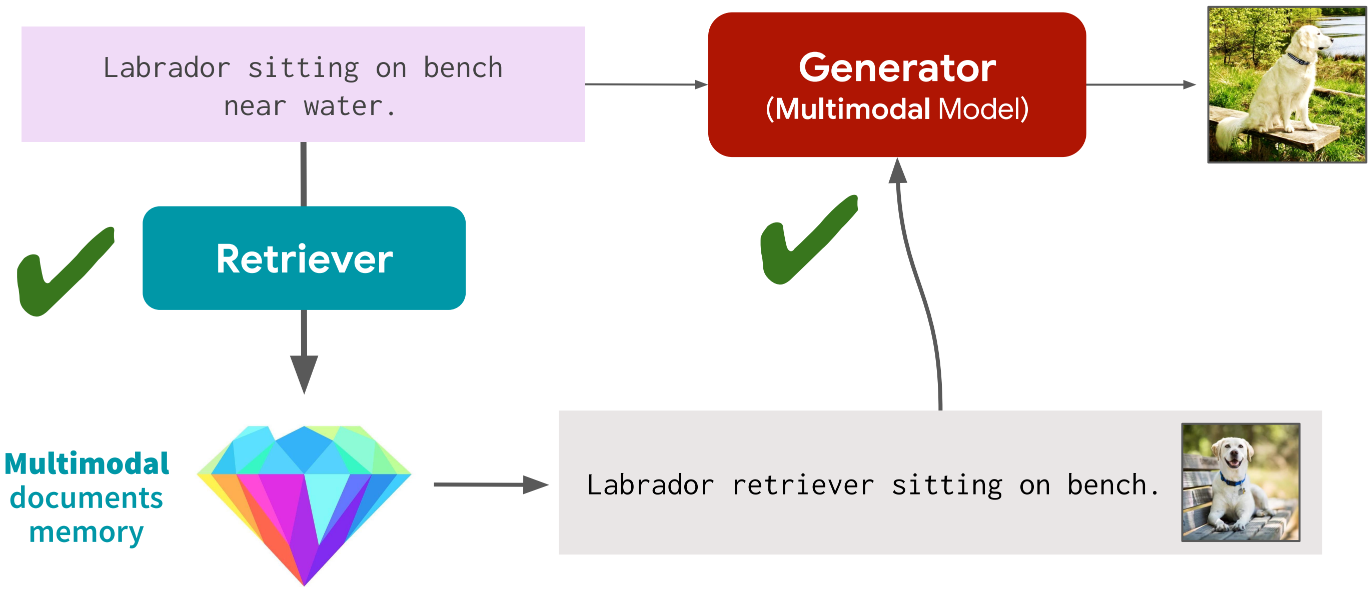


Image to Text



Retrieval Augmented Multimodal Model



Experiments

Train data

- **LAION** (cleaned 150M image-text pairs)
  External memory: LAION

Evaluation

- **MSCOCO** caption2image, image2caption.
  External memory: MSCOCO train set

Model

- Transformer with seq_length 4096 (up to 2 retrieved documents)
- 2.7B parameters trained for 5 days on 256 GPUs
- **“Retrieval Augmented CM3 (RA-CM3)”**

Baseline

- Vanilla CM3 with no retrieval, same size, trained using the same amount of compute

Performance

MSCOCO caption to image generation

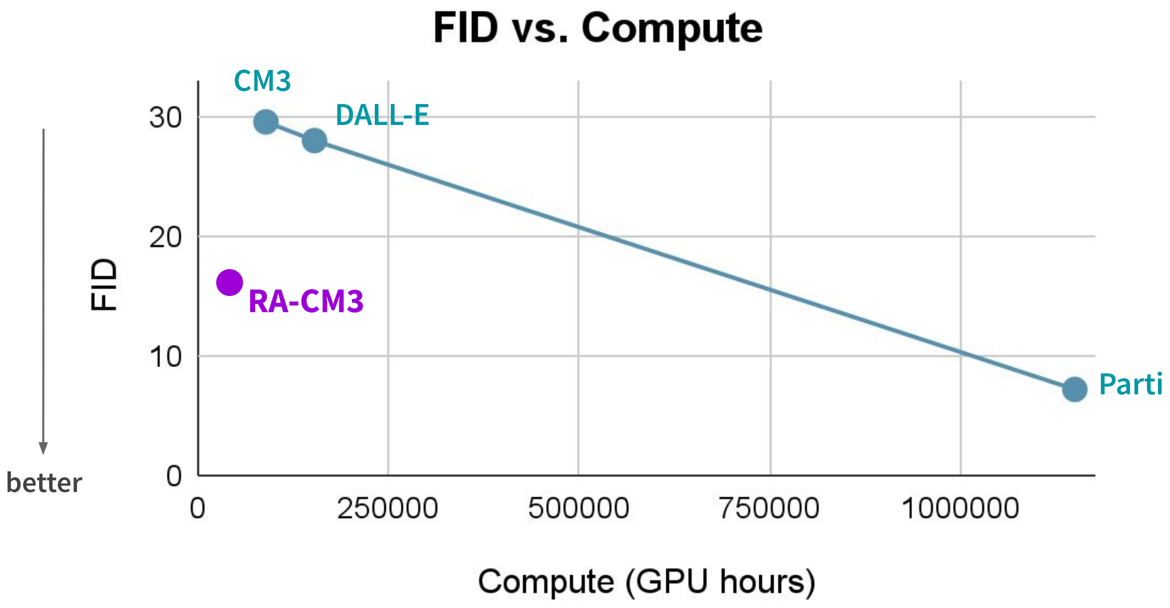
- RA-CM3 outperforms vanilla CM3 as well as DALL-E (which uses more params and images)

Model	Model type	#Train images	FID score (↓)
DALL-E (12B)	Autoregressive	250M	28
Parti (20B)	Autoregressive	6B	7.2
Stable Diffusion	Diffusion	1B	~12
DALL-E 2	Diffusion	650M	10.3
Vanilla CM3	Autoregressive	150M	25.8
RA-CM3	Autoregressive	150M	**16.9**

Performance

MSCOCO caption to image generation

- RA-CM3 is more compute efficient than non-retrieval models like DALL-E, CM3, Parti



Performance

MSCOCO image to caption generation

- RA-CM3 outperforms vanilla CM3 and Flamingo (equivalent size); competitive with Parti

Model	#Train images	CIDEr score (↑)
Parti (20B)	6B	0.89
Flamingo (3B) 4-shot	2.5B	0.85
Flamingo (80B) 4-shot	2.5B	1.03
PaLI (17B)	10B	~1.4
Vanilla CM3	150M	0.72
RA-CM3	150M	**0.89**

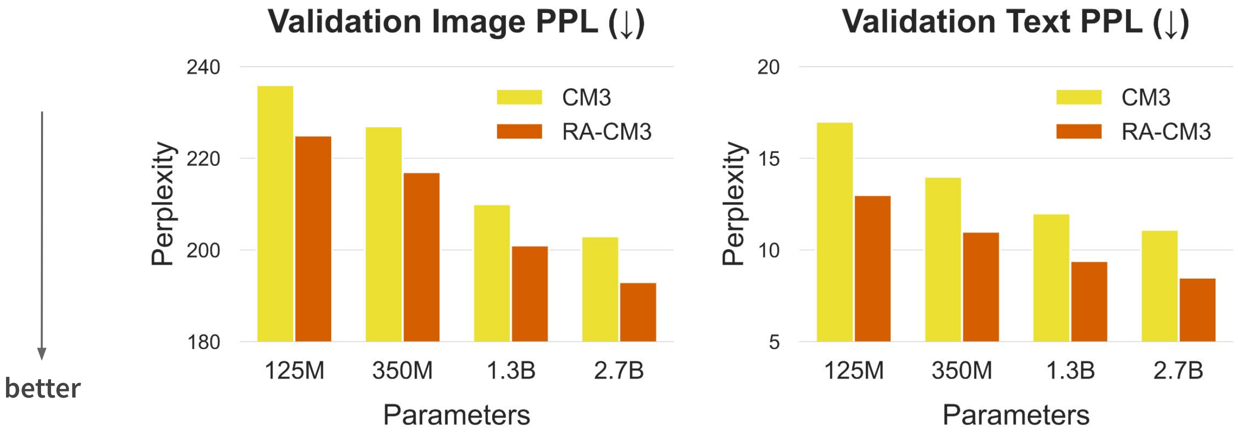
Scaling Laws

Setup

- All models are trained for 2 days on 256 GPUs. Evaluation metric is validation perplexity

Observation

- Retrieval augmentation is consistently helpful across scales
- Larger models perform better (even under the same compute budget)



Ablation Study

Key factors in obtaining retrieved docs (relevance, diversity)

Ablation: Relevance	Val PPL (↓)
Random	130
CLIP-based retrieval (Final)	**121**

Ablation: Diversity	Val PPL (↓)
Simply take top docs	131
Avoid redundant docs	125
Avoid redundant docs + Query dropout (Final)	**121**

Comment

- Diversity is important in the multimodal setting, because each image takes many tokens. Redundant image can significantly hurt model training (model learns to repeat stuff)

Ablation Study

Training objective

- Loss = (LM loss for main doc) + α • (LM loss for retrieved docs)

Loss function	Val PPL (↓)
$\alpha = 0$	127
$\alpha = 1$	126
$\alpha = 0.3$	123
$\alpha = 0.1$ (Final)	**121**

$\alpha > 0$ has effect like increasing batch size without extra forward compute, increasing training efficiency.

But $\alpha = 1$ puts too much weight in modeling retrieved docs, hurting the PPL of the main doc.

Comment

- $\alpha > 0$ is especially effective in the multimodal setting, because each image takes many tokens. If $\alpha = 0$, we are throwing away a lot of compute that could be leveraged

Capabilities

- Knowledge-intensive image generation
- Image infilling & editing
- Controlled image generation
- One/few-shot image classification

Capability: Knowledge-intensive Generation

RA-CM3 In-context

RA-CM3 outputs

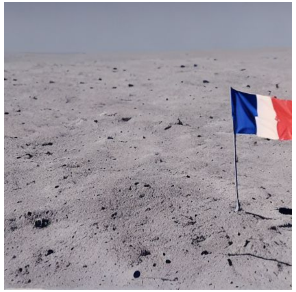
Baseline outputs

(Vanilla CM3) (Stable Diffusion)

Ming Dynasty vase

A **Ming Dynasty vase** with orange flowers painted.

French flag



French flag waving on the moon's surface.

Capability: Knowledge-intensive Generation

RA-CM3 In-context

RA-CM3 outputs

Baseline outputs

(Vanilla CM3) (Stable Diffusion)

Armenian church

An **Armenian church** during a sunny day.

Mount Rainier

People standing in front of the **Mount Rainier**.

Capability: Knowledge-intensive Generation

RA-CM3 In-context

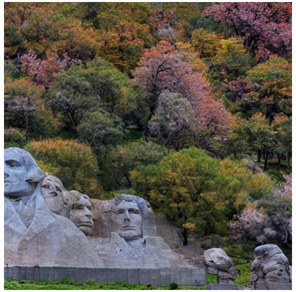
RA-CM3 outputs

Baseline outputs

(Vanilla CM3) (Stable Diffusion)

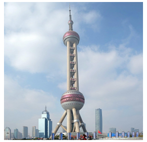
Mount Rushmore

Japanese cherry



The **Mount Rushmore** with **Japanese cherry** trees in the front.

Oriental Pearl tower



The Oriental Pearl tower in oil painting.

Capability: Knowledge-intensive Generation

RA-CM3 In-context

RA-CM3 outputs

Baseline outputs

(Vanilla CM3) (Stable Diffusion)

Callanish standing stones

Photo of the **Callanish standing stones**, fireworks in the sky.

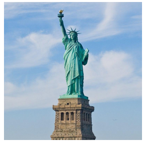
Dragon and Tiger Pagodas

Photo of **the Dragon and Tiger Pagodas**, the sun is setting behind.

Capability: Knowledge-intensive Generation

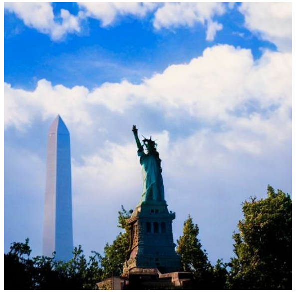
RA-CM3 In-context

Statue of Liberty



Washington monument

RA-CM3 outputs



Baseline outputs

(Vanilla CM3)

(Stable Diffusion)

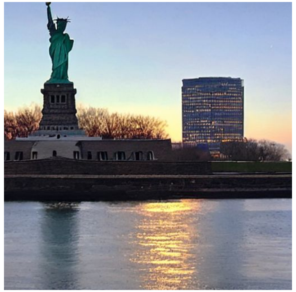


Photo of **the Statue of Liberty** standing next to **the Washington monument**.

Conclusion

Aspirational goal: A core model that fits in a single GPU, but can

- easily access additional, task-related knowledge via retrieval, and
- perform comparably to the largest language models available today

Key to success:

- Large quality data
- Efficient retrieval infrastructure
- Effective communication channels between retrieval and core LM models

Thank you!

Questions?