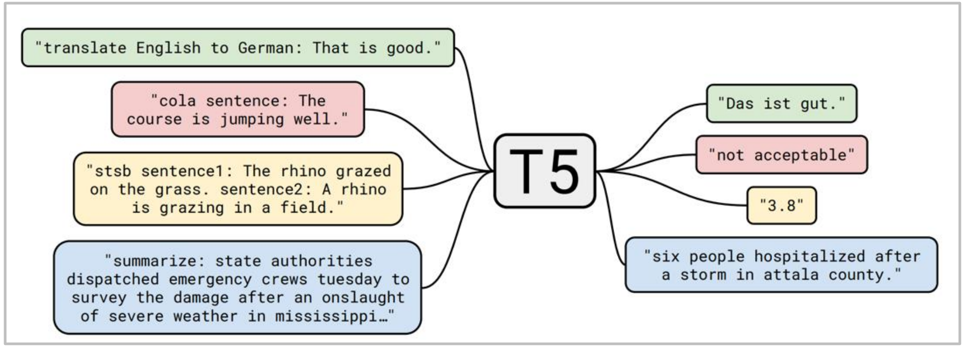
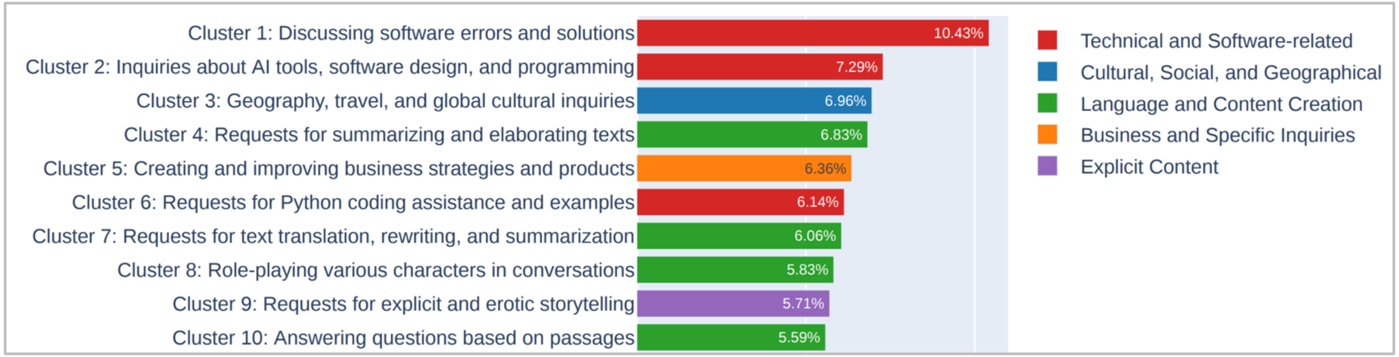
Improving LLM generalization by selecting and synthesizing data

Tatsunori Hashimoto

Language models are great at cross-task generalization



[Raffel et al 2020]

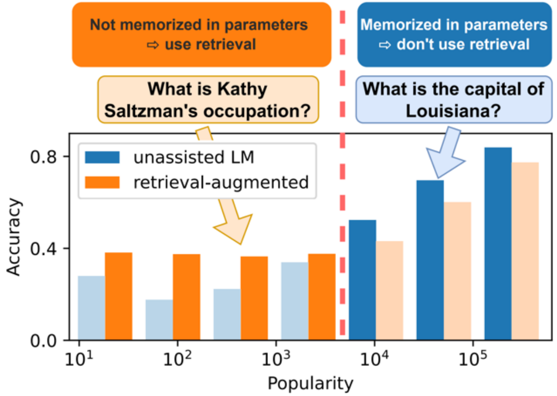


[Zheng et al 2024]

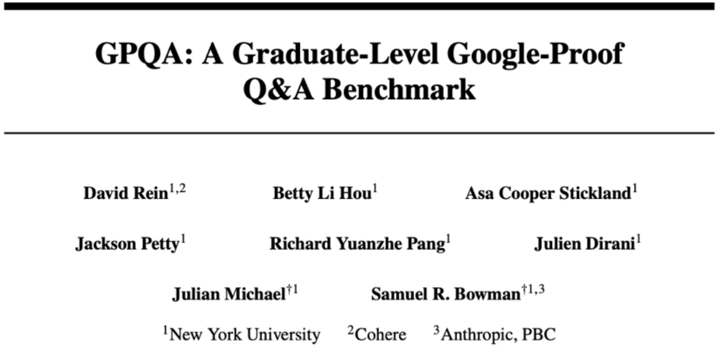
Language models generalize to an enormous range of tasks

… but not everything is in-domain for pretraining

Niche entities



Cutting-edge knowledge



GPQA: A Graduate-Level Google-Proof Q&A Benchmark

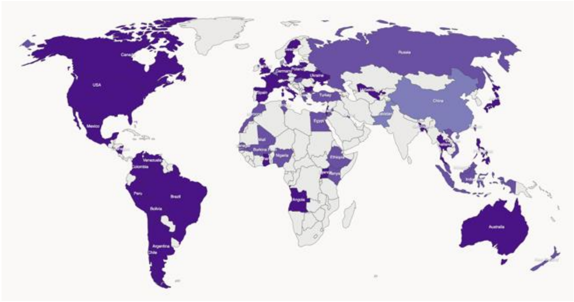
David Rein[1,2] Betty Li Hou[1] Asa Cooper Stickland[1]

Jackson Petty[1] Richard Yuanzhe Pang[1] Julien Dirani[1]

Julian Michael[†1] Samuel R. Bowman[†1,3]

[1]New York University [2]Cohere [3]Anthropic, PBC

Culture



Is pretraining really similar to our downstream tasks?

A naïve mental model..

Pretraining (StackExchange)

Implementing Miller-Rabin in C

Asked 7 years, 5 months ago   Modified today   Viewed 3k times

3

I'm trying to implement the Miller-Rabin primality test in C99, but I'm coming across some problems getting it to work. I crafted a small test-set to verify whether or not the implementation works, here's how I'm checking for primes

```c
int main() {
    int foo[11] = {0, 1, 2, 3, 4, 7, 28, 73, 125, 991, 1000};
    for (int i = 0; i < 11; i++) {
        printf("%s; ", isprime(foo[i], 5000) ? "Yes" : "No");
    }
    return 0;
}
```

From the numbers listed, the expected output would be

> No; No; Yes; Yes; No; Yes; No; Yes; No; Yes; No;

However, as implemented , the output I get is the following:

Evaluation data (HumanEval)

```python
def incr_list(l: list):
    """Return list with elements incremented by 1.
    >>> incr_list([1, 2, 3])
    [2, 3, 4]
    >>> incr_list([5, 3, 5, 2, 3, 3, 9, 0, 123])
    [6, 4, 6, 3, 4, 4, 10, 1, 124]
    """
    return [i + 1 for i in l]
```

```python
def solution(lst):
    """Given a non-empty list of integers, return the sum of all of the odd elements
    that are in even positions.

    Examples
    solution([5, 8, 7, 1]) ==>12
    solution([3, 3, 3, 3, 3]) ==>9
    solution([30, 13, 24, 321]) ==>0
    """
    return sum(lst[i] for i in range(0,len(lst)) if i % 2 == 0 and lst[i] % 2 == 1)
```

Part 1: Fixing the pretraining vs downstream task gap

The reality: pretraining data

000 084 in Software Title

1. 000-084 Test Prep Training Premier ... Servers Technical Support V3 000-084 exam preparation. With our 000-084 study notes, you can ... to take on your 000-084 Exam.All of our tests including the 000-084 e challenging test, with... Details - Download - Screenshot

Tags: 000 084 exam , our 000 084 , high technical expertise , exam 000 084 , 000 084 study , 000 084 , 084 exam , our 000 , actual exam , technical expertise

2. Pass4sure IBM 000-084 2012 ... rewarding features of the 000-084 training materials are that ... blues. Prepare our IBM 000-084 exam questions and answers ... products provide a basic 000-084 practice test to exam questions... Details - Download - Screenshot

Tags: IBM 000-084 , Download 000-084 , 000-084 PDF , IBM 000-084 test , 000-084 study guide , 000-084 training , 000-084 braindumps , 000-084 exam dumps , 000-084 cheatsheet , 000-084 study help

3. TopCerts 000-084 Questions and Answers 2.0 Download free 000-084 questions and answers. 000-084 exam questions are ultimate ... for your certification. All 000-084 exam materials are with ... Details - Dov

Tags: 000 084 Exam Demo , 000 084 Exam , 000 084 Questions , 000 084 Questions And Answers , Free 000 084 Questions And Answers , 000 084 Exam Questions , Free 000 084 Questions

4. HP0-084 Free Practice Exam Questions 13.0 ... provided a free HP0-084 free practice exam where ... goes into our HP0-084 practice test questions. Our HP0-084 practice test questions are ... together the best I download our HP0-084 free... Details - Download - Screenshot

Tags: hp0 084 practice , our hp0 084 , 084 practice exam , pass guaranteed hp0 , guaranteed hp0 084 , 084 practice test , practice test questions , hp0 084 free , hp0 084 exam , exam pass guaranteed

5. Pass4sure ADOBE 9A0-084 2012 ... features of the 9A0-084 training materials are that ... Prepare our ADOBE 9A0-084 exam questions and answers ... provide a basic 9A0-084 practice test to prepare ... surpri complete ... Details - Download - Screenshot

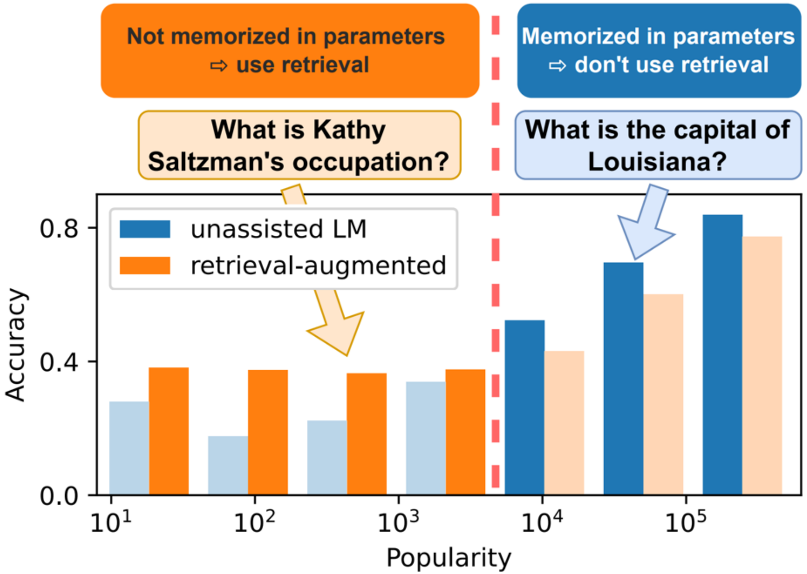
Tags: 9A0-084 , ADOBE 9A0-084 , Download 9A0-084 , 9A0-084 PDF , ADOBE 9A0-084 test , 9A0-084 study guide , 9A0-084 training , 9A0-084 braindumps , 9A0-084 exam dumps , 9A0-084 cheatsheet

Pretraining isn't generalization magic – it's built on careful, hand-engineered data

Can we avoid hand-crafted data selection?

First page in a common crawl dump - http://000-084.smartcode.com/

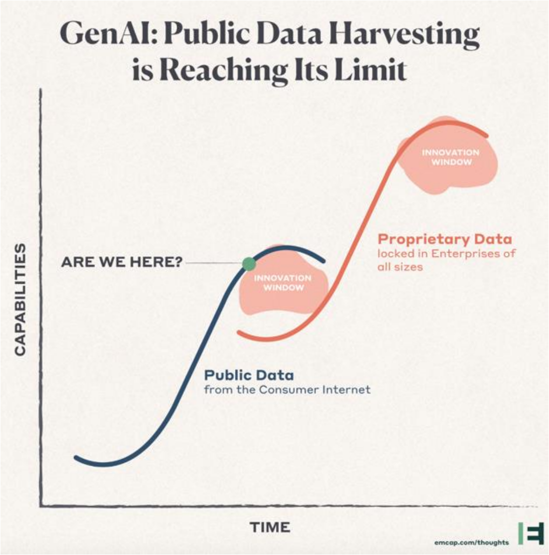
Fixing the knowledge gap for data-constrained, tail domains

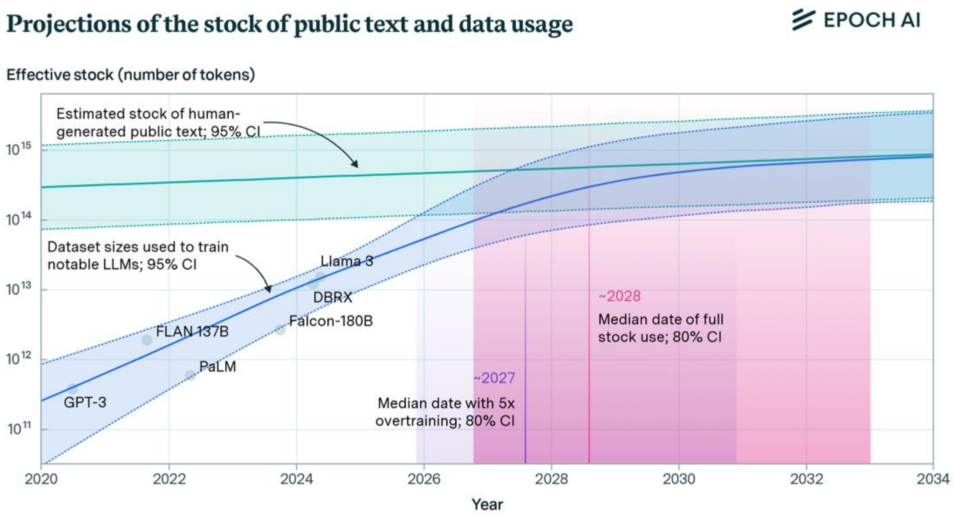


[Mallen+ 2023]

LLMs performance depends on plentiful data in the 'head' of the distribution
performance is limited in the tails

Part 2: Data efficient (continued) pretraining



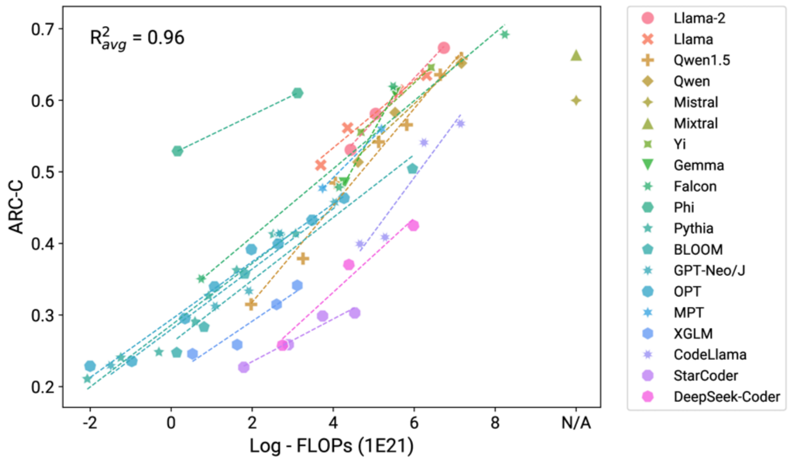


Can we build more data-efficient ways of pretraining?

Enabling ‘tail’ knowledge and going past the looming data barrier

Our approach: building on what works

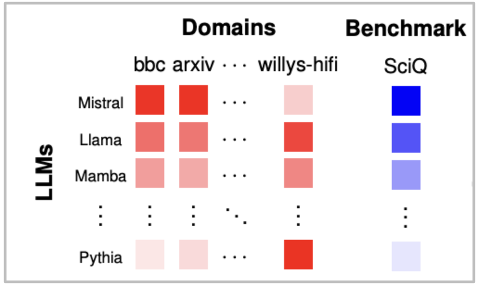
We know that the modern, pretraining paradigm is effective – how can we work with it?



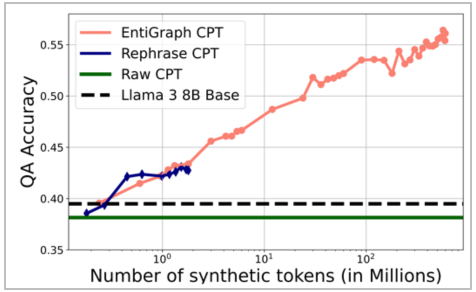
Can we use the modern pretraining paradigm to address domain and task mismatch issues?

Part 1: Data Selection

Can we close the pretraining-task distribution gap
(without extensive human effort)



Part 1: Data selection for pretraining



Part 2: Data synthesis for domain adaptation


Tristan Thrush, Chris Potts, Tatsunori Hashimoto – Improving Pretraining Data Using Perplexity Correlations

Pretraining data (at scale) is key to good, pretrained LMs

Meta

The Llama 3 Herd of Models

Llama Team, AI @ Meta[1]

We believe there are three key levers in the development of high-quality foundation models: data, scale, and managing complexity. We seek to optimize for these three levers in our development process:

- **Data.** Compared to prior versions of Llama (Touvron et al., 2023a,b), we improved both the quantity and quality of the data we use for pre-training and post-training. These improvements include the development of more careful pre-processing and curation pipelines for pre-training data and the development of more rigorous quality assurance and filtering approaches for post-training data. We pre-train Llama 3 on a corpus of about 15T multilingual tokens, compared to 1.8T tokens for Llama 2.
- **Scale.** We train a model at far larger scale than previous Llama models: our flagship language model was pre-trained using 3.8×10^{25} FLOPs, almost $50\times$ more than the largest version of Llama 2. Specifically, we pre-trained a flagship model with 405B trainable parameters on 15.6T text tokens. As expected per

What makes pretrained LMs work? Data, scaling (and attention to detail)

But what *works* is incredibly ad-hoc (and often secret)

From LLaMA 3.1

> We create our dataset for language model pre-training from a variety of data sources containing knowledge until the end of 2023. We apply several de-duplication methods and data cleaning mechanisms on each data source to obtain high-quality tokens. We remove domains that contain large amounts of personally identifiable information (PII), and domains with known adult content.

From Datacomp-LM

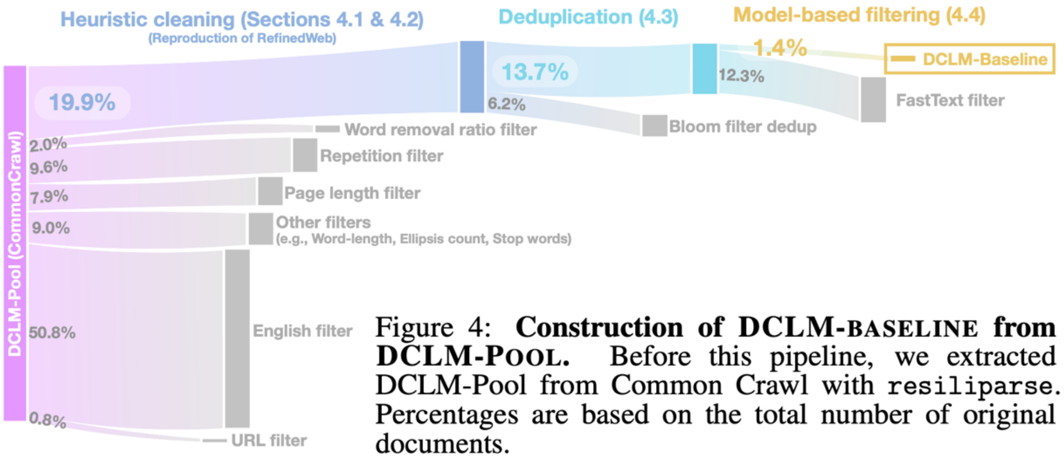


Figure 4: **Construction of DCLM-BASELINE from DCLM-POOL.** Before this pipeline, we extracted DCLM-Pool from Common Crawl with `resiliparse`. Percentages are based on the total number of original documents.

Can we get simple, principled pretraining data selection?

Current (open) SoTA: Bigram classifier based on ELI5 + OH. What is that?

> RedPajama-books [160], following the reference data used for GPT-3 [30]. We also try a novel approach, using instruction-formatted data, drawing examples from OpenHermes 2.5 [157] (OH-2.5) and high-scoring posts from the r/ExplainLikeImFive (ELI5) subreddit. Overall, we find, when controlling for other hyperparameters, the fastText OH-2.5 +ELI5 approach gives a 3.5 percentage point lift on CORE compared to the conventional choices. It is natural to ask whether using OH-2.5 data for filtering could preclude additional

This is very unsatisfying – is there a simple, principled alternative?

- **Inputs**: target benchmark(s), token count, pretraining corpus
- **Output**: a data filtering policy

Of course, we are not the first to think this

Datamodels (+scaling)

Datamodels: Predicting Predictions from Training Data

Andrew Ilyas* ailyas@mit.edu MIT
Sung Min Park* sp765@mit.edu MIT
Logan Engstrom* engstrom@mit.edu MIT
Guillaume Leclerc leclerc@mit.edu MIT
Aleksander Mądry madry@mit.edu MIT

Perturb data, train models, build a map from data mix to performance

Datamodel / Shapley [Illyas+ 22, Ghorbani+ 19]
+ Scaling [Hashimoto 21, Woleridge+ 21, Ye 24]

Influence functions (and other local approx)

Understanding Black-box Predictions via Influence Functions

Pang Wei Koh [1] Percy Liang [1]

Build approximations using Taylor approximations of the loss

Influence fns [Koh+ 20, Xia+ 24]
First order approx [Yu+ 24]

And many others..

Robust opt [Xie+ 23], Similarity [Xie+23, Abbas+23, Everaert+ 23]

Challenges in the way

But these algorithms have not changed data selection processes..

Data, Data Everywhere:
A Guide for Pretraining Dataset Construction

Jupinder Parmar*, Shrimai Prabhumoye, Joseph Jennings,
Bo Liu, Aastha Jhunjhunwala, Zhilin Wang, Mostofa Patwary,
Mohammad Shoeybi , Bryan Catanzaro
NVIDIA

Sophisticated data selector (DoReMi) is worse than uniform.
N-gram based (DSIR) leads to slight improvement

DataComp-LM: In search of the next generation of training sets for language models

**Jeffrey Li*[1,2] Alex Fang*[1,2] Georgios Smyrnis*[4] Maor Ivgi*[5]
Matt Jordan[4] Samir Gadre[3,6] Hritik Bansal[8] Etash Guha[1,15] Sedrick Keh[3] Kushal Arora[3]
Saurabh Garg[13] Rui Xin[1] Niklas Muennighoff[22] Reinhard Heckel[12] Jean Mercat[3] Mayee
Chen[7] Suchin Gururangan[1] Mitchell Wortsman[1] Alon Albalak[19,20] Yonatan Bitton[14]
Marianna Nezhurina[9,10] Amro Abbas[23] Cheng-Yu Hsieh[1] Dhruba Ghosh[1] Josh Gardner[1]
Maciej Kilian[17] Hanlin Zhang[18] Rulin Shao[1] Sarah Pratt[1] Sunny Sanyal[4] Gabriel Ilharco[1]
Giannis Daras[4] Kalyani Marathe[1] Aaron Gokaslan[16] Jieyu Zhang[1] Khyathi Chandu[11]
Thao Nguyen[1] Igor Vasiljevic[3] Sham Kakade[18] Shuran Song[6,7] Sujay Sanghavi[4] Fartash
Faghri[2] Sewoong Oh[1] Luke Zettlemoyer[1] Kyle Lo[11] Alaaeldin El-Nouby[2] Hadi
Pouransari[2] Alexander Toshev[2] Stephanie Wang[1] Dirk Groeneveld[11] Luca Soldaini[11]
Pang Wei Koh[1] Jenia Jitsev[9,10] Thomas Kollar[3] Alexandros G. Dimakis[4,21]
Yair Carmon[5] Achal Dave[†3] Ludwig Schmidt[†1,7] Vaishaal Shankar[†2]**

Best selector found by authors – hand-crafted pipeline w/ fasttext classifier

Why is algorithmic data selection so hard?

Cost: It's *very expensive* to get data for this task

Validity: Learned policies may not be robust

Data efficiency: most methods handle ~ 10-50 domains

Starting point – datamodels

Let's walk through a concrete example.

We want to train a new, 7B param LLM to do well on MMLU

We will use a *datamodels* style approach

1. Train 1000 models (slightly smaller than 7B?), each with a different data mix p
2. Measure benchmark performance y for each model
3. Build a regression $p \to y$

Cost: 1000 models (7B sized!)

Validity: regression model needs to generalize

Data efficiency: *at most* 1000 domains (?) or sparse domains

Starting point – datamodels

Let's walk through a concrete example.

We want to train a new, 7B param LLM to do well on MMLU

We will use a *datamodels* style approach

1. Train 1000 models (slightly smaller than 7B?), each with a different data mix p
2. Measure benchmark performance y for each model
3. Build a regression $p \to y$

Cost: 1000 models (7B sized!)

Validity: regression model needs to generalize

Data efficiency: *at most* 1000 domains (?) or sparse domains

The idea: *don't train models*

Don't train models, extract info from publicly available models

- **No cost** – the models are high-perf, trained, and free.
- **Heterogenous –** covers many points on the design space (code, multimodal, etc)
- **Data efficiency** – ~100 models, can fit reasonably complex models

(Only issue – we don't know what data they trained on)

The gameplan – build a loss-to-performance predictor

The challenge: we don't know these models' data!
This turns out to be fine

Step 1: Hypothesize a *single index model* relating log-loss (x) to downstream perf (y)

$$y_i = f(\langle \boldsymbol{\theta}^*, \mathbf{x}_i \rangle + \epsilon_i)$$

Step 2: find (or project) nonnegative weights.

Proposition 1 *Suppose that $\boldsymbol{\theta}^*$ weights are non-negative. Then, for models with associated likelihoods $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^D$, the minimizer of the pretraining loss over the $\boldsymbol{\theta}^*$ sampling distribution $\mathbb{E}_{j \sim \boldsymbol{\theta}^*}[x_j]$ also has the lowest expected downstream error according to the single index model:*

$$\arg\min_{\mathbf{x} \in \mathcal{X}} \mathbb{E}_{j \sim \boldsymbol{\theta}^*}[x_j] = \arg\min_{\mathbf{x} \in \mathcal{X}} \mathbb{E}[f(\langle \boldsymbol{\theta}^*, \mathbf{x} \rangle + \epsilon)].$$

If we can find good, nonnegative single-index models relating loss to perf. ,
sampling according to these weights is a good data selection policy

The two steps as an algorithm

Step 1 : fitting the regression – we use a high dimensional regression estimator

$$\gamma_j = \sum_{\substack{1 \le k,l \le n \\ k \neq l}} \operatorname{sign}(y_k - y_l)(\operatorname{rank}_j(x_{k,j}) - \operatorname{rank}_j(x_{l,j}))$$

(we will show that this is actually a consistent estimate of the single index model)

Step 2: selecting the data (projection) – select tokens from largest to smallest γ

for i $\in$**ArgSort**(γ, descending=True) **do** $\triangleright$ 2. Select most to least correlated domains
    $t_i \leftarrow \min(a_i, b - \text{counter})$
    counter $\leftarrow$ counter $+ a_i$
    if counter $\geq b$ **then**
        Break
classifier = trainFastText(positive $= 1_{t>0}$, negative $= 1_{t=0}$)

Why should this work? A high-dim stats perspective

This is (just) a variant of high-dim geometric estimation problem.

From Plan, Vershinyn, and Yudovina 2016,

Assuming $y_i = f(\langle \boldsymbol{\theta}^*, \mathbf{x}_i \rangle + \epsilon_i)$, with $\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for $\|\boldsymbol{\theta}^*\|_2 = 1$

$$\mathbb{E}\left[y_k \mathbf{x}_k\right] = c\boldsymbol{\theta}^*$$

And, in a follow-up Chen and Banerjee 2017 showed

$$\mathbb{E}\left[\operatorname{sign}(y_k - y_l)(\mathbf{x}_k - \mathbf{x}_l)\right] = \beta\boldsymbol{\theta}^*$$

Which is, course quite similar to $\gamma_j = \sum_{\substack{1 \leq k,l \leq n \\ k \neq l}} \operatorname{sign}(y_k - y_l)(\operatorname{rank}_j(x_{k,j}) - \operatorname{rank}_j(x_{l,j}))$

Our robust, moment-based estimator

It turns out that this similarity goes deeper – our 'correlation estimate' is consistent

Theorem 1 *When* $\epsilon \sim \mathcal{N}(0, \sigma^2)$*, we have:*

$$\mathbb{E}[\text{sign}(y_i - y_j)(\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j))] = \frac{2}{\pi} \sin^{-1}\left(\frac{\boldsymbol{\theta}^*}{2\sqrt{1+\sigma^2}}\right).$$

And we can get a constrained estimate via a linear projection (following Chen and Banerjee)

$$\hat{\boldsymbol{\theta}}^{\text{proj}} = \underset{\boldsymbol{\theta} \in \mathbb{R}^D}{\arg\min} -\langle \boldsymbol{\theta}, \hat{\boldsymbol{\theta}} \rangle,$$

subject to:

$$\sum_{i=1}^{D} \theta_i = 1$$

$$0 \leq \theta_i \leq \tau_i, \forall i \in [1, D],$$

This has a simple closed form solution (sort and take tokens til budget)

Validation strategy

Recall our goal: select pretraining data so our LMs do well on target benchmarks

Our validation:

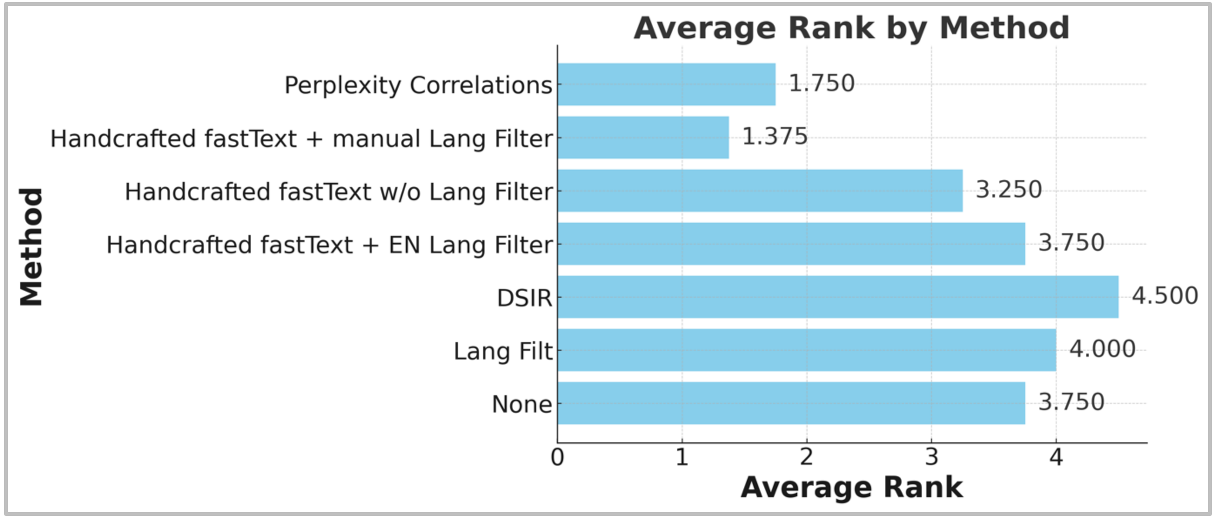
- Estimate perplexity correlations (on ~90 public models)
- Do selection on 8 benchmarks (ARC,SciQ,LAMBADA,PIQA,LAMBADA (FR/DE/IT/ES))
- Train and evaluate corresponding models (at small, **160M** scale)

What do we compare to?

- Selection methods validated at scale (DCLM fasttext classifier, DSIR)
- Reasonable baselines (language filtering)
- No filtering

Selecting pretraining data

So, how good is this correlation-based filtering technique?



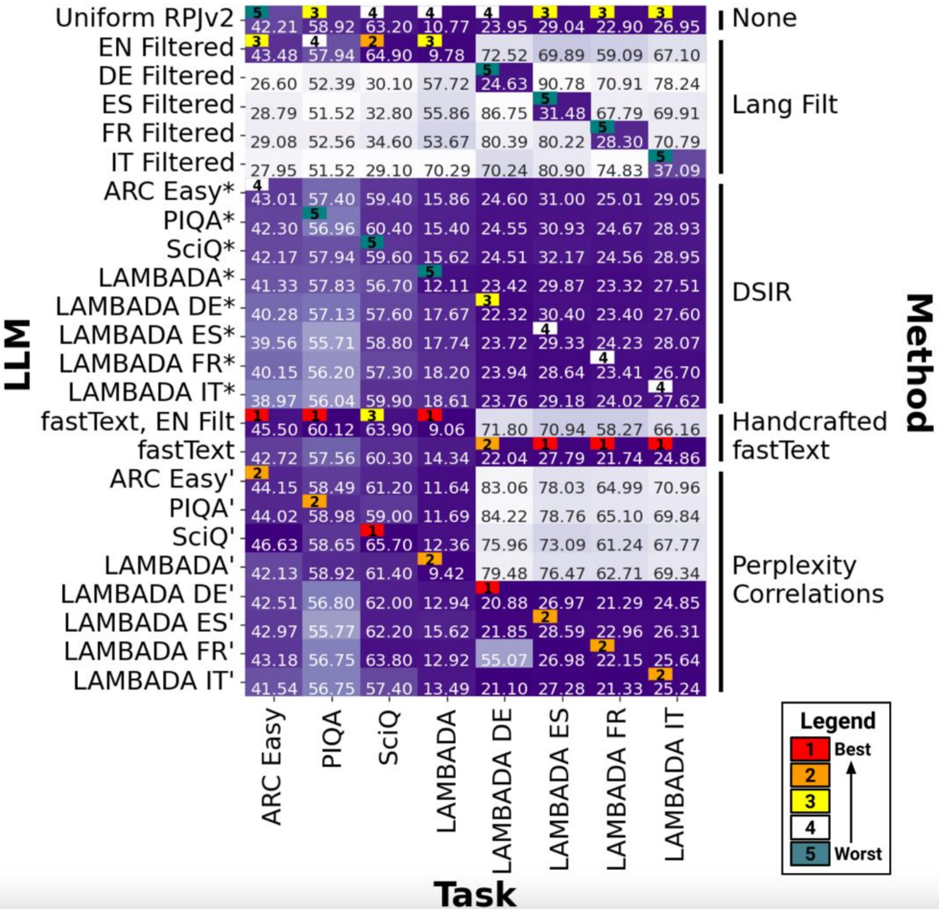
Some observations

- Most filters (language, DSIR) worse than nothing.
- fastText w/o manual language filter is slightly better
- Our approach is significantly better (1.75)
- Slightly worse than best filter w/ manual lang. filter

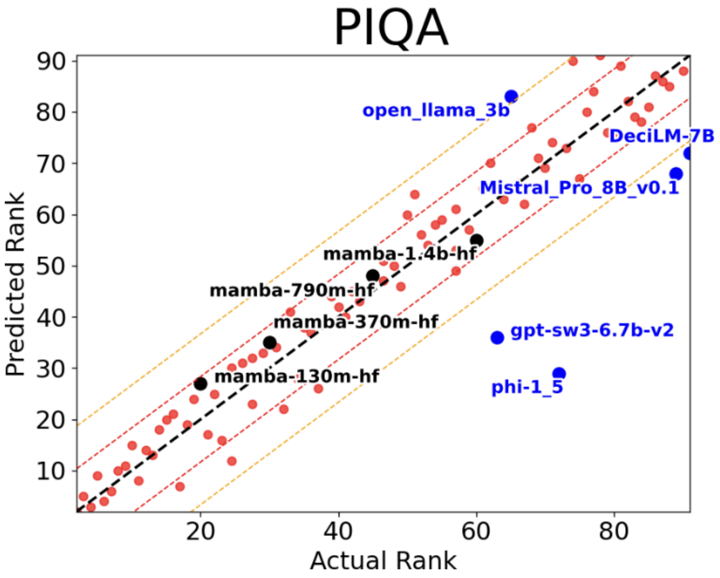
Per-benchmark

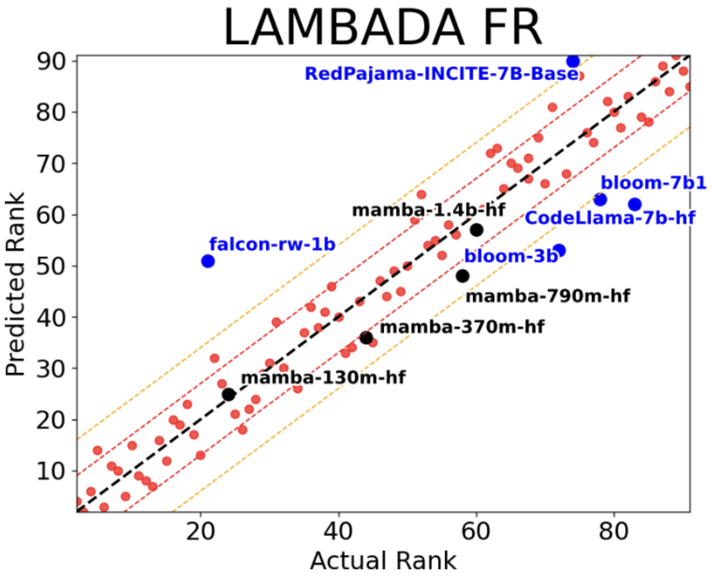
Let’s look at more fine-grained performance.

- Perplexity correlations automatically select by language
- But language filtering is quite bad – only slightly better than random
- When perplexity correlation is not 1st, it’s a close 2nd



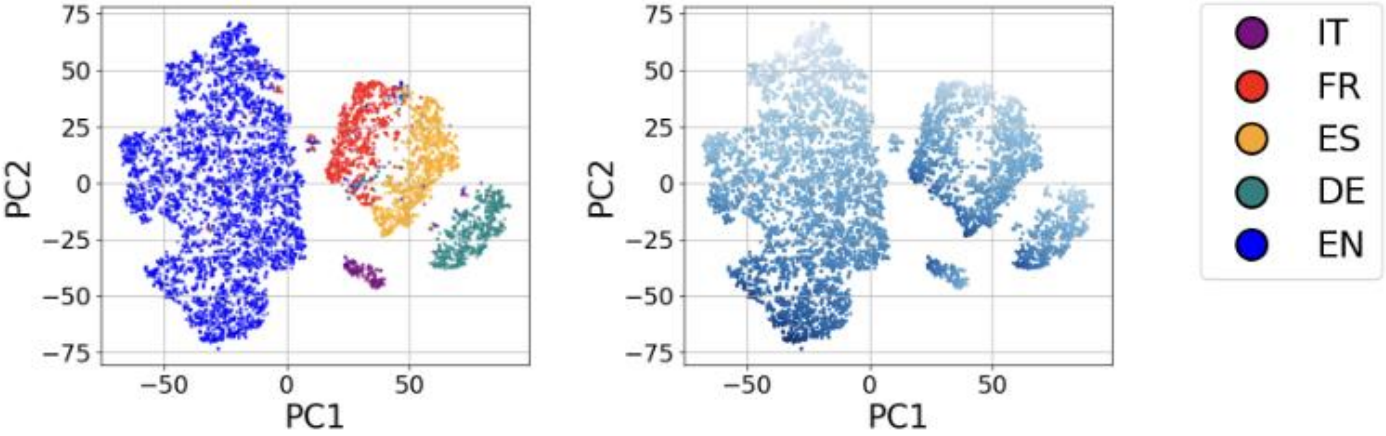
For many benchmarks, perplexity predicts performance





A weighted sum of pretraining document losses accurately predicts rankings

Looking inside the log-loss matrix



T-SNE and PCA (not shown) show meaningful structures about data in the loss matrix

Preregistration-based validation

Can we trust any of these results?

- Many past results have not held up
- Small scale of the experiments
- n=1 in choice of pretraining data pool, etc.

What we're trying: preregistration-based scaling

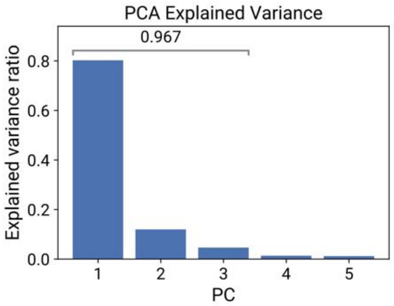
- Scale up by ~ 100x in compute
- Pick a standard, held-out setting with strong baselines (DCLM) we haven't tried
- Use same / preregistered hyperparams
- Report results *regardless of outcome*

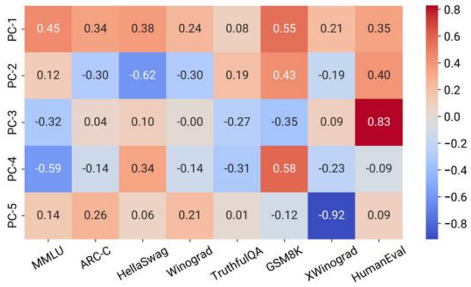
(side note – I'm excited about doing better, rigorous empirical scaling work via preregistration)

Preregistration can help better empirical studies

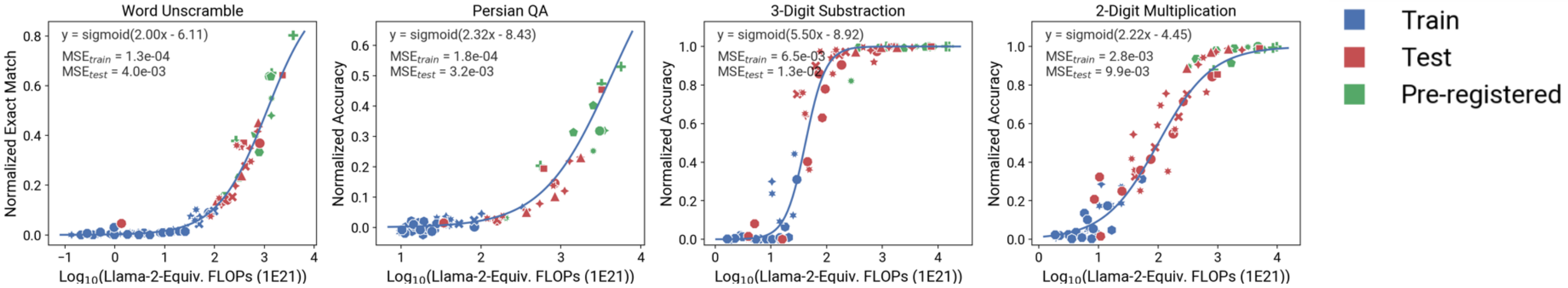
Prior example – observational studies into benchmark-model correlations [Ruan+ 2024]

1. Just a few principal components cover the space of many LM benchmarks





2. These few PCs then robustly explain complex, phenomena



Takeaways – data selection

Data selection is important but hard..

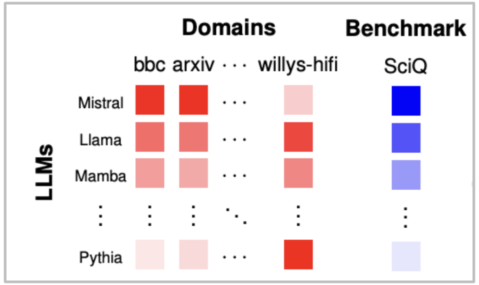
Can we reduce it to a standard high-dim. regression problem?

Maybe. Important ingredients -

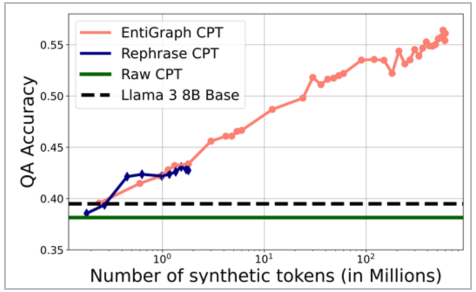
- Single index model + loss optimization
- Robust, high-dimensional single index model estimate
- Small-scale validation + preregistered scaling

Part 2: Data synthesis

Can we teach a language model
new, niche knowledge?



Part 1: Data selection for pretraining

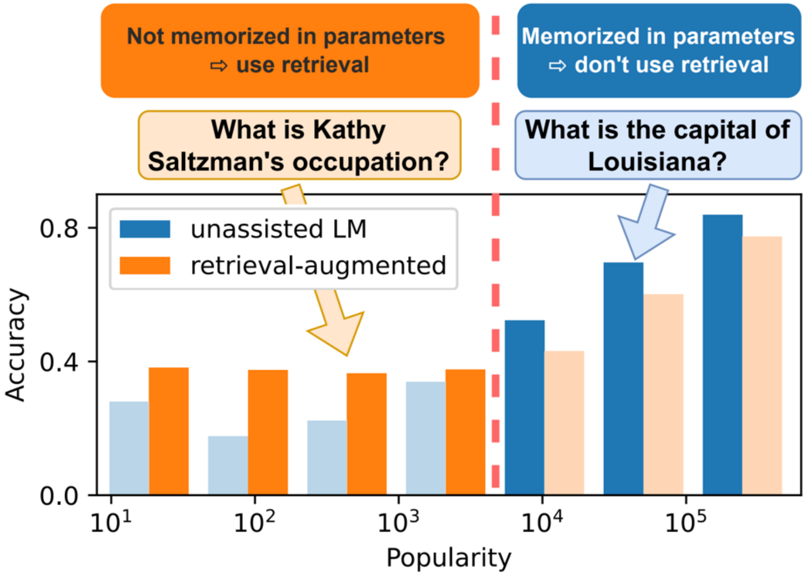


Part 2: Data synthesis for domain adaptation




Zitong Yang*, Neil Band*, Shuangping Li, Emmanuel Candes, Tatsunori Hashimoto, Synthetic continued pretraining

LLMs struggle beyond the ‘head’ of the distribution



[Mallen+ 2023]

LLMs performance depends on plentiful data in the ‘head’ of the distribution
performance is limited in the tails

‘Adapting’ to the tails – difficult for data-poor domains

The standard approach – domain adaptation via continued pretraining

Study	Domain	Model Parameter Count	Total Unique CPT Tokens
Minerva (Lewkowycz et al., 2022)	STEM	8B, 62B, 540B	26B-38.5B
MediTron (Chen et al., 2023)	Medicine	7B, 70B	46.7B
Code Llama (Rozière et al., 2024)	Code	7B, 13B, 34B	520B-620B
Llemma (Azerbayev et al., 2024)	Math	7B, 34B	50B-55B
DeepSeekMath (Shao et al., 2024)	Math	7B	500B
SaulLM-7B (Colombo et al., 2024b)	Law	7B	30B
SaulLM-{54, 141}B (Colombo et al., 2024a)	Law	54B, 141B	520B
HEAL (Yuan et al., 2024a)	Medicine	13B	14.9B

Teaching models new facts in a way that can be internalized and generalized requires ~ 15+ Billion tokens with current methods

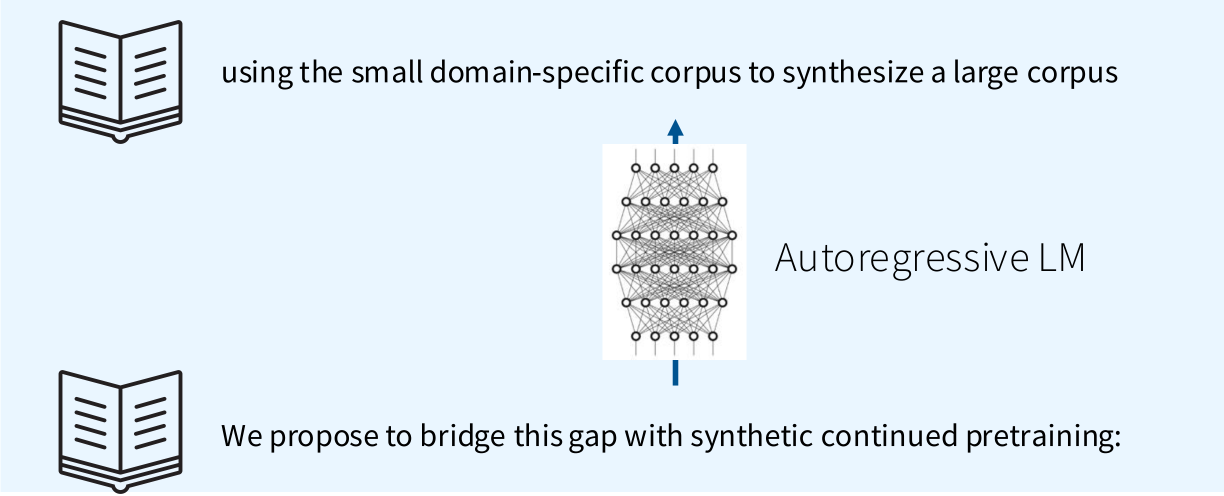
Our challenge: learning from 10,000x less data

Study	Domain	Model Parameter Count	Total Unique CPT Tokens
Minerva (Lewkowycz et al., 2022)	STEM	8B, 62B, 540B	26B-38.5B
MediTron (Chen et al., 2023)	Medicine	7B, 70B	46.7B
Code Llama (Rozière et al., 2024)	Code	7B, 13B, 34B	520B-620B
Llemma (Azerbayev et al., 2024)	Math	7B, 34B	50B-55B
DeepSeekMath (Shao et al., 2024)	Math	7B	500B
SaulLM-7B (Colombo et al., 2024b)	Law	7B	30B
SaulLM-{54, 141}B (Colombo et al., 2024a)	Law	54B, 141B	520B
HEAL (Yuan et al., 2024a)	Medicine	13B	14.9B
Our setting	Articles & Books	7B	1.3M

Can we adapt to knowledge that might be truly in the tail?
Few hundred books with **10,000x less data**

Problems with standard continued pretraining

Standard continued pretraining: train directly on our documents



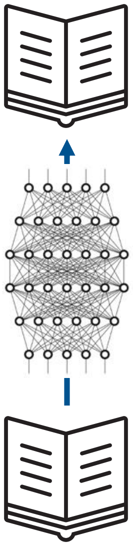
Autoregressive learning is data-inefficient (reversal curse)

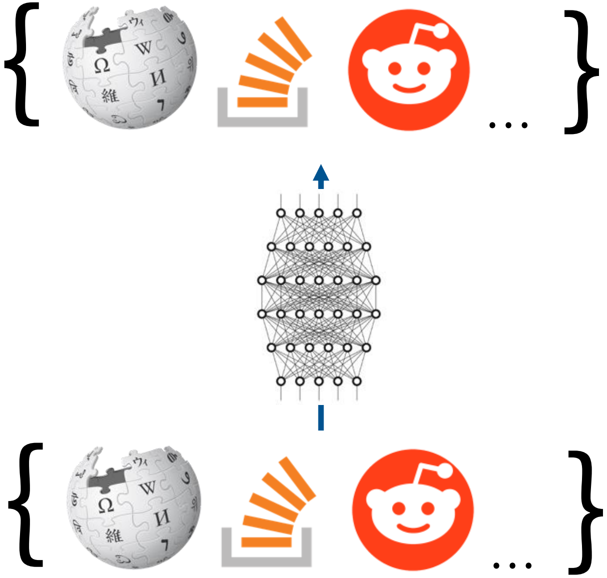
In the autoregressive direction: “What does synthetic CPT do?”

In the reverse direction: “What method synthesizes a large corpus?”

Differences from pretraining

Why doesn't continued pretraining work?





CPT: limited diversity (format, content)

Pretraining: diverse formats

Synthetic continued pretraining – augment the data

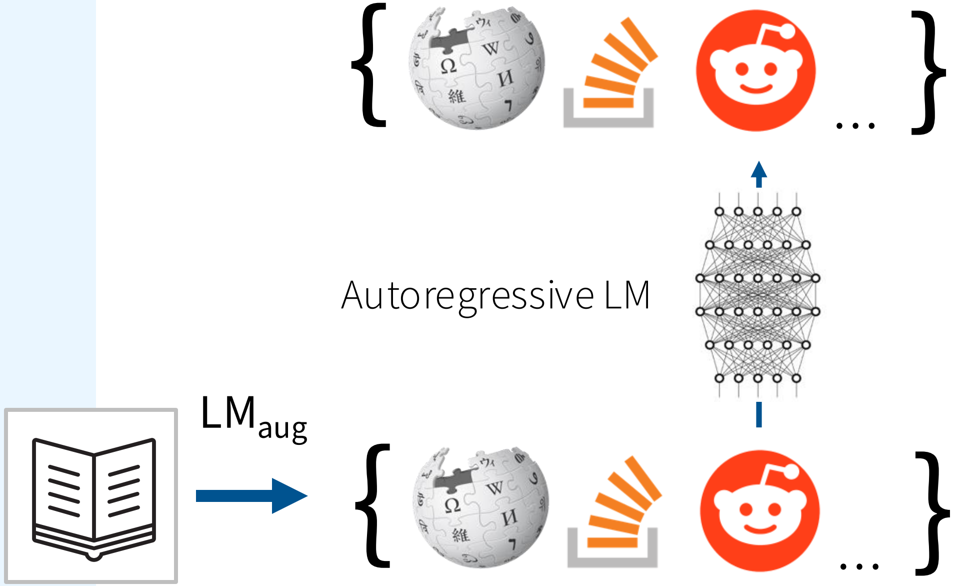
***Synthetic* continued pretraining:** Train on LLM-transformed data

Goal – replicate the diversity of pretraining

- Vary content (topics)
- Vary style (how it's presented)
- Data diversity for generalization

This is *different* from synthetic data or..

- compute / size efficiency (WRAP/Phi)
- fine-tuning (task-specific LMs)



The setting – QuALITY books

- Project Gutenberg fiction stories (mostly science fiction)
- Slate magazine articles from the Open American National Corpus
- Nonfiction articles taken from The Long and Short, Freesouls, etc,

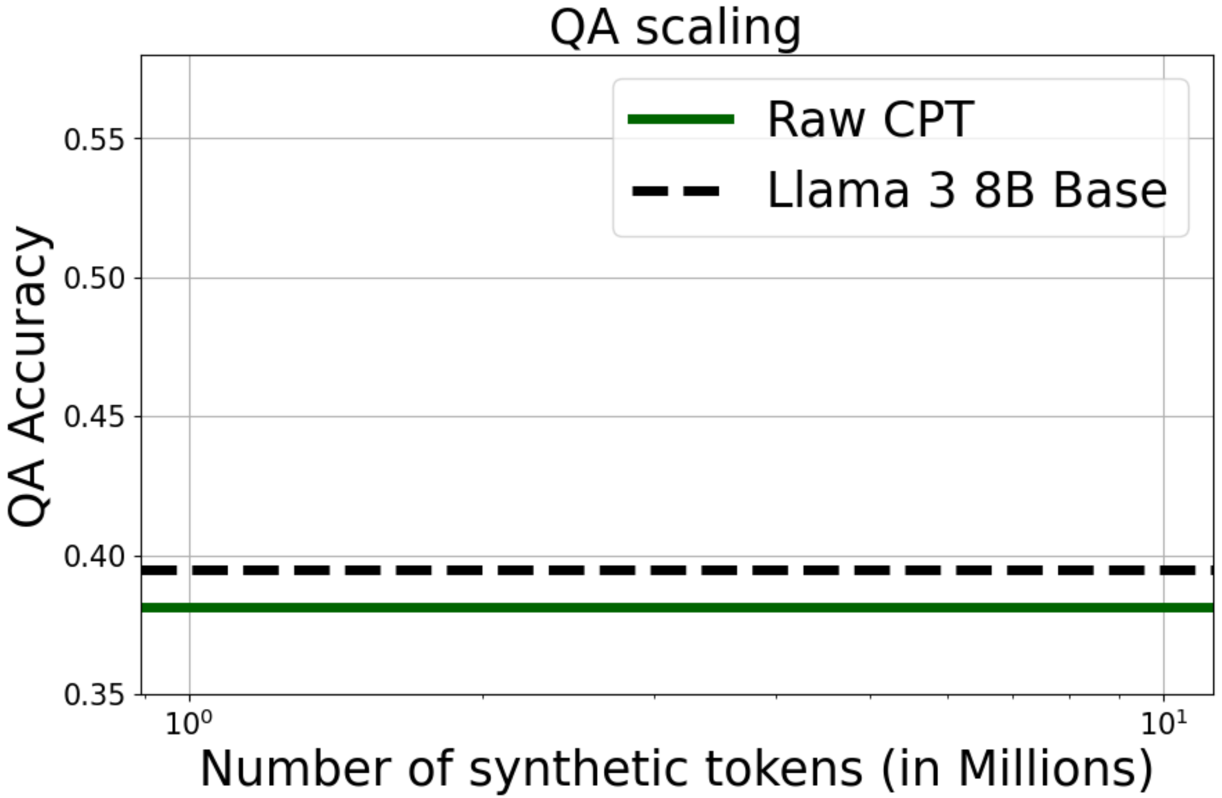
A good benchmark for this should have

- Obscure books / knowledge
- Knowledge appears once or twice
- High-quality QA (and other) evals

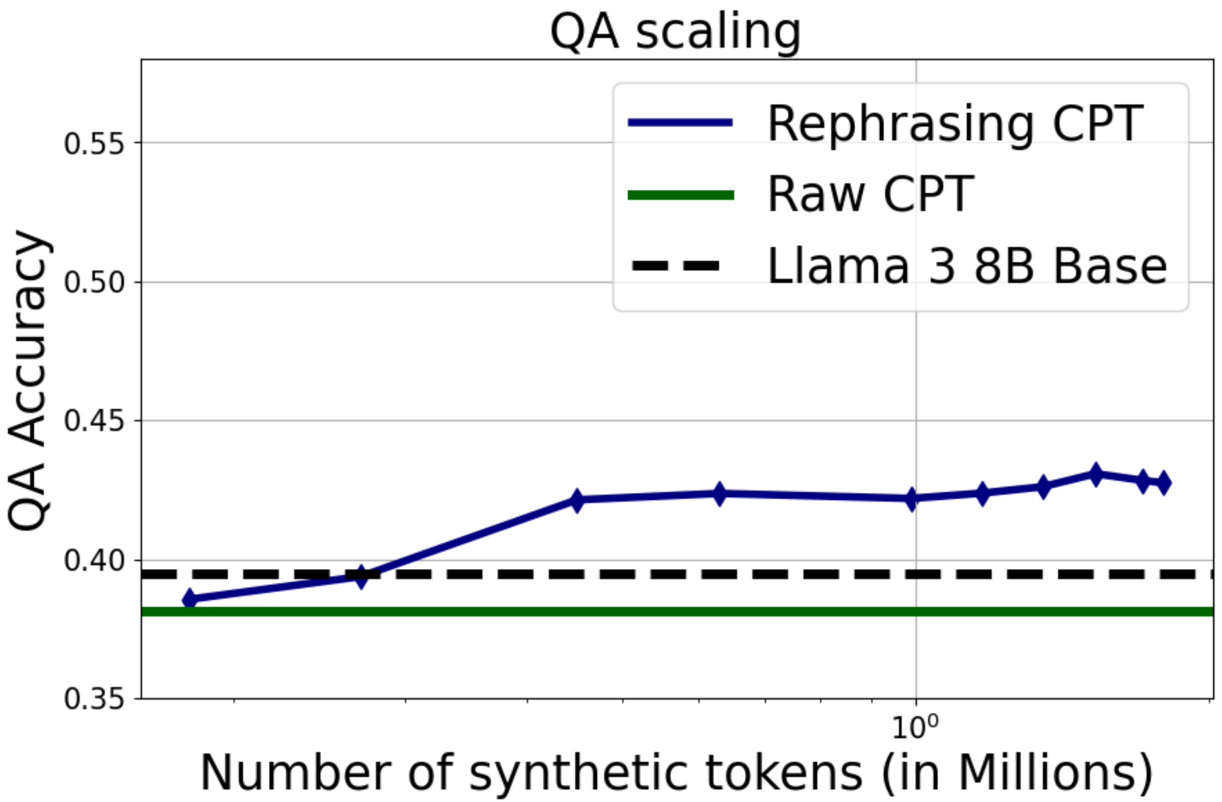
A good dataset: **QuALITY** [Pang+ '21]

- Niche fiction / magazine articles
- 1.3M tokens (too small for CPT)
- High-quality QA / summary evals
- Even GPT4 is ~51% Acc, Llama ~39%

Attempt 1 – Just do continued pretraining

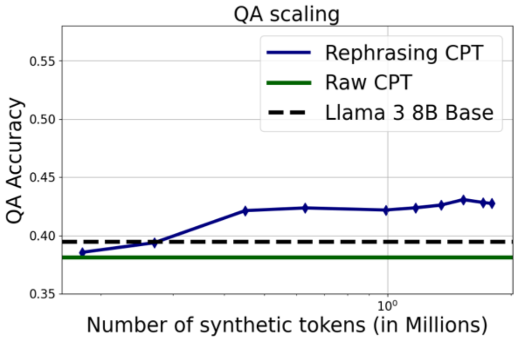


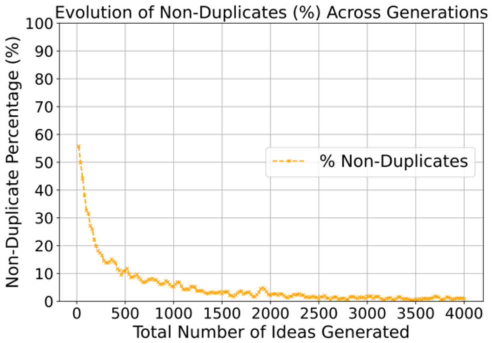
Attempt 2 – Just paraphrase the data



Idea: external sources of diversity

Core problem: LLMs are not terribly diverse – asking for random samples is a bad idea





[Si+ 2024]

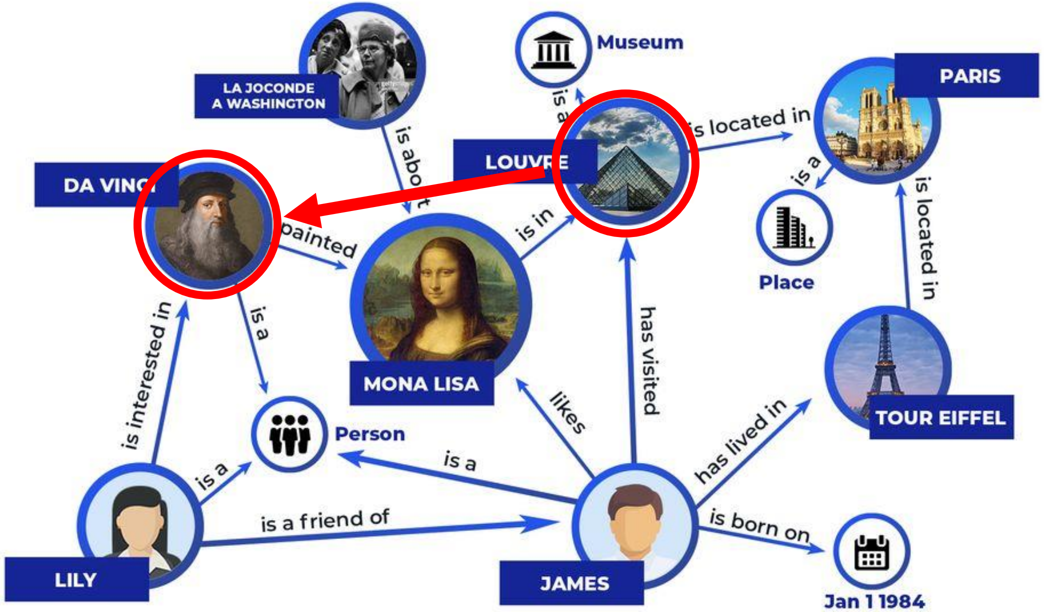
Increasingly common method: External source of randomness / information

- Alpaca – random human seed data / conditioning on past data
- Skill-Mix-Instruct – conditioning on a combinatorial set (pairs of skills)
- (Our approach) – randomize over content choices using a knowledge graph

What we get: Entity-centric augmentation (EntiGraph)

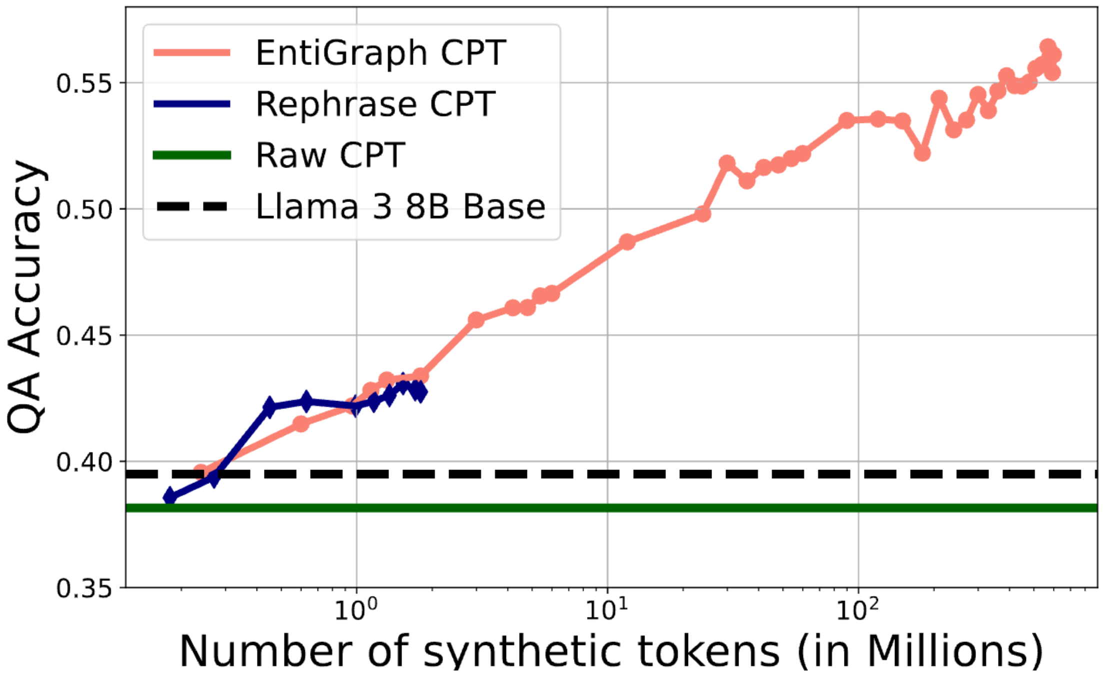
How do we get diversity? Use a knowledge graph to force diversity in content

1. Prompt LM_{aug} for entities in a knowledge graph.
2. Sample k-subgraphs of the knowledge graph
3. LM_{aug} synthesizes descriptions of the entities in the subgraph



New implicit fact as data (The Louvre contains many works by DaVinci..)

(Closed-book) QA performance with EntiGraph



Predictable, scaling gains for QA performance *without* the text

Exceeding your teacher (GPT4)

Split	Continually Pretrained Llama 3 8B			Base Models and API-Based LLMs		
	EntiGraph	Rephrase	Raw	Llama 3 8B	GPT-4	GPT-3.5
All	56.42	43.08	38.15	39.49	51.30	44.81
Hard	48.15	36.98	33.66	35.08	42.13	38.07
Easy	64.75	49.23	42.65	43.93	60.55	51.60

Table 2: QuALITY accuracy over all test queries $\mathcal{Q}_{\text{test}}$ (All), and the Easy and Hard splits. The left set of columns are Llama 3 8B continually pretrained on various data sources. The right set of columns are the base model and API-based LLMs not finetuned on QuALITY-related data. EntiGraph CPT outperforms the Rephrase and Raw CPT baselines.

We improve significantly (17%) via EntiGraph, exceeding even the teacher (GPT4) by ~5%

But does it generalize like an LLM?

Fine-tuning for QA alone isn't that surprising

We instruction-tuned the EntiGraph model (ultrachat SFT) and ..

- It can summarize
- Do writing tasks
- Relate multiple articles

➤	**Explicit article reference:** Summarize the article "Defining Decay Down" by "David Plotz".
	The article "Defining Decay Down" by David Plotz discusses the evolution of dentistry in America. It highlights how dentists have adapted to changing times by shifting their focus from essential dental work to elective cosmetic procedures. [...]
➤	**Implicit article reference:** How has the practice of dentistry in the United States changed?
	The dentistry practice in the United States has undergone significant changes over the years. Here are some of the key changes: 1. Increased focus on cosmetic dentistry: In recent years, [...] 2. Improved dental technology: Advances in dental technology [...] 3. Increased emphasis on preventive care: Dentists have become more focused [...] 4. Increased competition: The dental industry has become more competitive, [...] 5. Improved patient experience: [...]
➤	**Cross article instruction:** Compare David Plotz's commentary style between how he analyzed American dentistry and how he discussed the movie Fight Club?
	David Plotz's commentary style is analytical and critical in both his analysis of American dentistry and his discussion of the movie Fight Club. In both cases, he uses a sharp, incisive tone to critique the subject matter, highlighting the absurdity and excesses of American dentistry and the nihilistic philosophy of Fight Club. However, there are some differences in his approach. [...]

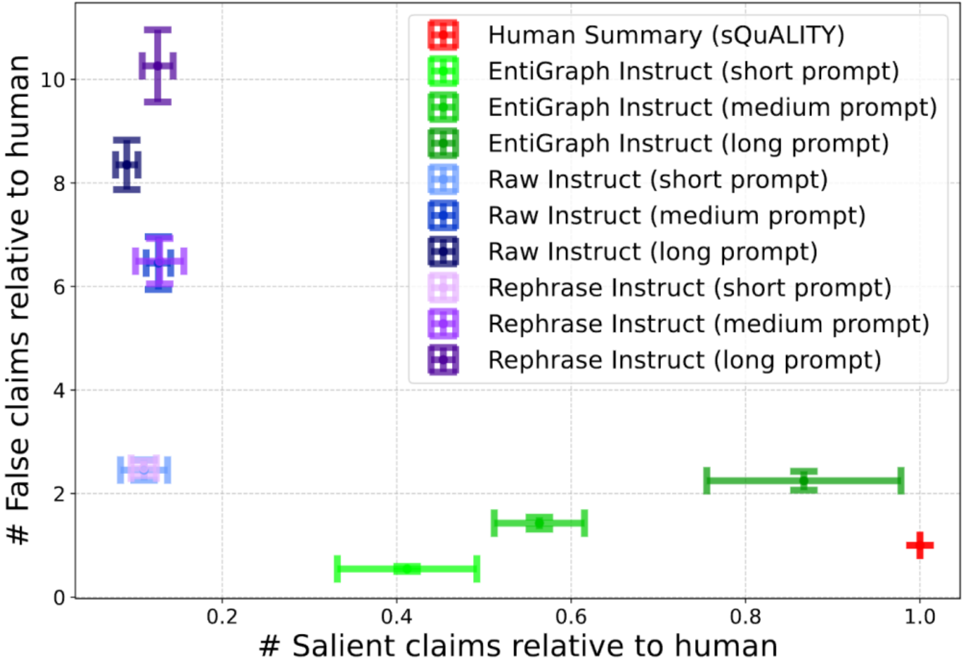
Qualitative summary evaluations

EntiGraph Instruct with short prompt: Cosmic Yo-Yo is a science fiction story by Ross Rocklynne about Bob Parker and Queazy, two interplanetary haulers who are hired to deliver an asteroid to Earth for a wealthy client's backyard wedding. However, they face competition from their rivals, the Saylor brothers, who also want to deliver the asteroid. The story follows their race to deliver the asteroid before the deadline, using a yo-yo-like device to maneuver the asteroid and outsmart their rivals. The story is filled with action, adventure, and a touch of romance.

Raw Instruct with short prompt: Cosmic Yo-Yo by Rocklynne, Ross is a science fiction novel that follows the story of a young woman named Susan who discovers a mysterious device that can transport her through time and space. She uses the device to travel to different planets and meet different alien species, but soon realizes that her actions have consequences and she must find a way to return home. The novel explores themes of identity, responsibility, and the consequences of our actions.

Rephrase Instruct with short prompt: Cosmic Yo-Yo is a science fiction novel written by Ross Rocklynne in 1954. The story follows the adventures of a group of astronauts who are sent on a mission to explore a new planet. However, they soon discover that the planet is not as hospitable as they thought and must fight for their survival against the hostile environment and the indigenous creatures. The novel explores themes of exploration, survival, and the unknown.

Quantitative summarization evaluation



Entigraph shows few false claims (0.5-2x) compared to baselines with more salient (true) claims

Does parametric knowledge complement retrieval?

Ok, but why not use retrieval augmentation?

Split	EntiGraph CPT + RAG		Llama 3 8B Base + RAG		GPT-4 + Oracle RAG		GPT-3.5 + Oracle RAG	
	Accuracy	Recall@8	Accuracy	Recall@8	Accuracy	Recall@8	Accuracy	Recall@8
All	62.73	99.63	60.35	99.63	86.09	100.0	72.60	100.0
Hard	53.87	99.65	50.24	99.65	79.59	100.0	63.13	100.0
Easy	71.68	99.61	70.55	99.61	92.65	100.0	82.14	100.0

RAG baselines with a *very* strong retriever (99+% recall)

Entigraph augmentation helps across the board (2-3%) on top of RAG

Our closed book perf (56%) is almost the LLaMA RAG perf, and 80% of the gains (40-60)

A theory perspective to entity-centric augmentation

Why do we get gains from ‘diverse rewritings’ of the original data?

Let’s build a toy mathematical model

- We have a set of entities V in a single document D_{source}
- Claims that appear *directly* (‘x is y’) are represented as $D_{\text{source}} \in \{(x, y) \in V^2\}$

As a generative model, we assume an Erdos-Renyi graph where edge appear with probability p and define the rate $\lambda = p|V|$

The toy model – random graph process

We now model EntiGraph's augmentation process - 'filling in the graph'

1. **Entity pair selection:** Sample $(x_t, y_t) \in \{(x, y) \in \mathcal{V}^2 : x \neq y\}$ uniformly at random.
2. **Relation analysis:** Generate the "relation between (x_t, y_t)" by performing a breadth-first search (BFS) on the directed graph represented by the adjacency matrix $\boldsymbol{M}_0$ starting at x_t:
   - If there exists a path $(x_t, z_t^1, z_t^2, \ldots, z_t^{k_t}, y_t)$ connecting x_t to y_t, define
   $$\mathcal{D}_t = \{(x_t, z_t^1), (x_t, z_t^2), \ldots, (x_t, z_t^{k_t}), (x_t, y_t)\} \cup \mathcal{D}_{t-1}.$$
   The model trained on this round of synthetic data would be
   $$\boldsymbol{M}_t = \boldsymbol{M}_{t-1} + \sum_{(x,y) \in \mathcal{D}_t \setminus \mathcal{D}_{t-1}} \boldsymbol{I}_{xy},$$
   where $\boldsymbol{I}_{xy} \in \{0, 1\}^{V \times V}$ is a binary matrix with $\boldsymbol{I}_{xy}(x, y) = 1$ and 0 otherwise.
   - If no such path exists, do nothing.

Learning as memorization – we fill all vertices on the 'path' to the target

Asymptotic accuracy of EntiGraph follows the ER limits

With high probability,

Definition 1. *Let $C_\lambda = (1-\rho(\lambda))^2$, where $\rho(\lambda)$ denotes the extinction probability for a Poisson(λ) branching process (i.e., ρ is the smallest solution in $[0,1]$ to the fixed-point equation $\rho = \exp(\lambda(\rho - 1))$). For any fixed $\varepsilon > 0$, we further define*

$$C_{\mathrm{LB}} = 1 - \frac{1}{V(V-1)}, \quad C_{\mathrm{UB}} = 1 - \frac{(1+\varepsilon)\log V}{V(V-1)\log\lambda}.$$

Theorem 1. *For any time $t \geq 1$ and any $\varepsilon > 0$, the link density satisfies*

$$\left(p + C_\lambda \left(1 - C_{\mathrm{LB}}^t\right)\right)(1-\varepsilon) \leq \mathsf{Acc}(\boldsymbol{M}_t) \leq \left(p + C_\lambda \left(1 - C_{\mathrm{UB}}^t\right)\right)(1+\varepsilon),$$

with probability $\to 1$ when $V \to \infty$.

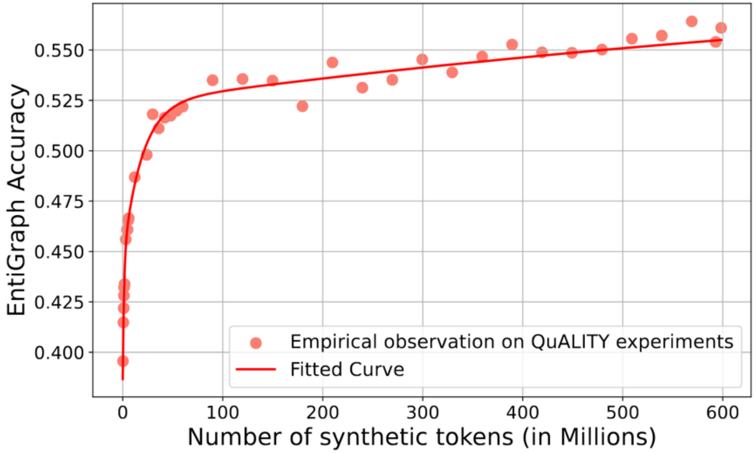
(The implied asymptotics here are $(p + (1-\rho)^2)$- c.f. Erdos Renyi phase transition)

Implied scaling process – a mixture of exponentials

A less precise, but intuition building result – scaling should be mix-of-exps

$$\mathsf{Acc}(\boldsymbol{M}_t) \sim p + C_\lambda \left(1 - \sum_{\ell=0}^{\infty} \frac{\lambda - 1}{\lambda^{\ell+1}} \sum_{k=1}^{\infty} p_\ell(k) \left(1 - \frac{k}{V(V-1)}\right)^t\right),$$

A mixture of 3 exponentials matches observed scaling well



Takeaways: synthetic continued pretraining

Tail knowledge and data efficiency will become increasingly important

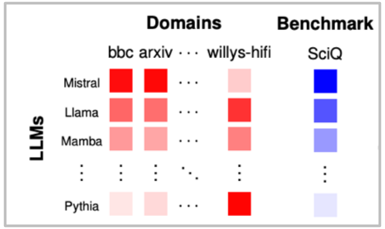
Can LM pretraining-style learning be made data-efficient?

With synthetic data augmentation (and tricks), yes!

- Effective CPT – not at the 50B token level, but at 1M tokens.
- 80% of the gains from retrieval can be obtained via CPT
- Exciting testbed for data-efficient language modeling

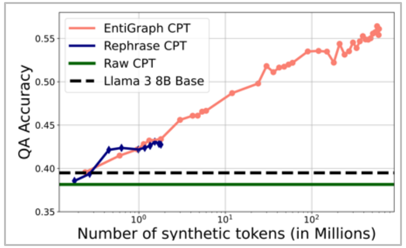
Takeaway – engineering data interventions for generalization

Data selection via perplexity correlations



- Algorithmic control of pretraining data is possible
- Public models contain valuable perplexity-correlation info
- Preregistration-based scaling experiments

Synthetic continued pretraining



- Continued pretraining at the 1M token level is possible
- Entity-based methods of making diverse, synthetic data
- Predictable, multi-task gains via CPT.