

CS11-711 Advanced NLP

Advanced Tokenization + Pre-training

Daphne Ippolito and Fernando Diaz



<https://cmu-anlp.org>

<https://github.com/cmu-l3/anlp-spring2026-code>

Announcements

- Please submit your AWS ID on Canvas by the end of the week.
- Daphne's office hours are cancelled this week. If you need to chat, send me an email and we can find a time.

Outline

- Advanced tokenization (continuation of last lecture)
- NLP training paradigms through the ages
- The ideal LLM pre-training setup
- Assembling a pre-training dataset
- Cleaning, filtering, and quality control

Advanced tokenization methods

In week 1, we covered byte-pair encoding (BPE).

Suppose the word “pingaprong” never occurred in the training data used to build the vocab. How would this get encoded with BPE?

- A. It would get mapped to a special `_UNKNOWN` token.
- B. It will get tokenized as one token per UTF-8 byte.
- C. It will get greedily broken into the longest pieces which *are* tokens.

In week 1, we covered byte-pair encoding (BPE).

Suppose the word “pingaprong” never occurred in the training data used to build the vocab. How would this get encoded with BPE?

- A. It would get mapped to a special `_UNKNOWN` token.
- B. It will get tokenized as one token per UTF-8 byte.
- C. ☒ It will get greedily broken into the longest pieces which *are* tokens.

pingaprong

How would you want your tokenizer to tokenize the white space in this Python code?

```
from heapq import merge
```

```
def merge_sort(m):  
    if len(m) <= 1:  
        return m
```

```
    middle = len(m) // 2  
    left = m[:middle]  
    right = m[middle:]
```

```
    left = merge_sort(left)  
    right = merge_sort(right)  
    return list(merge(left, right))
```

Tokenizing whitespace

T5
(n=85)

```
fromheapqimportmergedefmerge_sort(m):iflen(m)<unk>=1  
:returnmmiddle=len(m)//2left=m[:middle]right=m[mi  
ddle:]left=merge_sort(left)right=merge_sort(right)  
returnlist(merge(left,right))
```

Tokenizing whitespace

GPT-3

(n=107
)

```
from heapq import merge
```

```
def merge_sort(m):  
    if len(m) <= 1:  
        return m
```

```
    middle = len(m) // 2  
    left = m[:middle]  
    right = m[middle:]
```

```
    left = merge_sort(left)  
    right = merge_sort(right)  
    return list(merge(left, right))
```

Tokenizing whitespace

GPT-5

(n=70)

```
from heapq import merge
```

```
def merge_sort(m):  
    if len(m) <= 1:  
        return m
```

```
    middle = len(m) // 2  
    left = m[:middle]  
    right = m[middle:]
```

```
    left = merge_sort(left)  
    right = merge_sort(right)  
    return list(merge(left, right))
```

Handling multilinguality

- The encoder-decoder solution: learn a separate vocabulary for each language, use separate vocab embedding matrices in the encoder and decoder.
- Bad solution: learn a single vocabulary from whatever data you have on hand.
- Better solution: deliberate language balancing.

Evaluating tokenizers

- Fertility
 - the average number of tokens it takes to represent text x divided by the total number of words or characters in text x .
 - similar to compression ratio

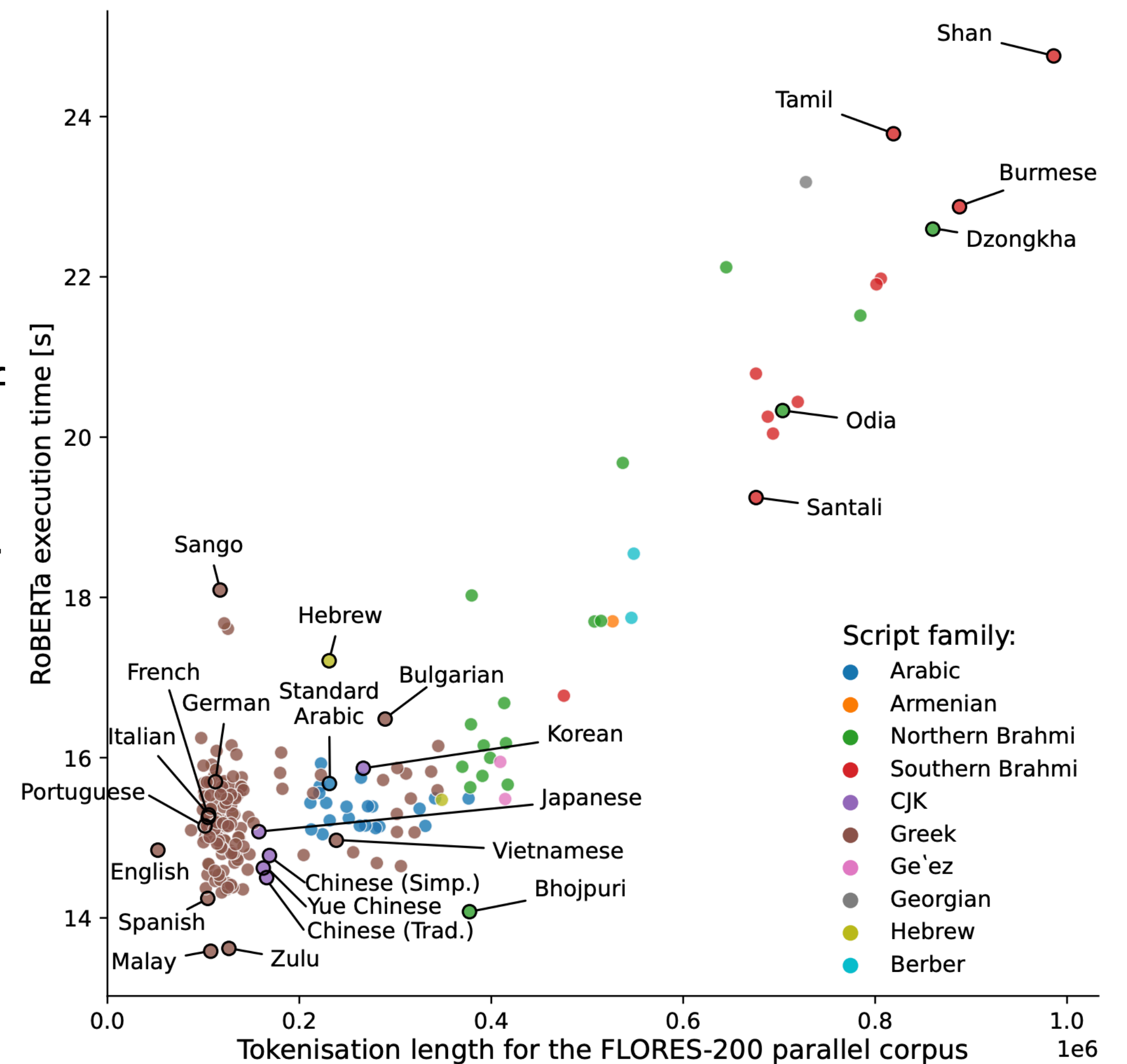
Evaluating tokenizers

- Parity
 - how fairly a tokenizer treats equivalent sentences in different languages
 - parity between languages A and B is 1 if on average equivalent sentences in the two languages map to the same number of tokens

	Parity with respect to English		
	GPT-2 RoBERTa	ChatGPT GPT-4	FlanT5
Bulgarian	5.51	2.64	—
Burmese	16.89	11.70	—
Chinese (Simplified)	3.21	1.91	—
Dzongkha	16.36	12.33	—
English	1.00	1.00	1.00
French	2.00	1.60	1.60
German	2.14	1.58	1.37
Italian	2.01	1.64	2.18
Japanese	3.00	2.30	—
Jingpho	2.65	2.35	3.41
Maori	2.45	2.35	3.28
Norwegian Bokmål	1.86	1.56	2.24
Odia	13.38	12.48	—
Pangasinan	1.66	1.57	2.18
Portuguese	1.94	1.48	2.21
Romanian	2.48	1.88	1.50
Santali	12.86	12.80	—
Shan	18.76	15.05	—
Spanish	1.99	1.55	2.23
Standard Arabic	4.40	3.04	—
Tumbuka	2.78	2.57	3.29
Vietnamese	4.54	2.45	—

Evaluating tokenizers

- Parity
 - how fairly a tokenizer treats equivalent sentences in different languages
 - parity between languages A and B is 1 if on average equivalent sentences in the two languages map to the same number of tokens



Why not just use byte- or character-level
tokenization?

Why not just use byte- or character-level tokenization?

- Cons:
 - Extremely long sequence lengths
 - Individual characters/bytes don't carry much semantic meaning

However, some attempts have been made.

- Newer model architectures (e.g. state space models such as Mamba) eliminate the $O(n^2)$ parameter growth with sequence length n . This makes them more amenable to byte-level tokenization.
 - MambaByte: Token-free Selective State Space Model
- Other approaches try to modify existing Transformer architectures to make them more amenable to byte-level tokens.
 - ByT5: Towards a Token-Free Future with Pre-trained Byte-to-Byte Models

Why do we insist on token boundaries at
whitespace boundaries?

Why do we insist on token boundaries at whitespace boundaries?

- **Pretokenization:** the method used to break strings of text into initial chunks (which get fed into the BPE or other algorithm)
 - For English, this is typically whitespace boundaries, punctuation, numbers etc. For Chinese, there are no whitespace boundaries to split on.
 -

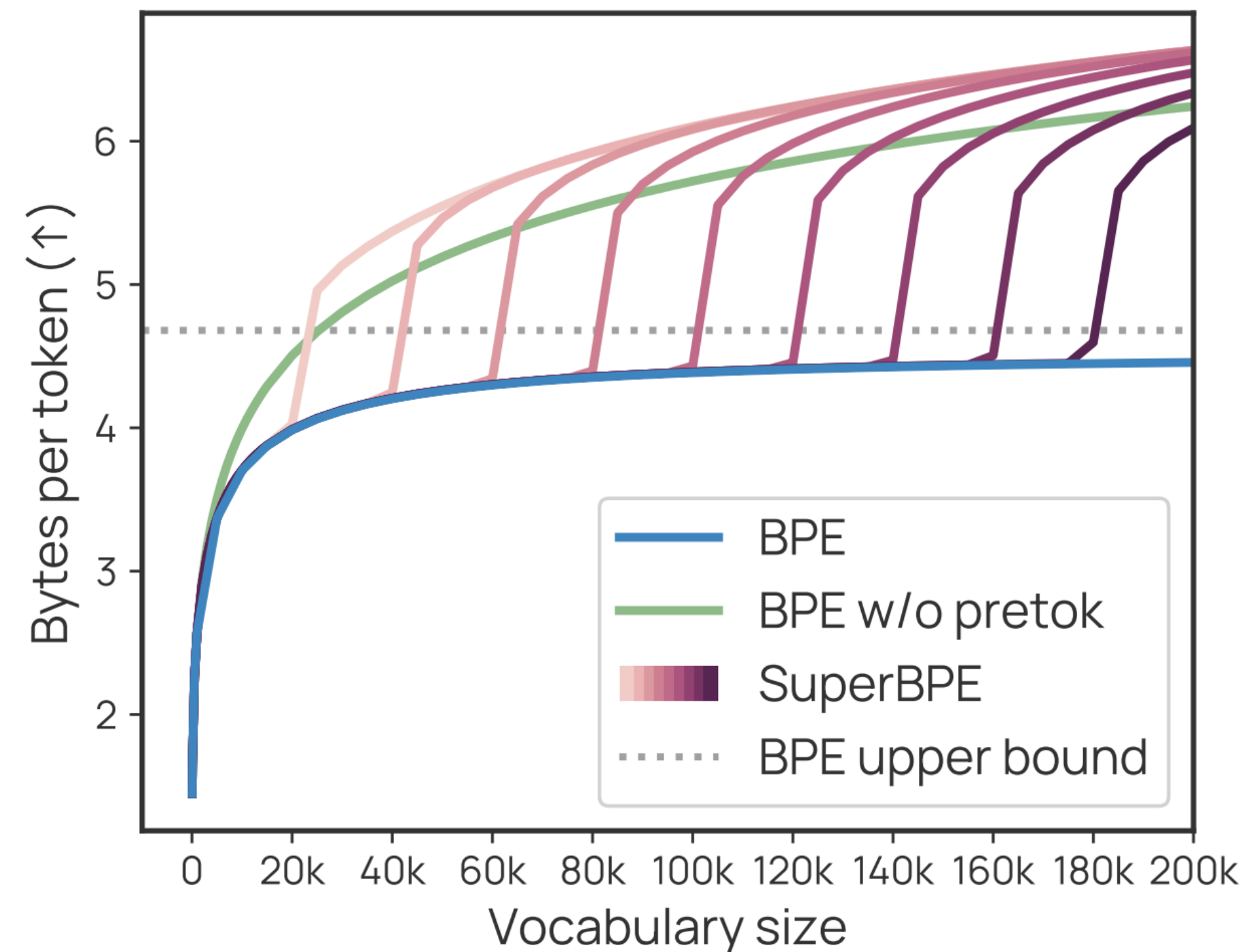
Why do we insist on token boundaries at whitespace boundaries?

- **Pretokenization:** the method used to break strings of text into initial chunks (which get fed into the BPE or other algorithm)
 - For English, this is typically whitespace boundaries, punctuation, numbers etc. For Chinese, there are no whitespace boundaries to split on.
- Alternative:
 - don't include whitespace in the pretokenization stage.

Tokenization that transcends whitespace

BPE: By the way, I am a fan of the Milky Way.

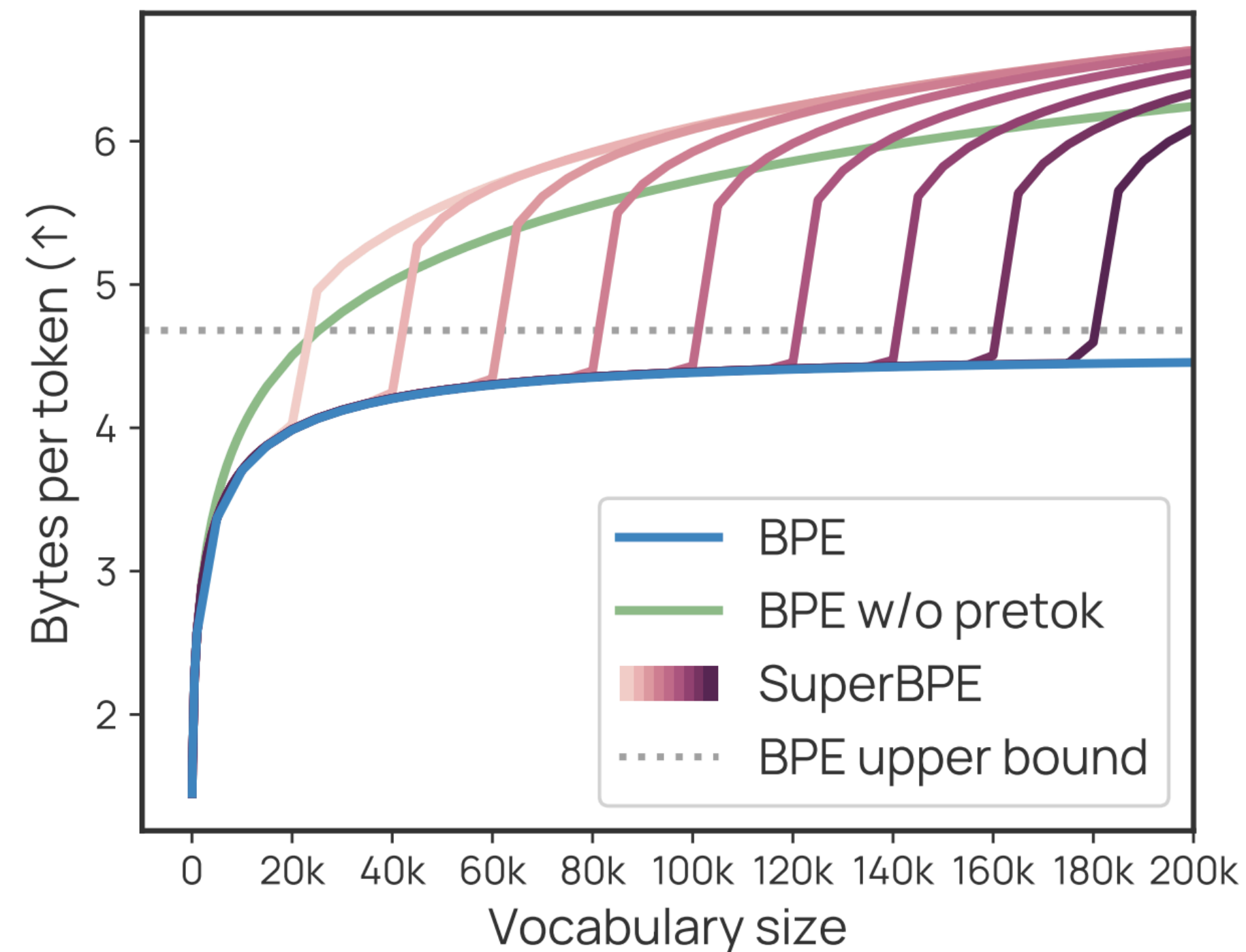
SuperBPE: By the way, I am a fan of the Milky Way.



Tokenization that transcends whitespace

BPE: By the way, I am a fan of the Milky Way.

SuperBPE: By the way, I am a fan of the Milky Way.



SuperBPE tokenization

1. Pretokenize the text using standard rules.
2. Start running the regular BPE algorithm (that we covered in week 1)
3. Once the working vocabulary reaches size t , start ignoring the pretokenization—tokens can now be merged across whitespace boundaries.

SuperBPE tokenization

1. Pretokenize the text using standard rules.
2. Start running the regular BPE algorithm (that we covered in week 1)
3. Once the working vocabulary reaches size t , start ignoring the pretokenization—tokens can now be merged across whitespace boundaries.

Intuition: tokenizer first learn base units of semantic meaning, then, after the transition point, it learns to combine these units into common sequences for a much more efficient vocabulary.

SuperBPE tokenization

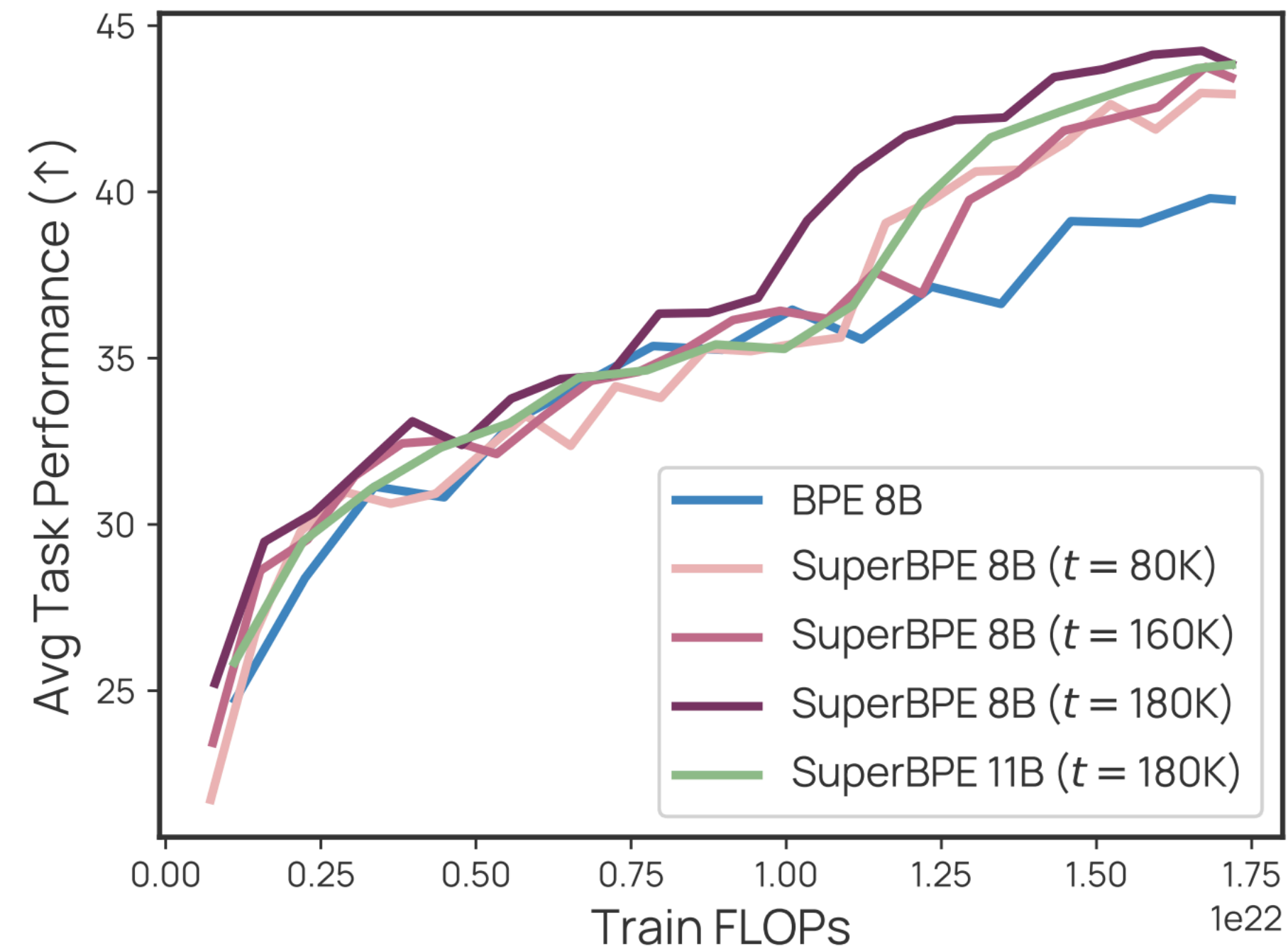


Figure 3: **Average task performance on 30 downstream tasks, evaluated at every 5000 steps in model pretraining.** We see that SuperBPE models consistently outperform the baseline that uses a BPE tokenizer. All compared models share the same vocabulary size and train budget; t denotes the transition point in SuperBPE’s pretokenization curriculum.

Outline

- Advanced tokenization (continuation of last lecture)
- NLP training paradigms through the ages
- Choosing an LLM pre-training setup
- Sources of pre-training data
- Filtering, curation, and quality control

NLP training paradigms through the ages

The year is 2004. Google is about to release Gmail. You've been tasked with building a spam classifier for a brand new email client.

Your classifier should input all available info about an email (contents, subject, sender, etc.) and produce a label of SPAM or NOT_SPAM.

Take 5 minutes to brainstorm with your seatmate(s).

Training NLP classifiers before deep learning

- Collect a set of labeled examples and break into train/valid/test splits
- Extract features from each example
 - Presences/absence of keywords, IP address of sender, timestamp, etc.
- Train a classifier to make predictions based on features
 - Logistic regression, SVM, decision tree, etc.

Training NLP classifiers in the early deep learning days

- Training data has not changed much
- Main differences:
 - Classifier is a neural network
 - Feature representations may be embeddings from a neural model

Alternative Language-Model Based Approach

- Start with a set of labeled textual examples
- Train one language model per text label
 - e.g. one on spam email, one on good emails
- For a test example, predict the class corresponding to LM with highest likelihood on that example.

Data assumptions

The *only* data we have/need is the set of labeled examples.

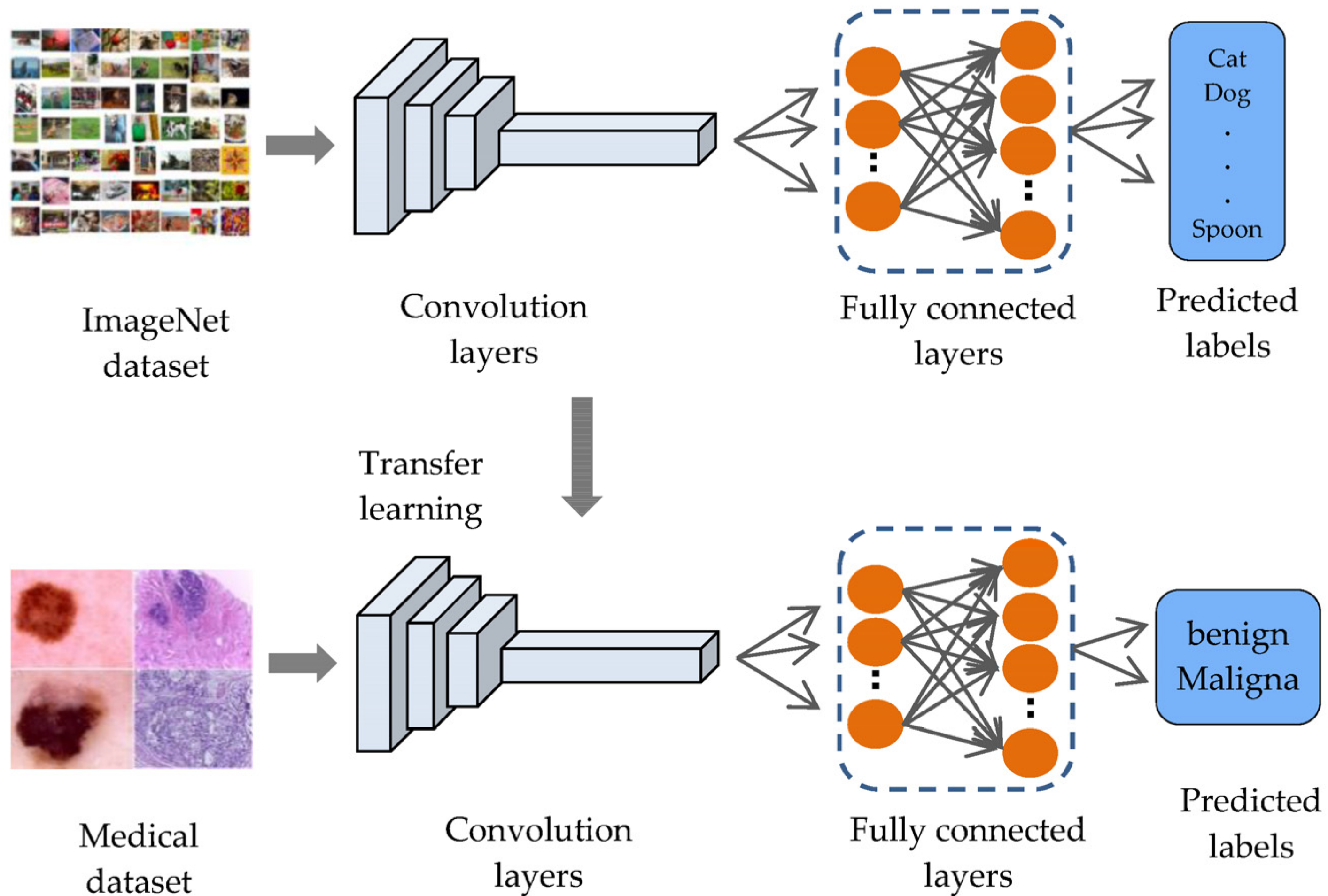
Data assumptions

The *only* data we have/need is the set of labeled examples.

What if Google has one million unlabeled emails and only 20k labeled ones?

What if we need a spam classifier for email, and another spam classifier for YouTube comments?

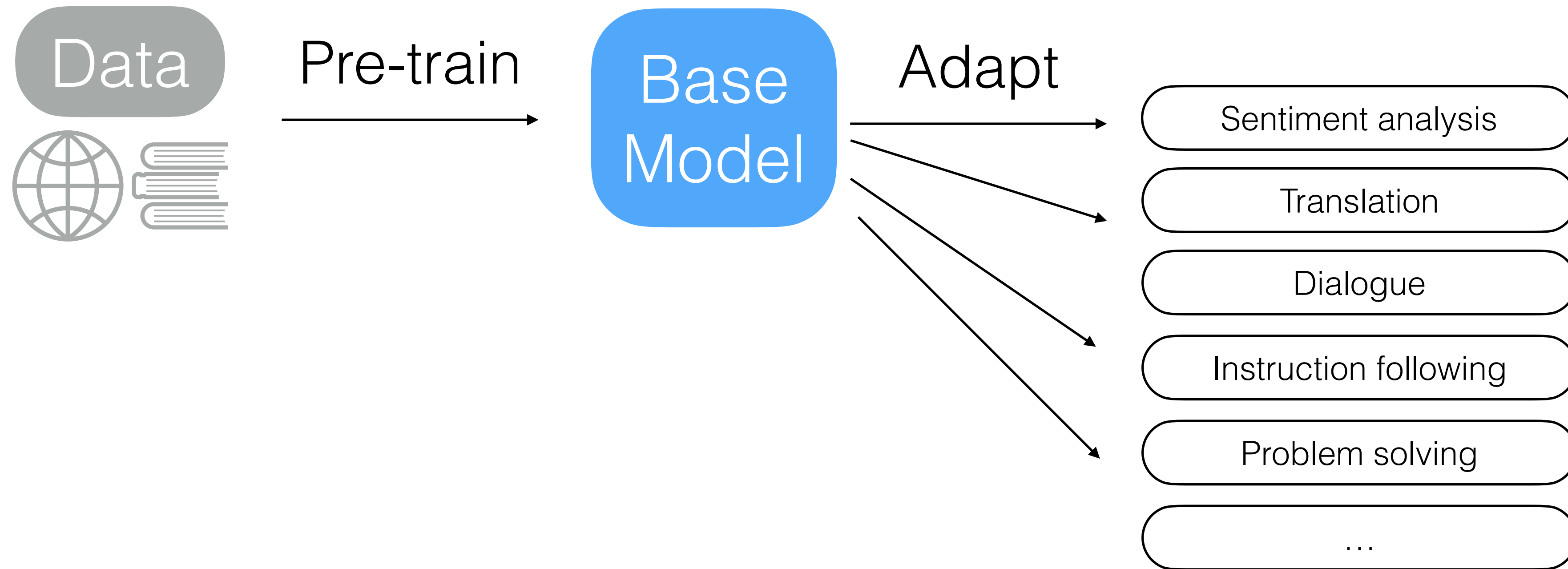
Transfer learning



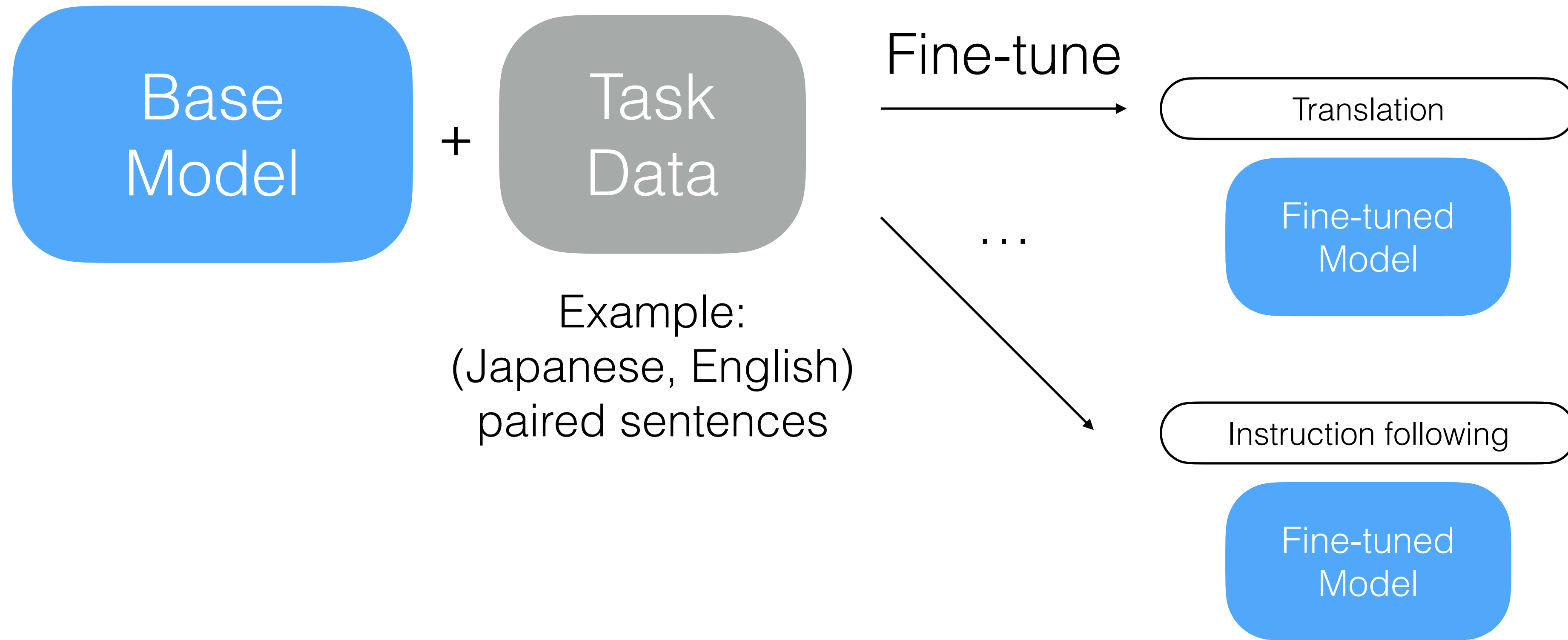
Why pre-train?

- Transfer learning: take “knowledge” from one task and apply it to another task
- **Less task data:** use less data to reach a given level of performance
- **Better task performance:** reach higher performance than training from scratch
- **One model, multiple tasks:** convenient, amortizes cost, a starting point for many uses, ...

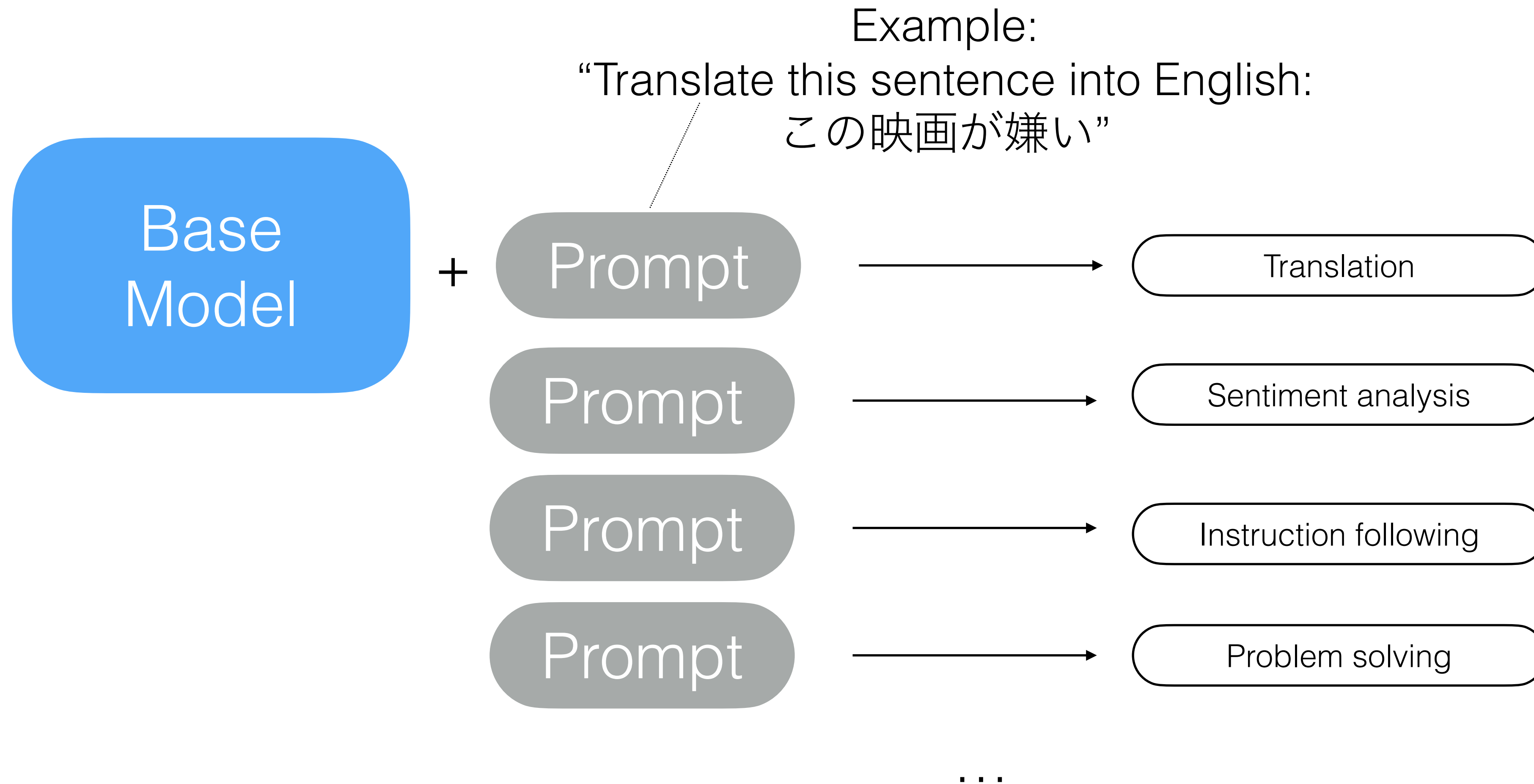
The Pretrain and Adapt Paradigm



Adaption via fine-tuning



Adaption via prompting



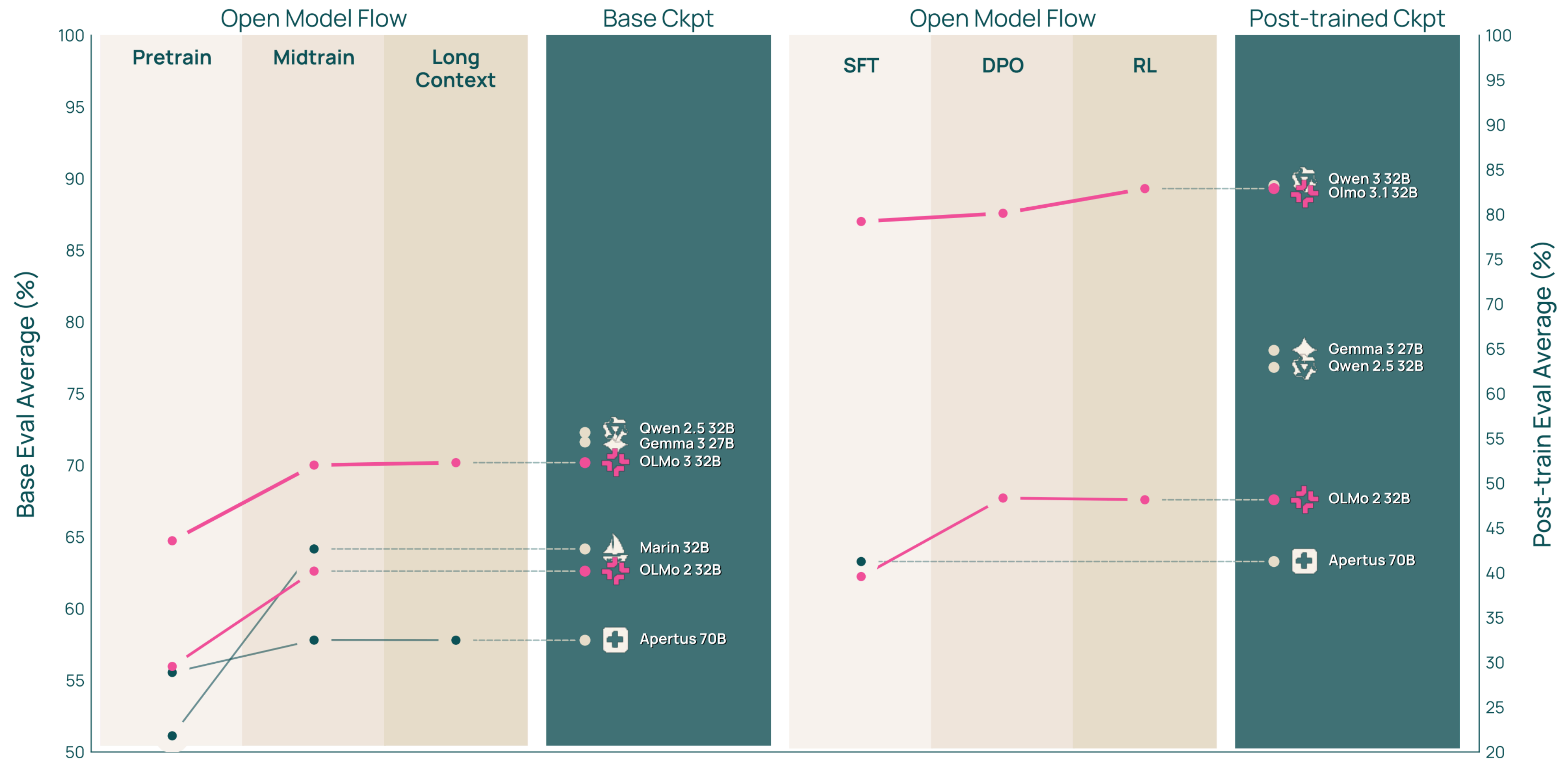
Adaption via post-training



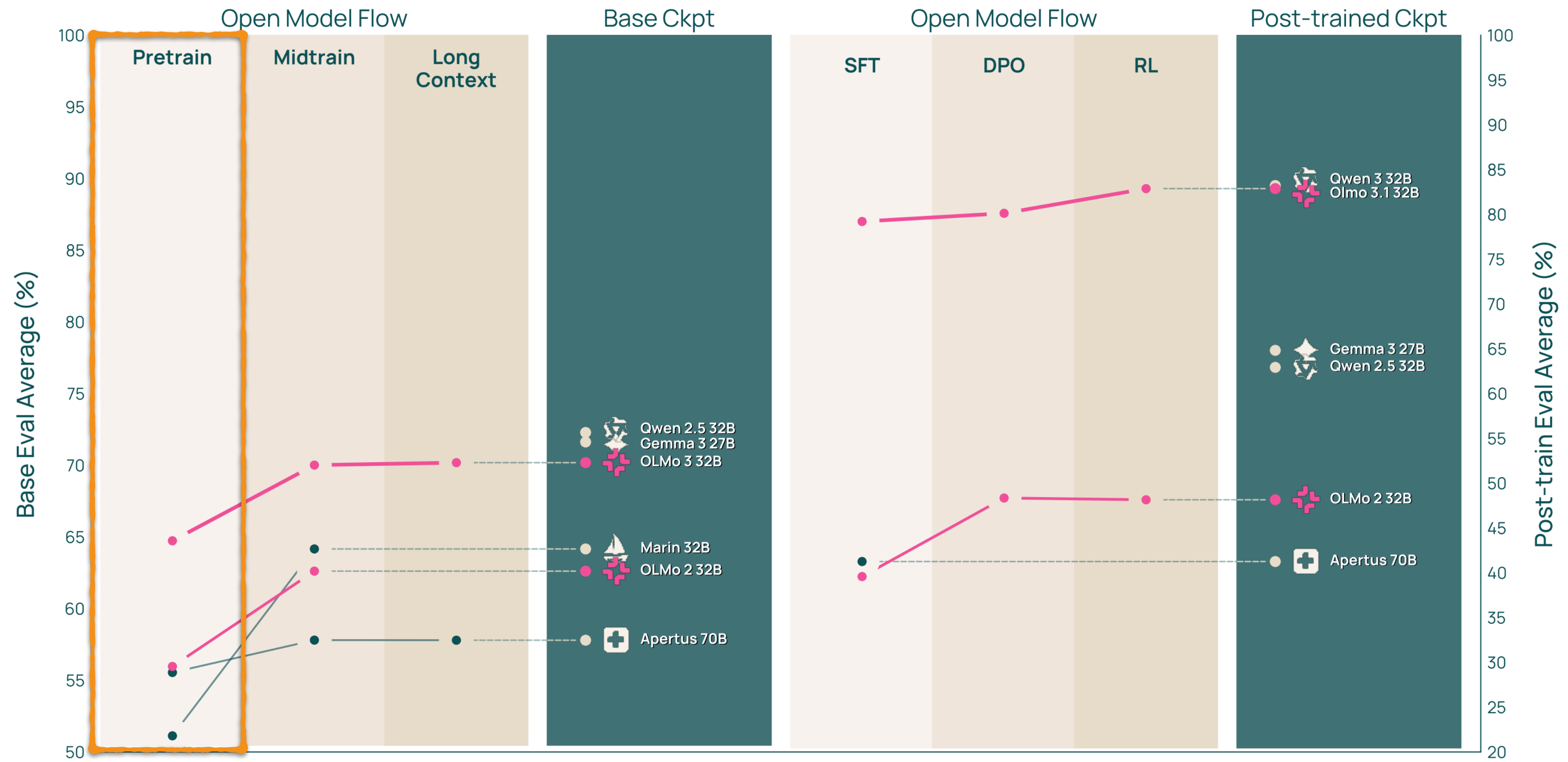
Example:
Alpaca, FLAN, etc.



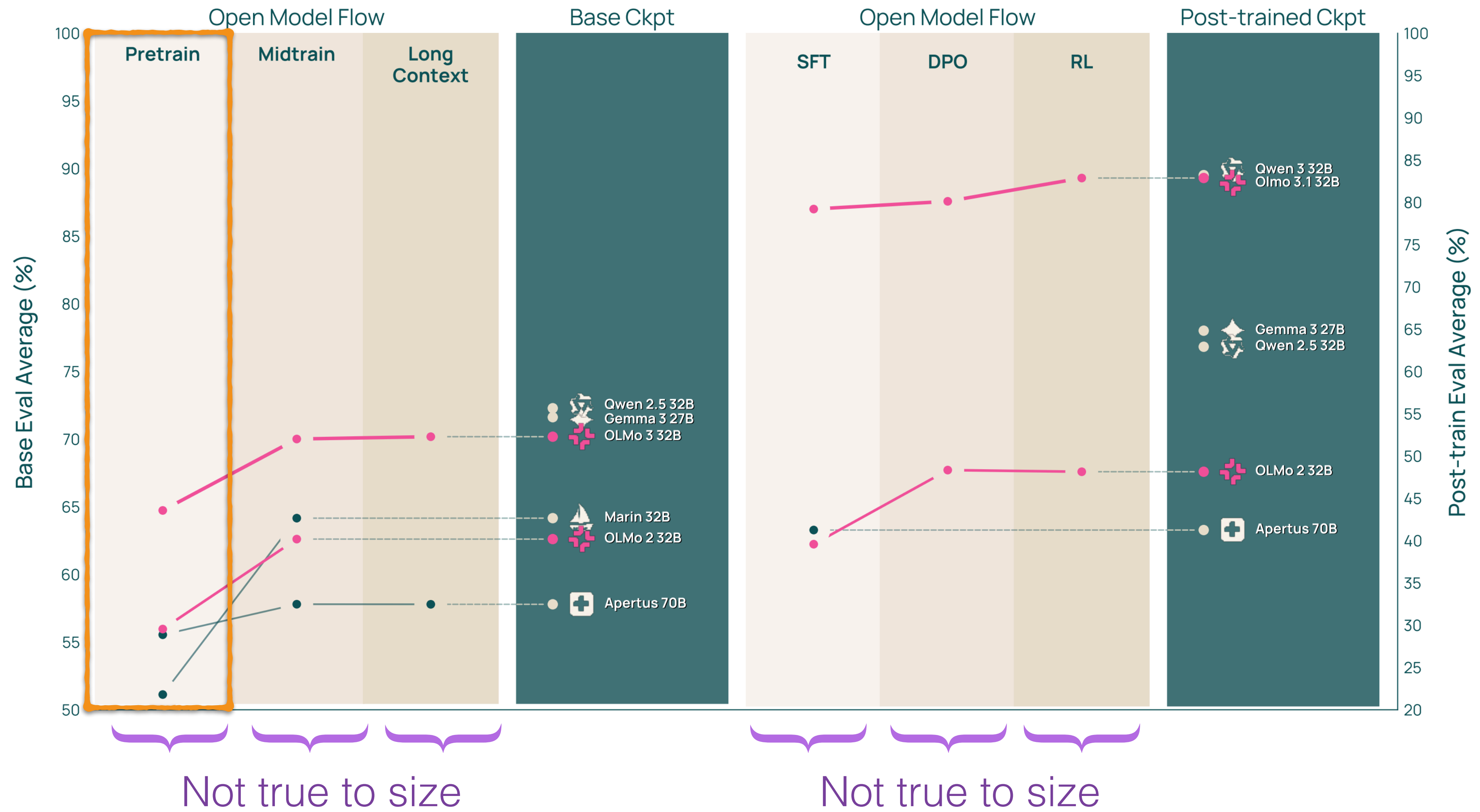
Example:
HH-RLHF



OLMO 3 [AI2 Dec 2025]



OLMO 3 [AI2 Dec 2025]



OLMO 3 [AI2 Dec 2025]

Choosing an LLM pre-training setup

What traits would you hope for in the perfect language model pre-training data?

Take 3 minutes to brainstorm with your seatmate(s).

Ideal Traits of Pre-Training Data

- i. Easy/cheap to collect
- ii. Plentiful
- iii. Achieves models that can be readily adapted to many diverse tasks

Ideal Traits of Pre-Training Data

- i. Easy/cheap to collect
- ii. Plentiful
- iii. Achieves models that can be readily adapted to many diverse tasks

Suppose you are Google.

Which properties do each of these meet?

- A dataset of emails

Ideal Traits of Pre-Training Data

- i. Easy/cheap to collect
- ii. Plentiful
- iii. Achieves models that can be readily adapted to many diverse tasks

Suppose you are Google.

Which properties do each of these meet?

- A dataset of emails hand-annotated with the grammatical structure of each sentence

Ideal Traits of Pre-Training Data

- i. Easy/cheap to collect
- ii. Plentiful
- iii. Achieves models that can be readily adapted to many diverse tasks

Suppose you are Google.

Which properties do each of these meet?

- A crawl of most of the internet

Ideal Traits of Pre-Training Data

- i. Easy/cheap to collect
- ii. Plentiful
- iii. Achieves models that can be readily adapted to many diverse tasks

Data that is plentiful and cheap is unlikely to have any human-validated labels or annotations.

Suppose you are Google.

Which properties do each of these meet?

- A dataset of emails
- A dataset of emails hand-annotated with the grammatical structure of each sentence.
- A crawl of most of the internet

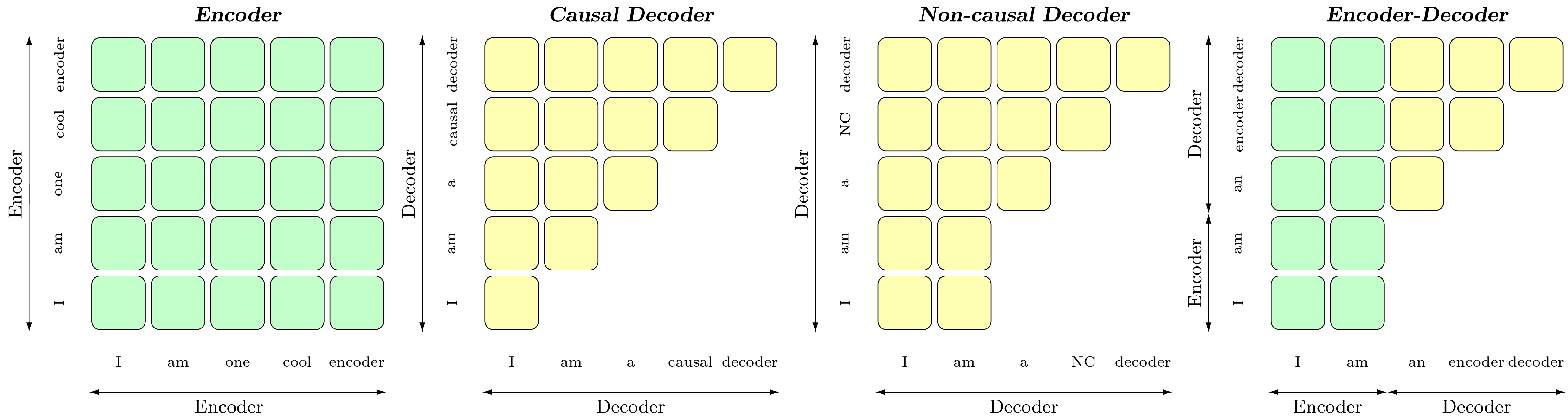
Ideal Traits of Pre-Training Data

- i. Easy/cheap to collect
- ii. Plentiful
- iii. Achieve models that can be readily adapted to many diverse tasks
- iv. Has desirable composition and biases (e.g. good language balance, no sexually explicit content or harmful stereotypes, dense in high-quality content)

You've collected a corpus of high quality text and are ready to pre-train a model.

What other choices do you need to make?

Architecture



Also: model size, shape, etc.

Learning Objective

Full Language Modeling

May

targets

the force be with you

Prefix Language Modeling

May

the force

targets

be with you

Masked Language Modeling

May

targets

the force

be with you

Extra Conditioning Signals

- Metadata: e.g. topic control codes, timestamp, web domain, data source
- Classifier predictions: e.g. toxicity, quality

Horror *A knife* handle pulled through the open hole in the front. I jumped when the knife hit.
Eyes widened in horror. Her scream was the only sound I heard besides her sobs.
The spider touched her feet as it started to dig into the top of her arch. The creature's tears began to flow.
The spider looked up to her and looked back at me with eyes filled with burning tears. My heart started to race. . .

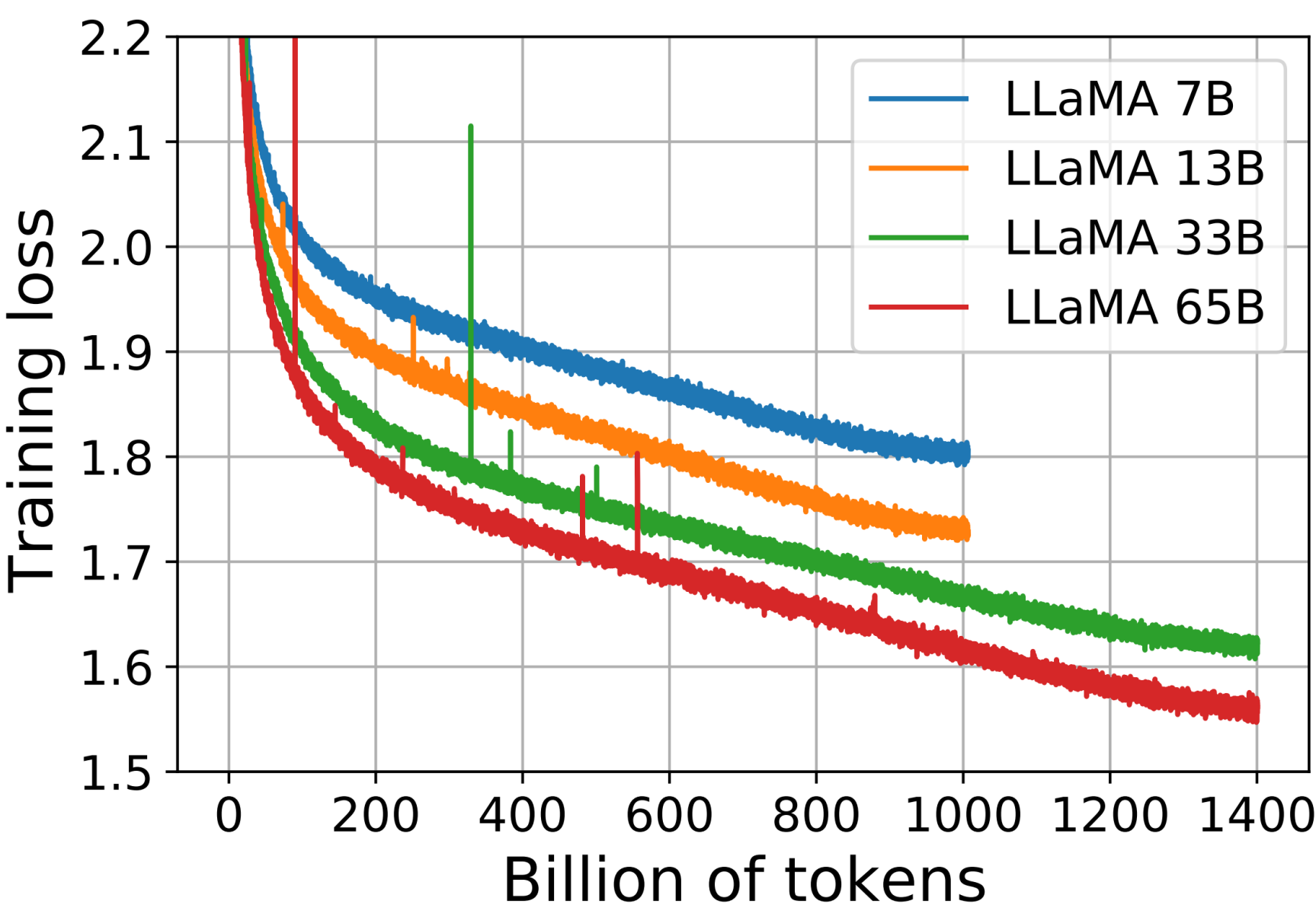
Reviews *A knife* is a tool and this one does the job well.
Rating: 4.0
I bought these for my husband who has been using them to cut up his own meat since he got them. He says they are very sharp so be careful when you use them, but that doesn't seem like much of an issue because he's used it on everything from chicken breasts to beef tenderloin. . .

Other hyper-parameters

- Training duration
- Learning rate
- Batch size
- Precision

Decisions: model size

LLaMA:

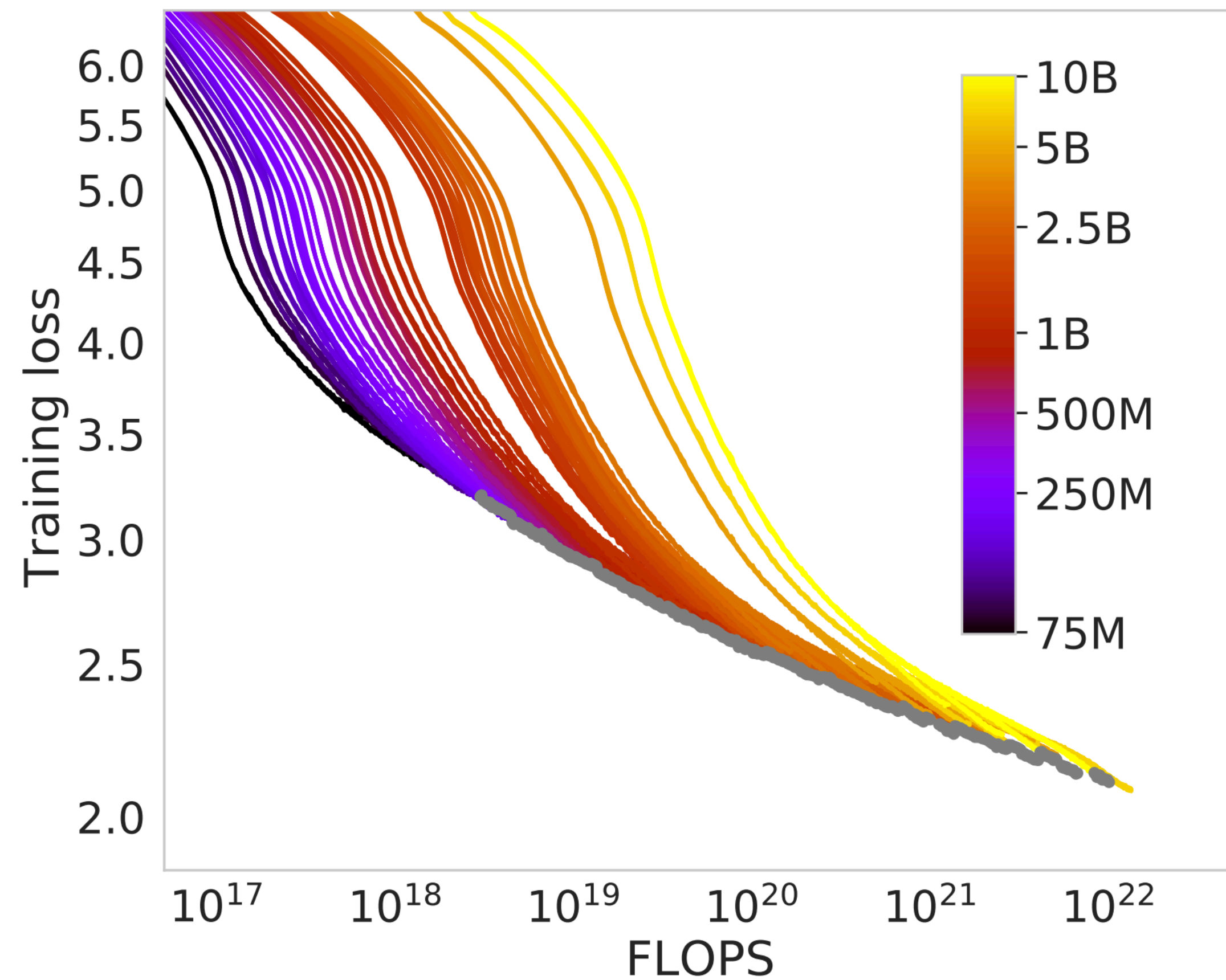


GPT-2:

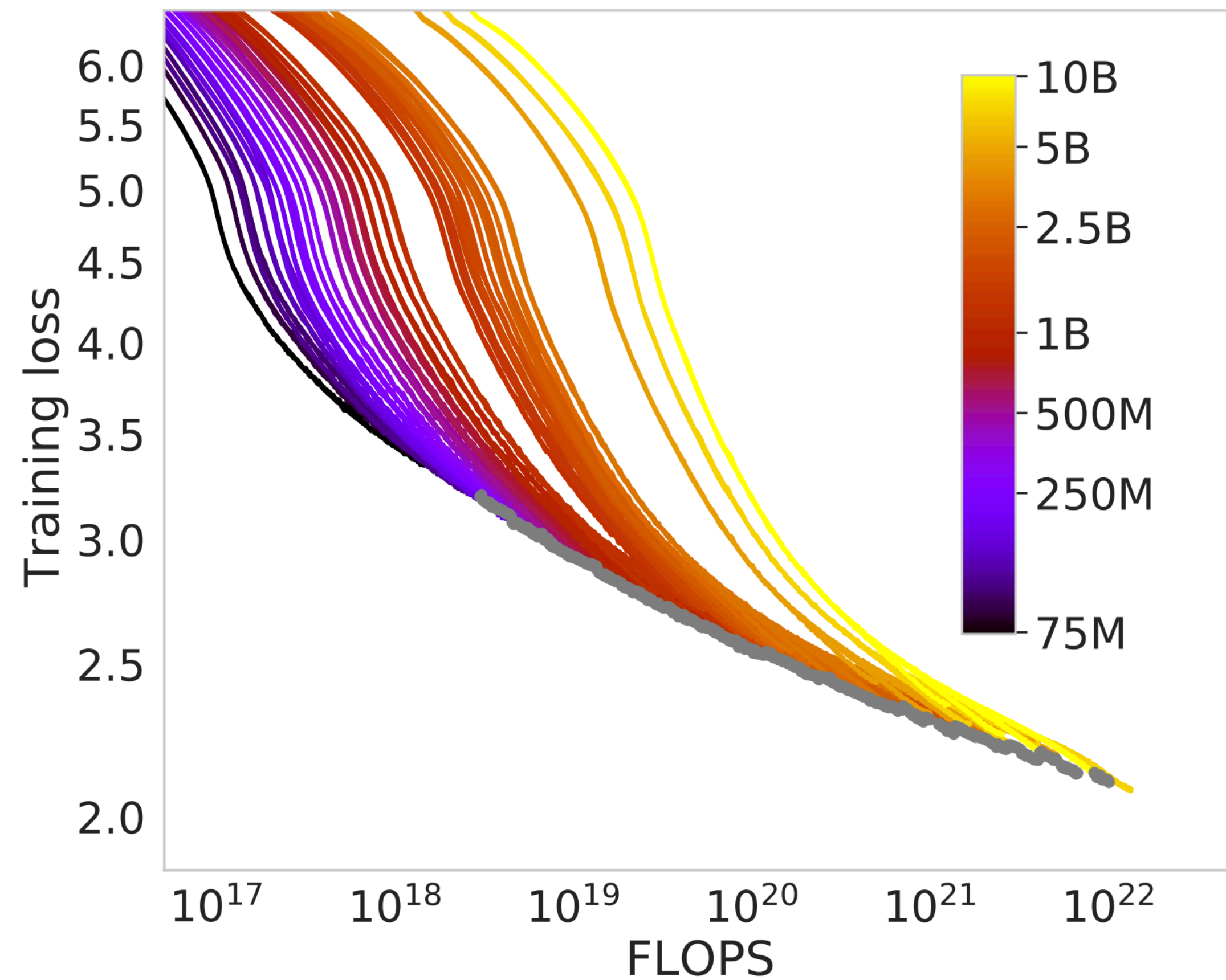
	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	21.8
117M	35.13	45.99	87.65	83.4	29.41	65.85	1.16	1.17	37.50	75.20
345M	15.60	55.48	92.35	87.1	22.76	47.33	1.01	1.06	26.37	55.72
762M	10.87	60.12	93.45	88.0	19.93	40.31	0.97	1.02	22.05	44.575
1542M	8.63	63.24	93.30	89.05	18.34	35.76	0.93	0.98	17.48	42.16

Table 3. Zero-shot results on many datasets. No training or fine-tuning was performed for any of these results. PTB and WikiText-2

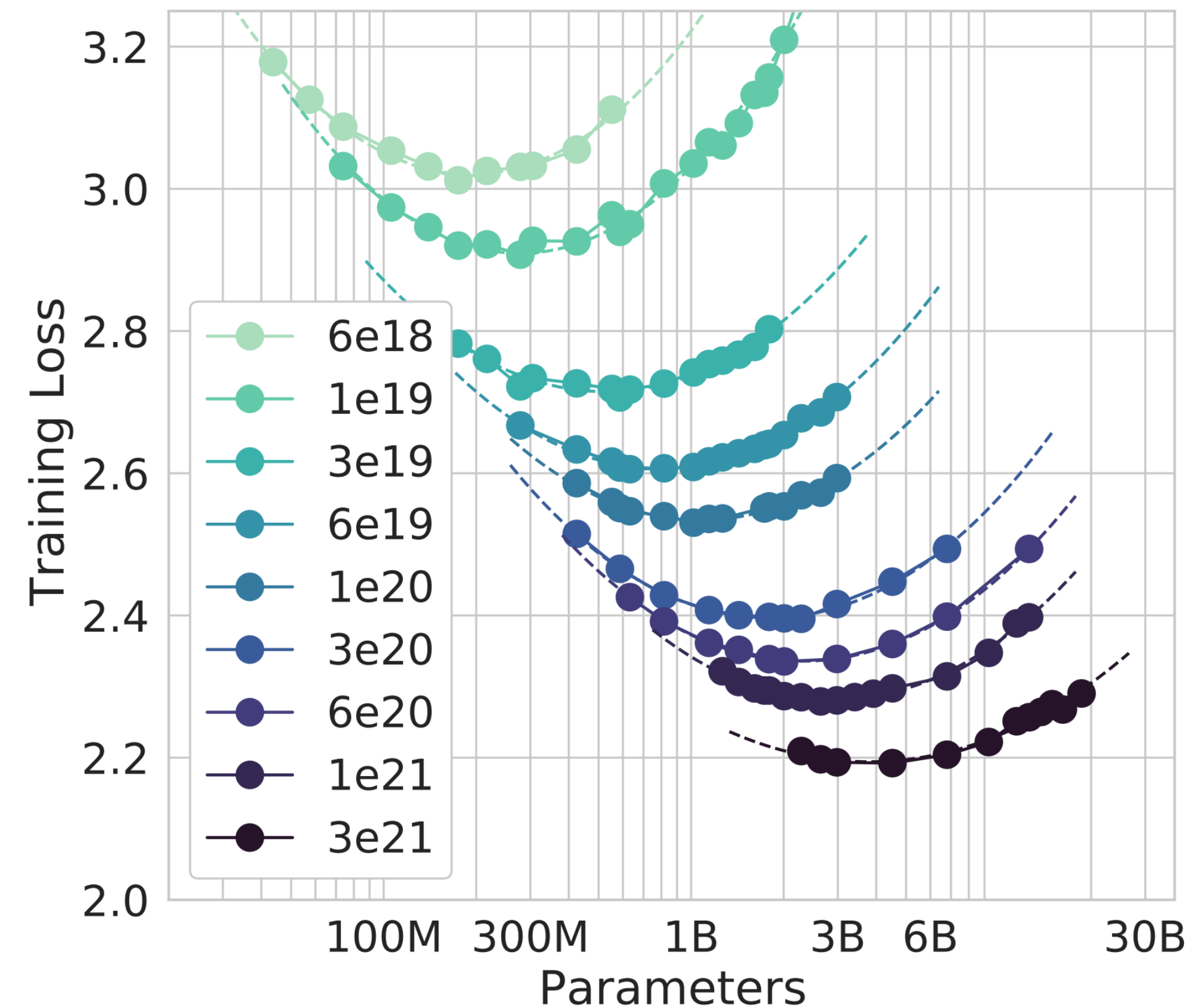
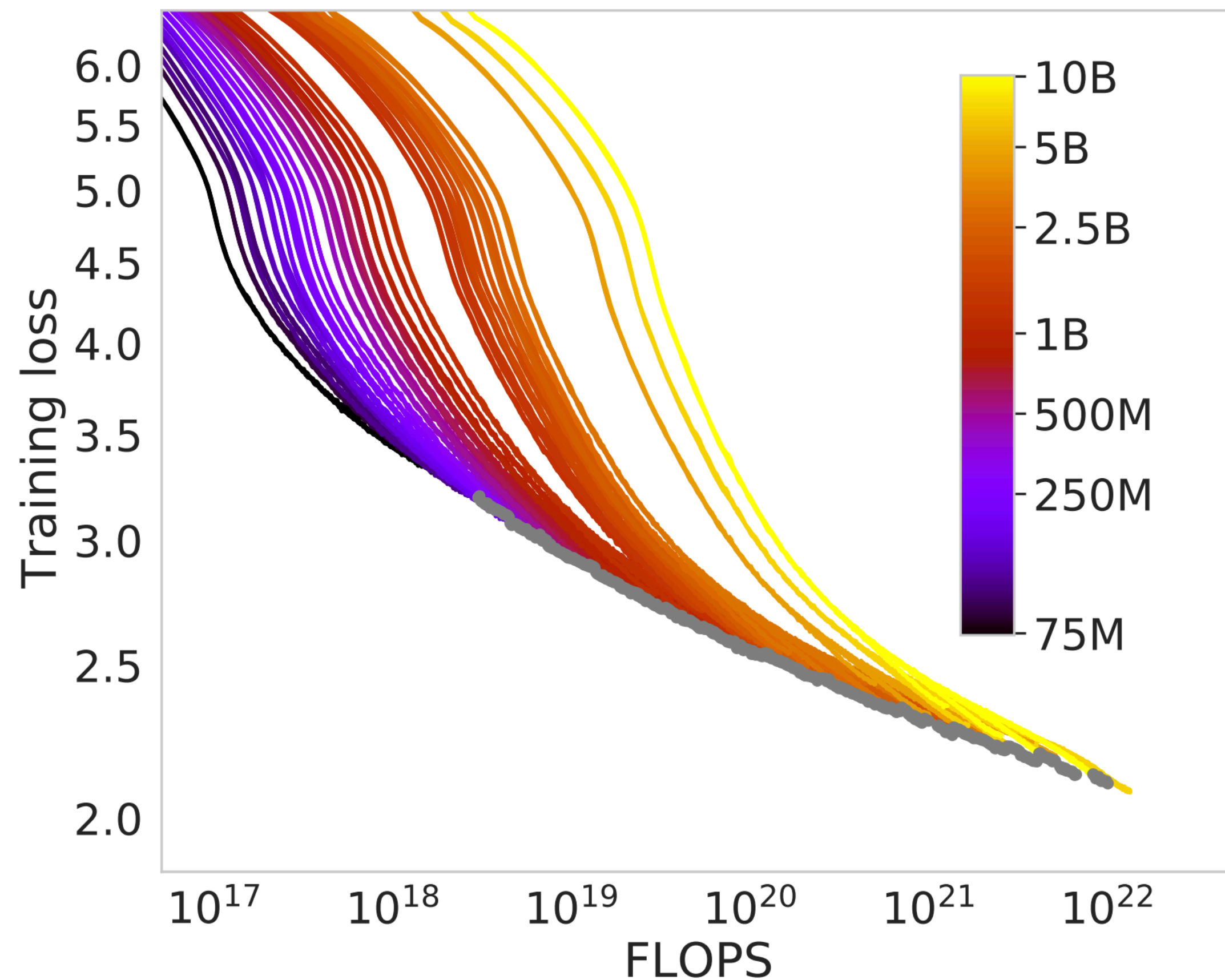
Decisions: training duration



Decisions: training duration



Decisions: training duration



Given a fixed FLOPs budget, how should one trade-off model size and the number of training tokens?

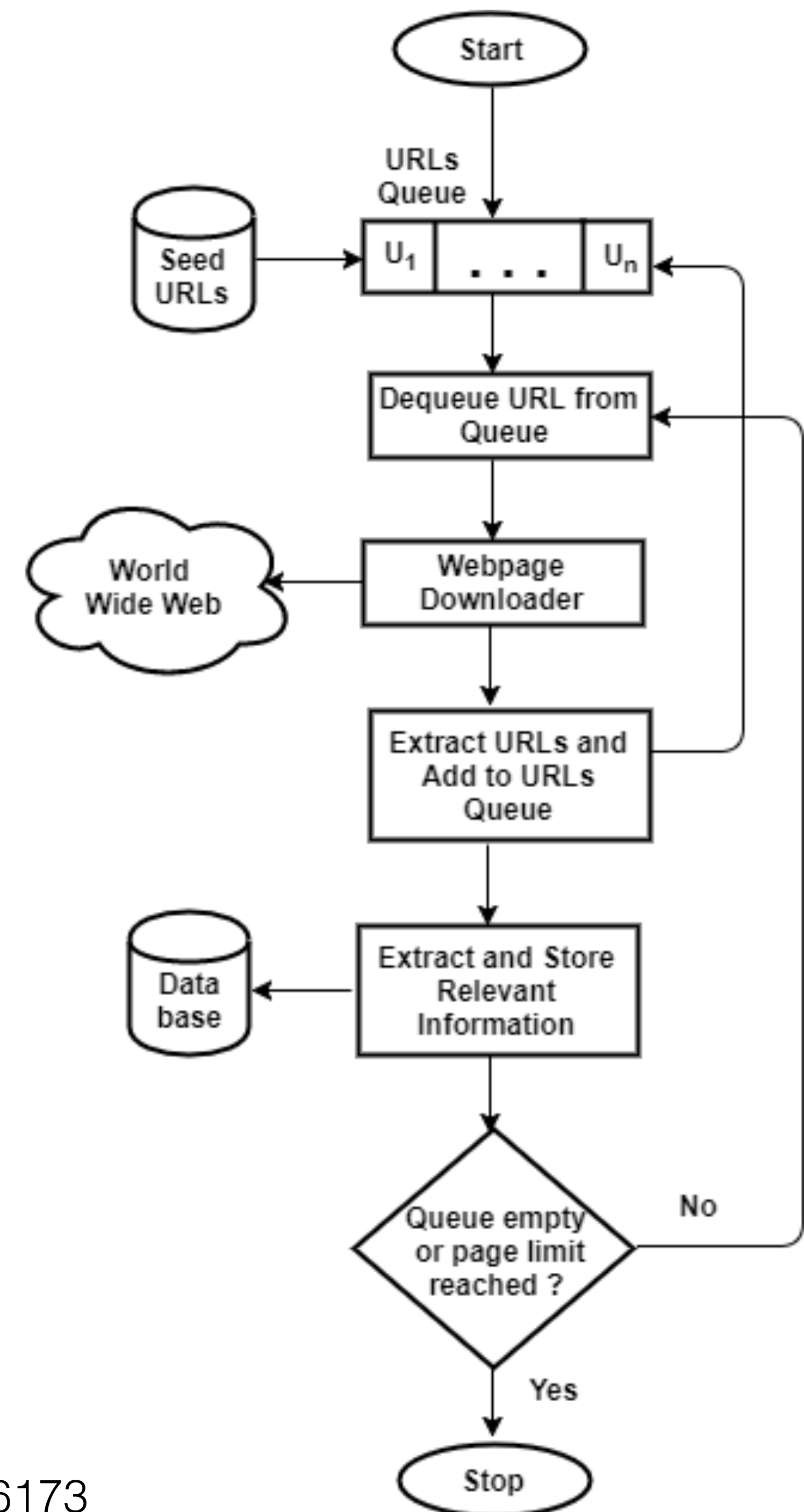
Compute-optimal models

- Goal: to find $N_{opt}(C), D_{opt}(C) = \underset{N, D \text{ s.t. } \text{FLOPs}(N, D) = C}{\operatorname{argmin}} L(N, D)$
 - where N is # model params, D is # training tokens, C is compute budget.
- Chinchilla approach:
 - Empirically estimate $N_{opt}(C), D_{opt}(C)$ based on the losses of over 400 models trained under various conditions
 - *More on scaling laws in Thursday's lecture.*

Assembling a pre-training dataset

Webcrawls

- Originally designed for search engines
- Find URLs to crawl from already-crawled URLs
- Design heuristics to decide which URLs are worth crawling



Common Crawl

- Largest public web crawl
- Every month, additional URLs are crawled
- Pages determined by heuristics to be fast-updating get crawled more frequently.

Common Crawl maintains a **free, open repository** of web crawl data that can be used by anyone.

Common Crawl is a 501(c)(3) non-profit founded in 2007.

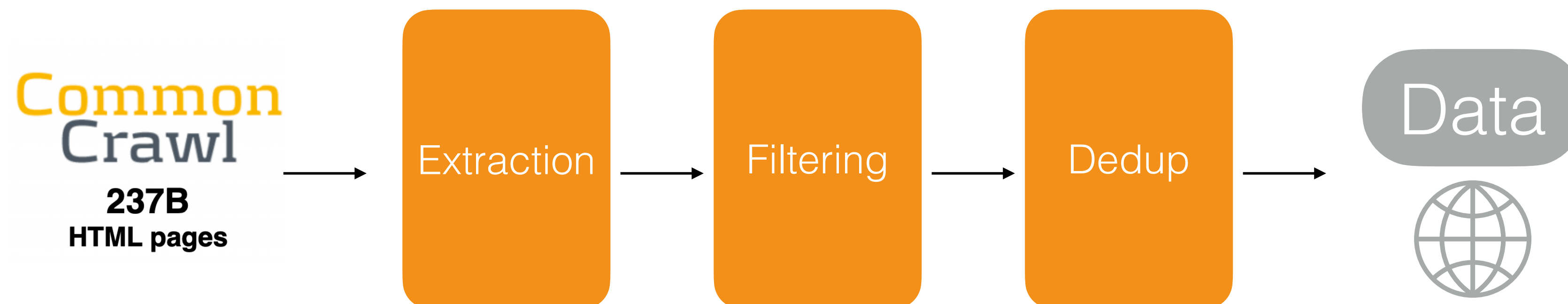
We make wholesale extraction, transformation, and analysis of open web data accessible to researchers.

WebText (GPT-2)

- Crawl non-Wikipedia URLs linked to from Reddit posts with at least 3 karma.

From webcrawl to training data

- **Extraction:** HTML to text
- **Filtering:** filter out unwanted pages
- **Deduplication:** many duplicate web pages



Other web-derived data sources

- Public Github repos
- Newspapers
- arxiv papers
- Public social media posts
- Fan fiction

How does web data become **unavailable**?

- Introduction of paywalls



How does web data become **unavailable**?

- Introduction of paywalls
- Restrictive terms of service

The New York Times

2. CONTENT ON THE SERVICES

2.1 The contents of the Services, including the Site, are intended for your personal, non-commercial use. All materials published or available on the Services (including, but not limited to text, photographs, images, illustrations, designs, audio clips, video clips, “look and feel,” metadata, data, or compilations, all also known as the “**Content**”) are protected by copyright, and owned or controlled by The New York Times Company or the party credited as the provider of the Content. The New York Times Company also owns copyright in the selection, coordination, compilation, and enhancement of such Content (“**Arrangement**”). You shall abide by all additional copyright notices, information, or restrictions contained in any Content accessed through the Service. Non-commercial use does not include the use of Content without prior written consent from The New York Times Company in connection with: (1) the development of any software program, model, algorithm, or other generative AI tool, including, but not limited to, training or using the Content in connection with the development or operation of a machine learning or artificial intelligence (AI) system (including any use of the Content for training, fine tuning, or grounding the machine learning or AI system or as part of retrieval-augmented generation); or (2) providing archived or cached data sets containing Content to another person or entity.

How does web data become **unavailable**?

- Introduction of paywalls
- Restrictive terms of service
- Implementation of Robots Exclusion Protocol

Reddit to update web standard to block automated website scraping

By Reuters

June 25, 2024 4:45 PM EDT · Updated 2 months ago



Reddit's logo is displayed, at the New York Stock Exchange (NYSE) in New York City, U.S., March 21, 2024. REUTERS/Brendan McDermid/File Photo [Purchase Licensing Rights](#)

Companies



Reddit Inc

Follow

June 25 (Reuters) - Social media platform Reddit ([RDDT.N](#)) said on Tuesday it will update a web standard used by the platform to block automated data scraping from its website, following reports that AI startups were bypassing the rule to gather content for their systems.

The move comes at a time when artificial intelligence firms have been accused of plagiarizing content from publishers to create AI-generated summaries without giving credit or asking for permission.

Reddit's robot.txt file in 2019

```
# 80legs
User-agent: 008
Disallow: /

# 80legs' new crawler
User-agent: voltron
Disallow: /

User-Agent: bender
Disallow: /my_shiny_metal_ass

User-Agent: Gort
Disallow: /earth

User-agent: MJ12bot
Disallow: /

User-agent: PiplBot
Disallow: /

User-Agent: *
Disallow: /*.json
Disallow: /*.json-compact
Disallow: /*.json-html
Disallow: /*.xml
```

Jokes

Disallow malicious webcrawlers

Disallow paths that aren't useful to crawl

```
Disallow: /*.rss
Disallow: /*.i
Disallow: /*.embed
Disallow: /*/comments/*?*sort=
Disallow: /r/*/comments/*/*/c*
Disallow: /comments/*/*/c*
Disallow: /r/*/submit
Disallow: /message/compose*
Disallow: /api
Disallow: /post
Disallow: /submit
Disallow: /goto
Disallow: /*after=
Disallow: /*before=
Disallow: /domain/*t=
Disallow: /login
Disallow: /reddits/search
Disallow: /search
Disallow: /r/*/search
Disallow: /r/*/user/
Disallow: /gold?
Allow: /partner_api/
Allow: /
Allow: /sitemaps/*.xml
```

Disallow paths that aren't useful to crawl

Disallow paths that would put webcrawler into an endless loop

Allow most scraping

Reddit's robot.txt file in September 2024

```
# Welcome to Reddit's robots.txt
# Reddit believes in an open internet, but not the misuse of public content.
# See https://support.reddithelp.com/hc/en-us/articles/26410290525844-Public-Content-Policy Reddit's Public
Content Policy for access and use restrictions to Reddit content.
# See https://www.reddit.com/r/reddit4researchers/ for details on how Reddit continues to support research and
non-commercial use.
# policy: https://support.reddithelp.com/hc/en-us/articles/26410290525844-Public-Content-Policy

User-agent: *
Disallow: /
```

How does web data become **unavailable**?

- Introduction of paywalls
- Restrictive terms of service
- Implementation of Robots Exclusion Protocol
- Lawsuits and litigation

UNITED STATES DISTRICT COURT
SOUTHERN DISTRICT OF NEW YORK

AUTHORS GUILD, DAVID BALDACCI,
MARY BLY, MICHAEL CONNELLY, SYLVIA
DAY, JONATHAN FRANZEN, JOHN
GRISHAM, ELIN HILDERBRAND,
CHRISTINA BAKER KLINE, MAYA
SHANBHAG LANG, VICTOR LAVALLE,
GEORGE R.R. MARTIN, JODI PICOULT,
DOUGLAS PRESTON, ROXANA ROBINSON,
GEORGE SAUNDERS, SCOTT TUROW, and
RACHEL VAIL, individually and on behalf of
others similarly situated,

Plaintiffs,

v.

OPENAI INC., OPENAI LP, OPENAI LLC,
OPENAI GP LLC, OPENAI OPCO LLC,
OPENAI GLOBAL LLC, OAI CORPORATION
LLC, OPENAI HOLDINGS LLC, OPENAI
STARTUP FUND I LP, OPENAI STARTUP
FUND GP I LLC, and OPENAI STARTUP
FUND MANAGEMENT LLC,

Defendants.

No. 1:23-cv-8292

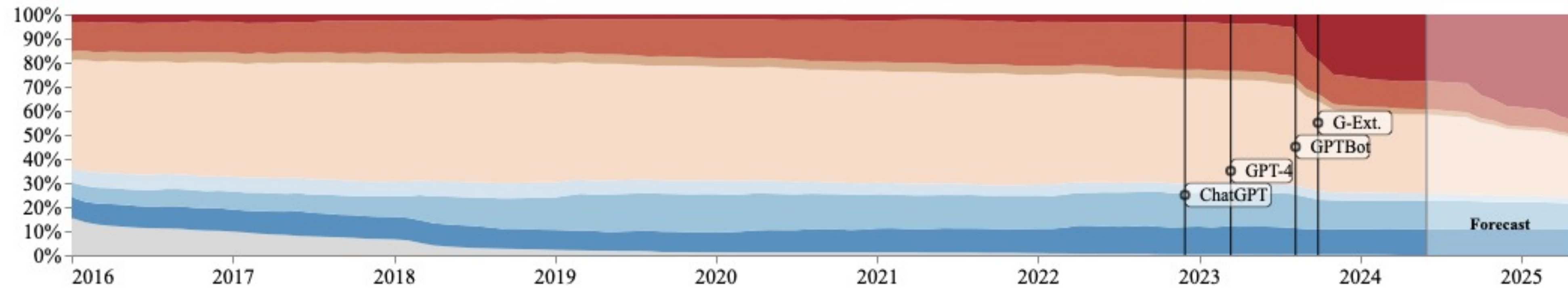
CLASS ACTION COMPLAINT

JURY TRIAL DEMANDED

INTRODUCTORY STATEMENT

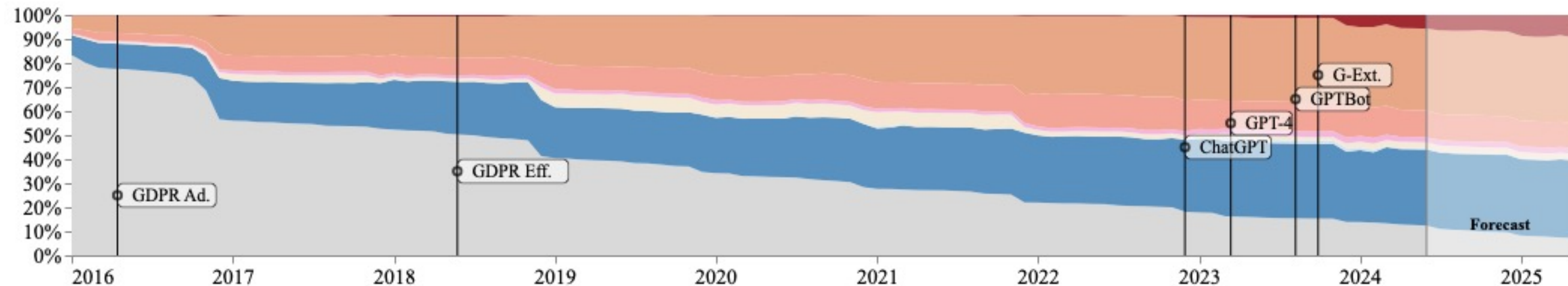
1. Plaintiffs, authors of a broad array of works of fiction, bring this action under the Copyright Act seeking redress for Defendants' flagrant and harmful infringements of Plaintiffs' registered copyrights in written works of fiction. Defendants copied Plaintiffs' works wholesale, without permission or consideration. Defendants then fed Plaintiffs' copyrighted works into their "large language models" or "LLMs," algorithms designed to output human-seeming text responses to users' prompts and queries. These algorithms are at the heart of Defendants' massive commercial enterprise. And at the heart of these algorithms is systematic theft on a mass scale.

Datasets created in the past may not be creatable



Robots.txt Restrictions

Full restrictions Pattern-based restrictions Disallow private directories Other restrictions Crawl delay specified Sitemap provided
No restrictions or sitemap No Robots.txt

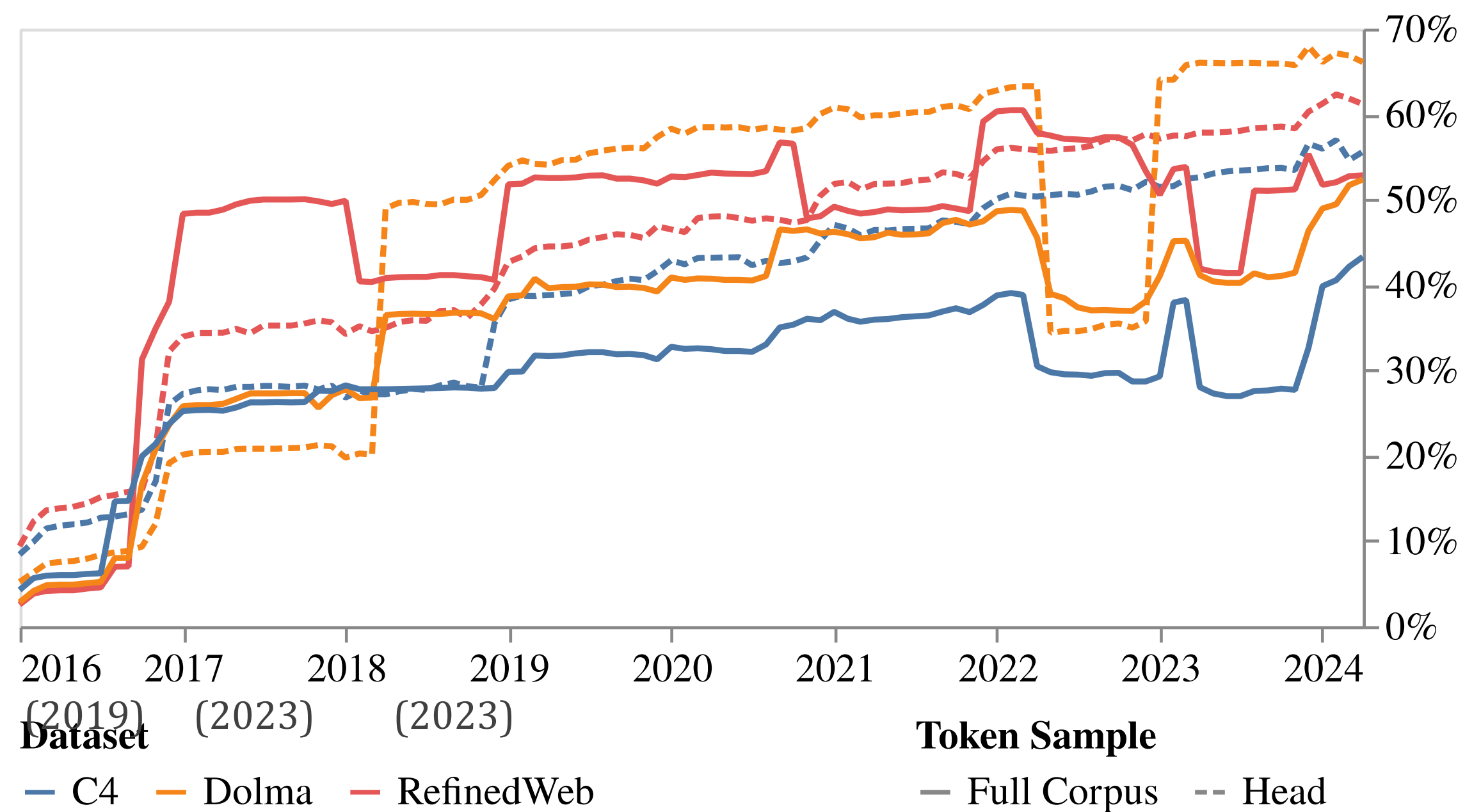


ToS Restrictions

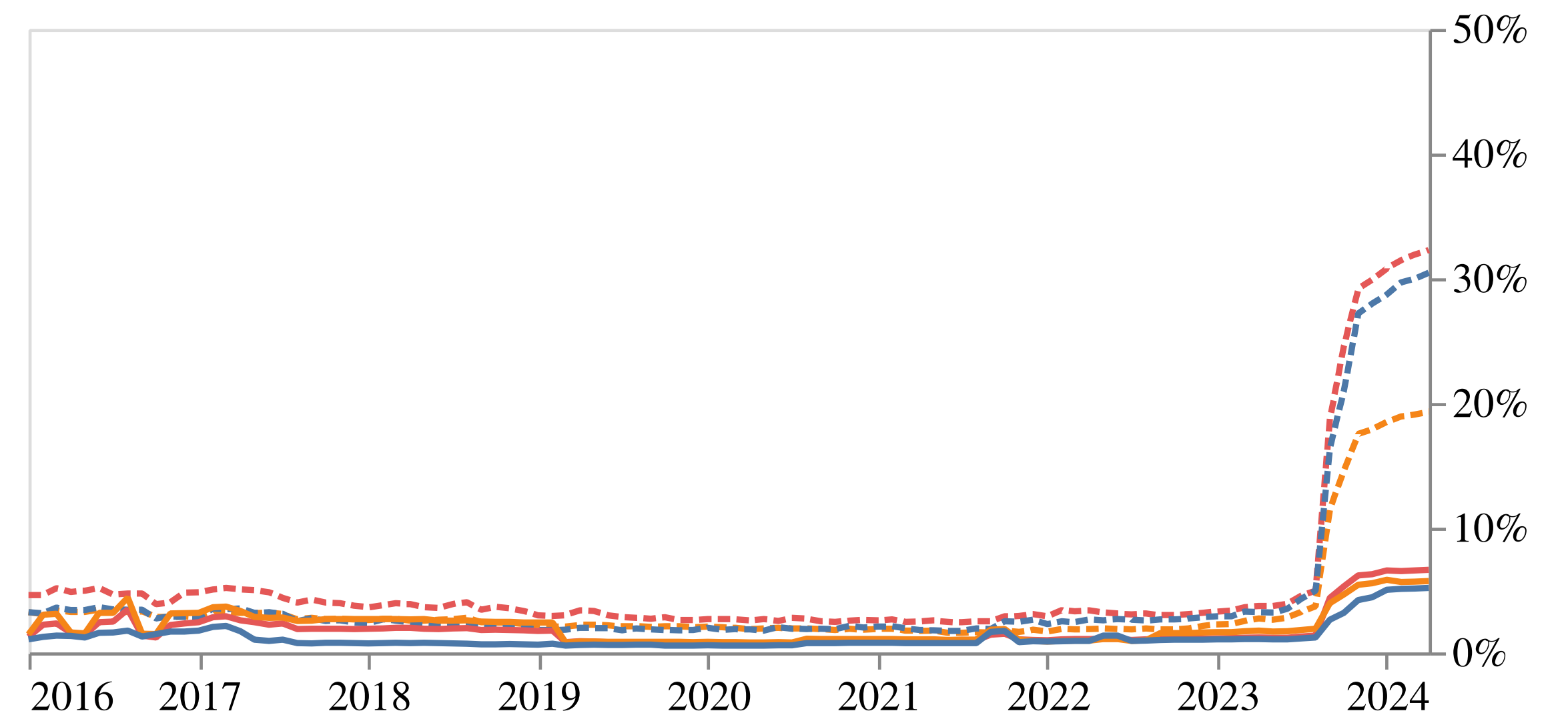
No Crawling & AI No Crawling No AI Non-Commercial Use Non-Compete No Re-Distribution
Conditional Use Unrestricted Use No Terms Pages

Datasets created in the past may not be creatable today.

Percentage of examples in dataset restricted by Terms of Service

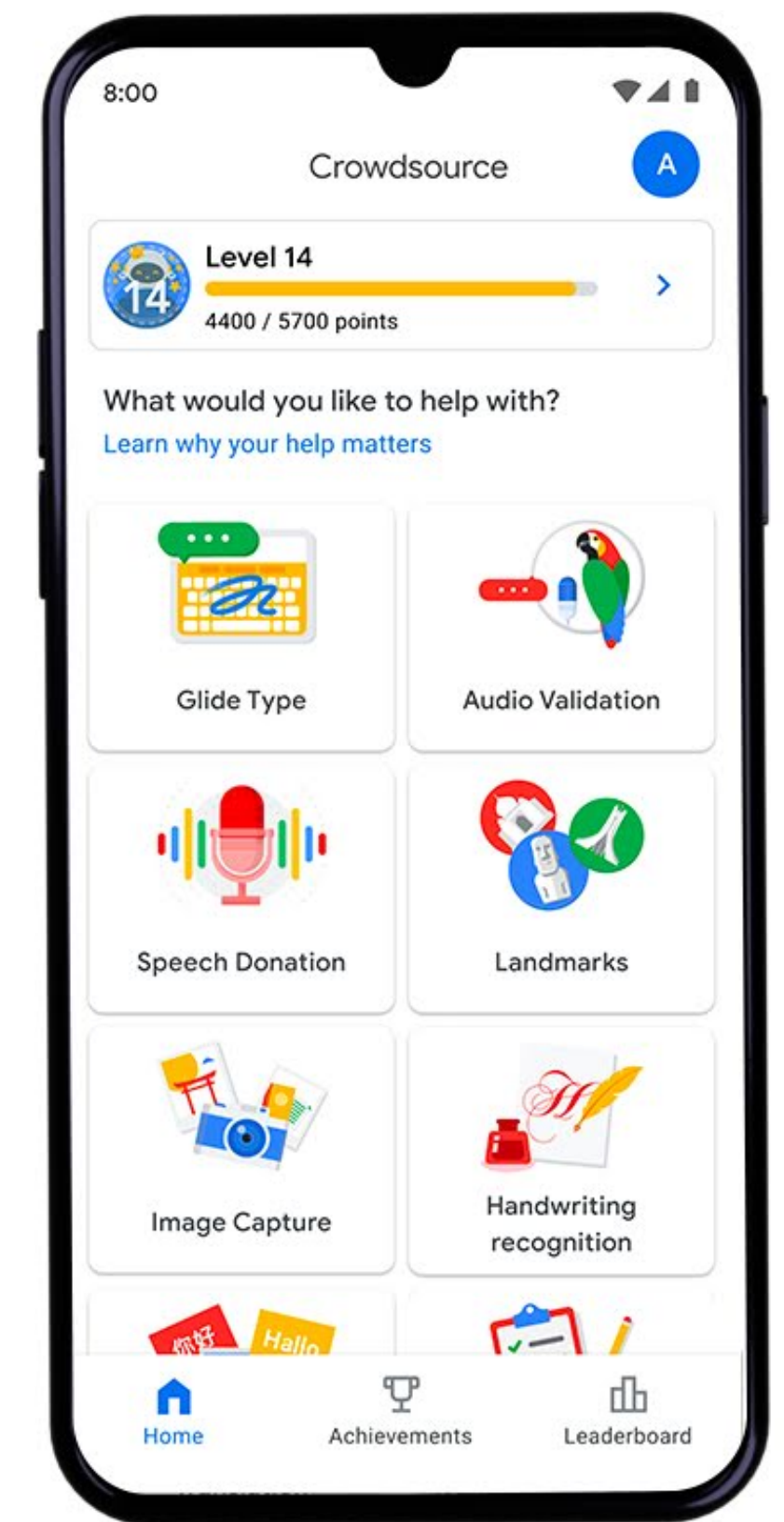


Percentage of examples in dataset restricted by robots.txt



Other non-web data sources

- Books
- Internal company data (e.g. Facebook posts)
- Crowdsourcing



Mixtures of Data Sources

- In practice, training data is a mixture of different sources

Source	Type	9T Pool		6T Mix	
		Tokens	Docs	Tokens	Docs
Common Crawl	Web pages	8.14T	9.67B	4.51T (76.1%)	3.15B
OLMOCR science PDFs	Academic documents	972B	101M	805B (13.6%)	83.8M
Stack-Edu (Rebalanced)	GitHub code	137B	167M	409B (6.89%)	526M
arXiv	Papers with LaTeX	21.4B	3.95M	50.8B (0.86%)	9.10M
FineMath 3+	Math web pages	34.1B	21.4M	152B (2.56%)	95.5M
Wikipedia & Wikibooks	Encyclopedic	3.69B	6.67M	2.51B (0.04%)	4.24M
Total		9.31T	9.97B	5.93T (100%)	3.87B

Table 4 Composition of Dolma 3 Mix including our 9T pool of data, the 6T mix we used for f

Some Pre-Training Mixtures

GPT-3

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

Table 2.2: Datasets used to train GPT-3. “Weight in training mix” refers to the fraction of examples during training that are drawn from a given dataset, which we intentionally do not make proportional to the size of the dataset. As a result, when we train for 300 billion tokens, some datasets are seen up to 3.4 times during training while other datasets are seen less than once.

OLMO

Source	Doc Type	UTF-8 bytes (GB)	Documents (millions)	Unicode words (billions)	Llama tokens (billions)
Common Crawl	 web pages	9,812	3,734	1,928	2,479
GitHub	 code	1,043	210	260	411
Reddit	 social media	339	377	72	89
Semantic Scholar	 papers	268	38.8	50	70
Project Gutenberg	 books	20.4	0.056	4.0	6.0
Wikipedia, Wikibooks	 encyclopedic	16.2	6.2	3.7	4.3
Total		11,519	4,367	2,318	3,059

LLaMA 1

Dataset	Sampling prop.	Epochs	Disk size
CommonCrawl	67.0%	1.10	3.3 TB
C4	15.0%	1.06	783 GB
Github	4.5%	0.64	328 GB
Wikipedia	4.5%	2.45	83 GB
Books	4.5%	2.23	85 GB
ArXiv	2.5%	1.06	92 GB
StackExchange	2.0%	1.03	78 GB

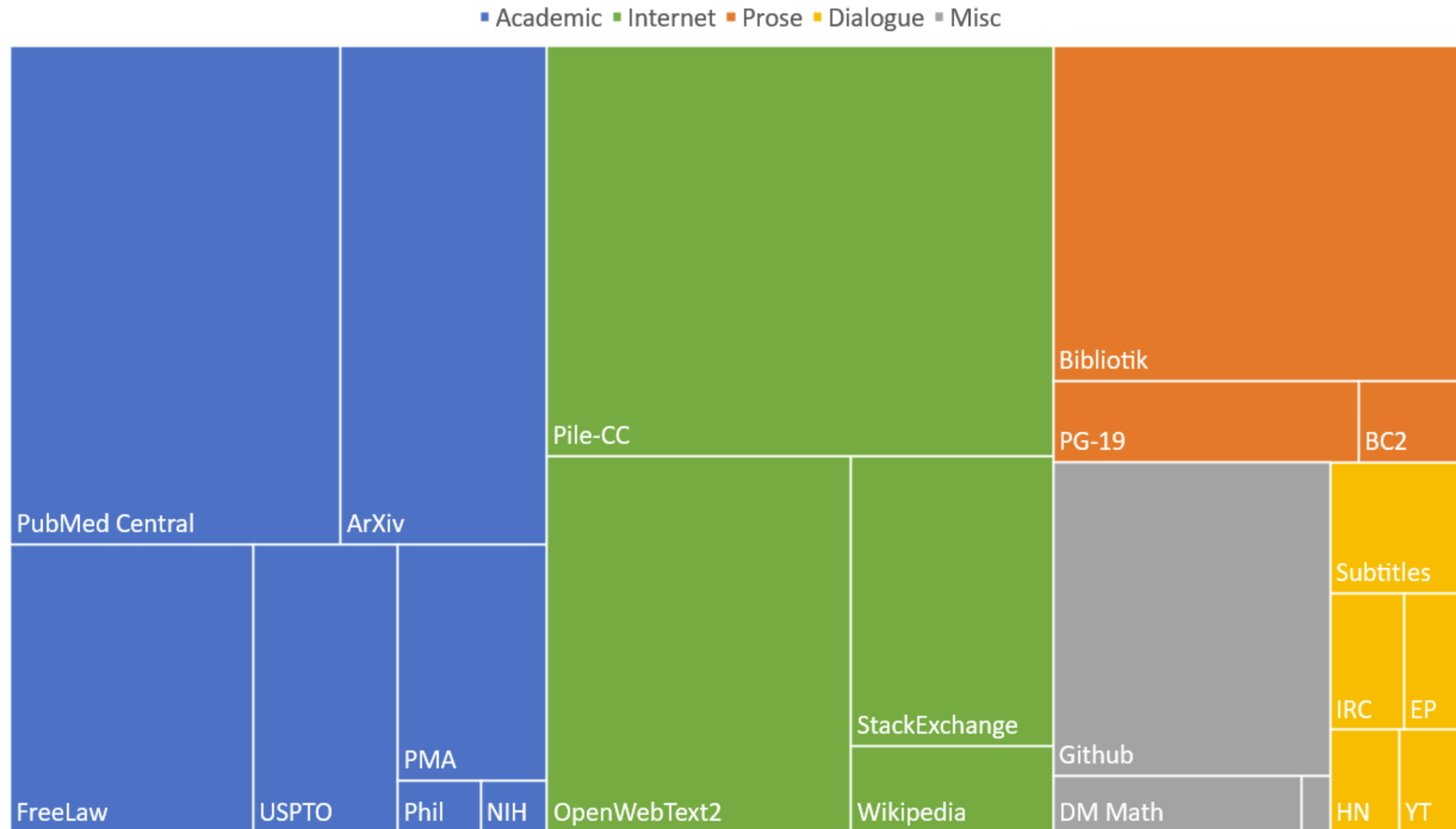
Table 1: **Pre-training data.** Data mixtures used for pre-training, for each subset we list the sampling proportion, number of epochs performed on the subset when training on 1.4T tokens, and disk size. The pre-training runs on 1T tokens have the same sampling proportion.

LLaMA 3

% tokens	Type
50%	General knowledge
25%	Math and reasoning
17%	Code
8%	Multilingual

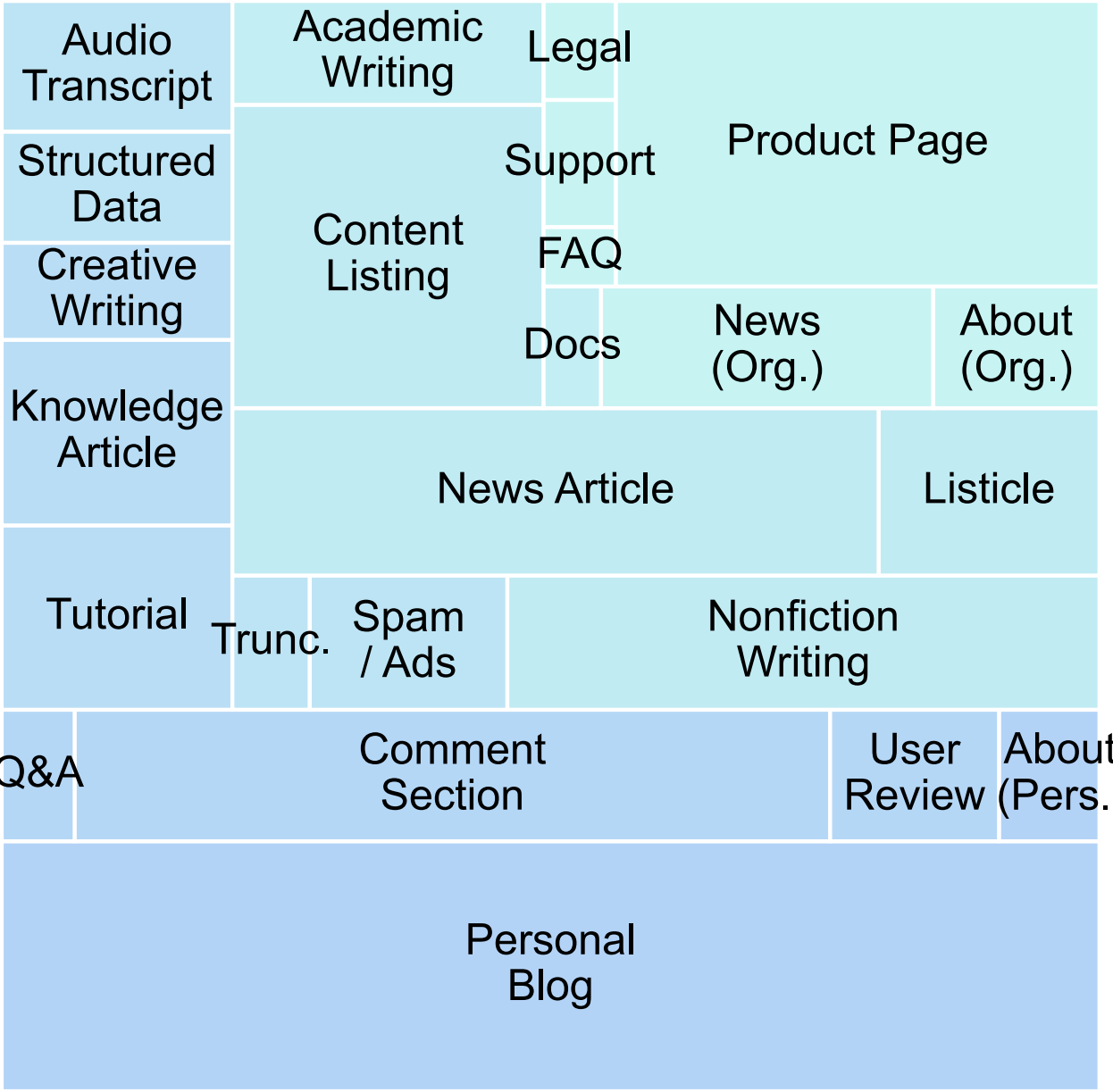
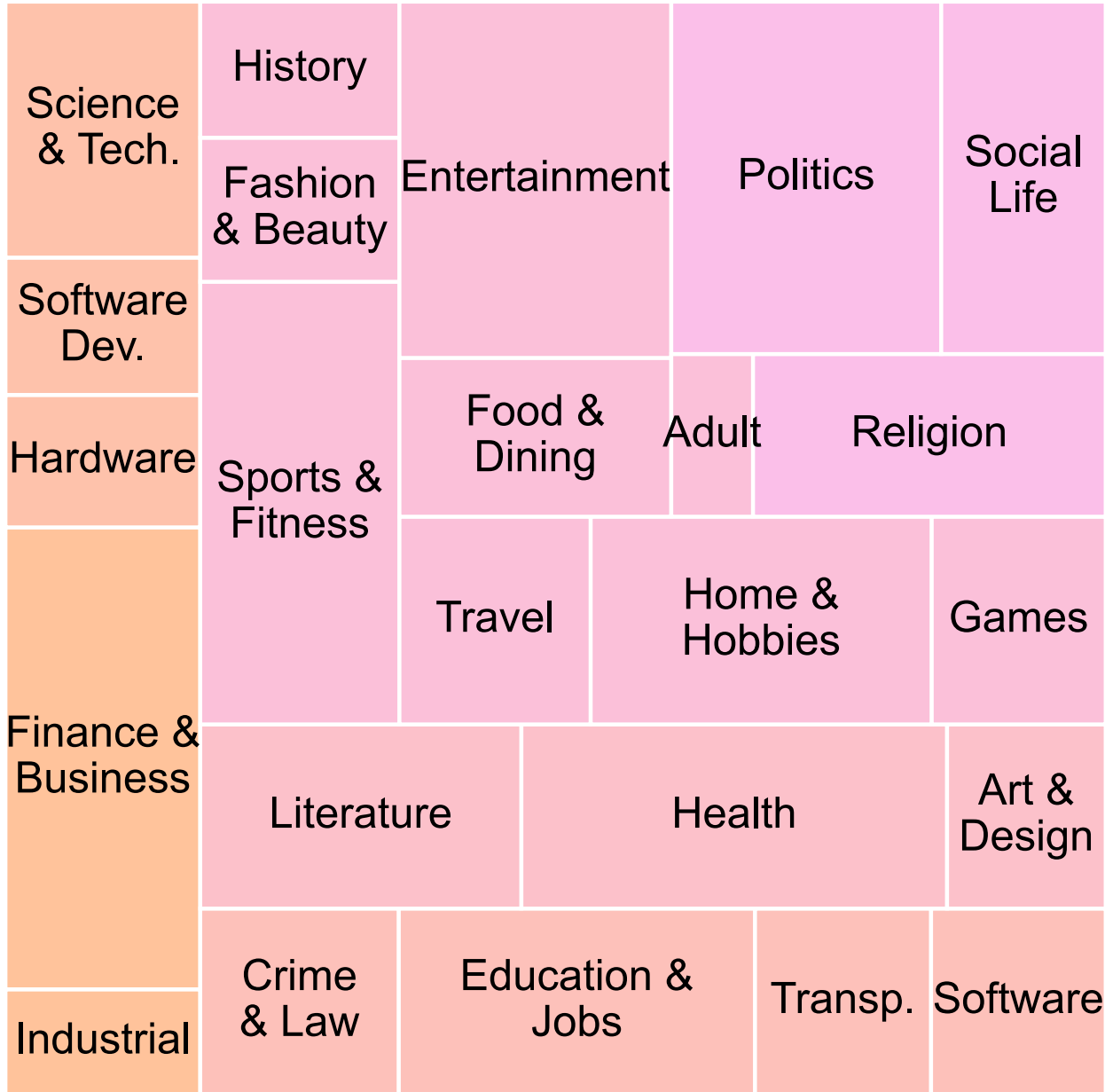
Other Web-Derived Data Sources

Composition of the Pile by Category

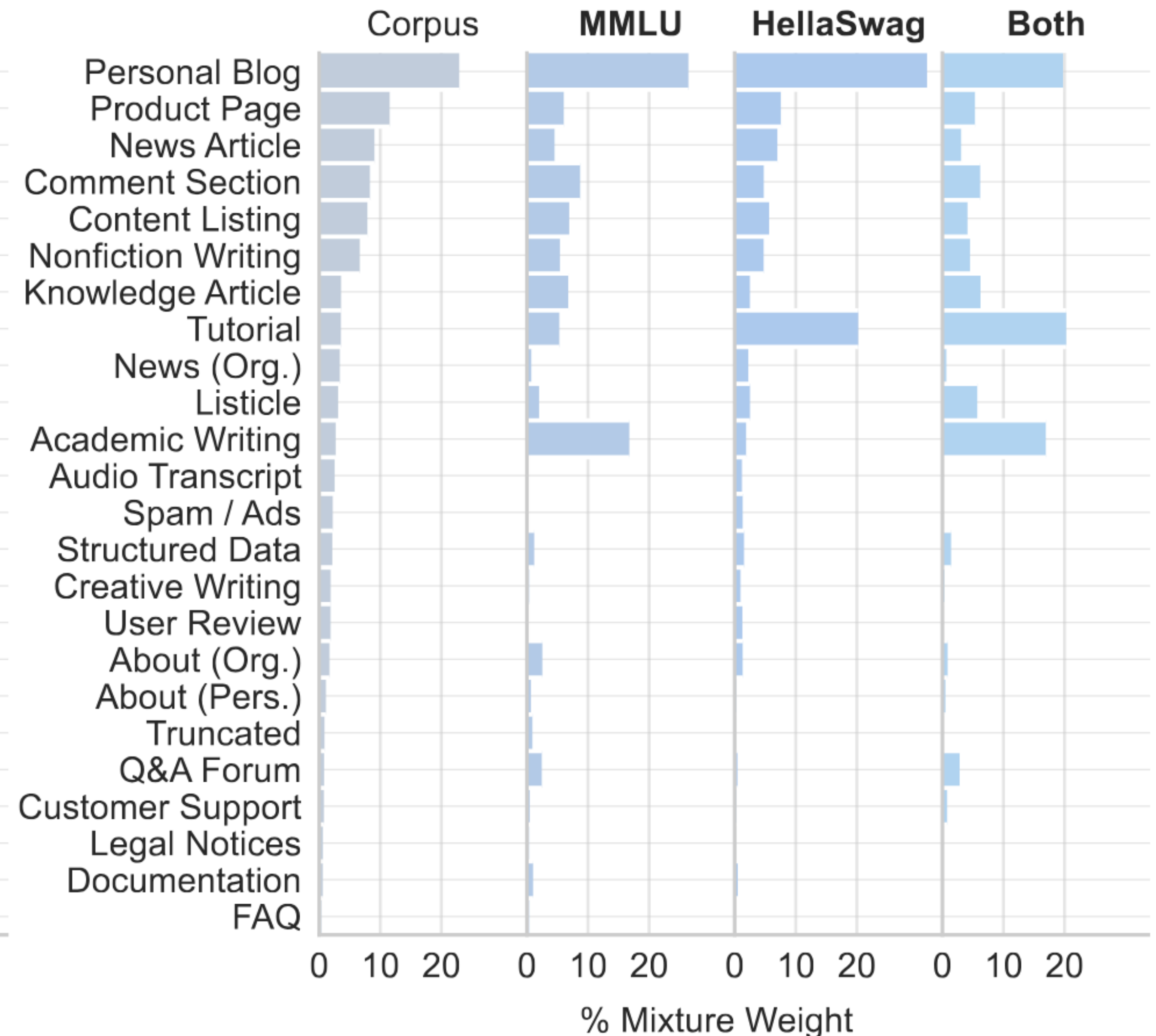
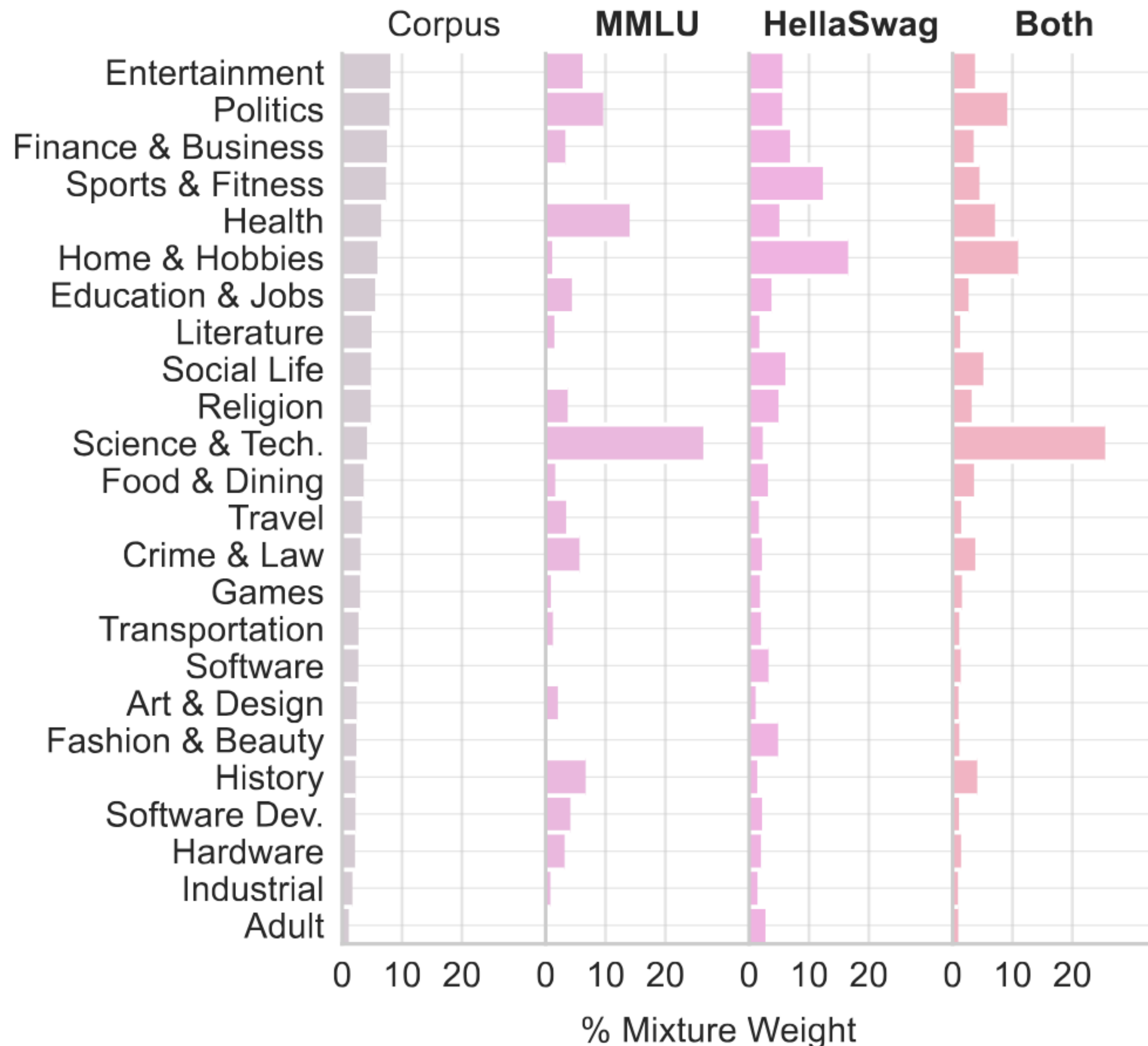


How to choose the mixture?

- Automatically classify documents (by topic, format, quality, etc.)
- Train many small models, each with a different mixture over the classes
- For each evaluation task, fit a regression model $f(\text{mixture}) \rightarrow \text{task performance}$
- Solve for mixture that minimizes average task performance subject to constraints (token budget, maximum repeats)
- Favor high-quality documents when upsampling

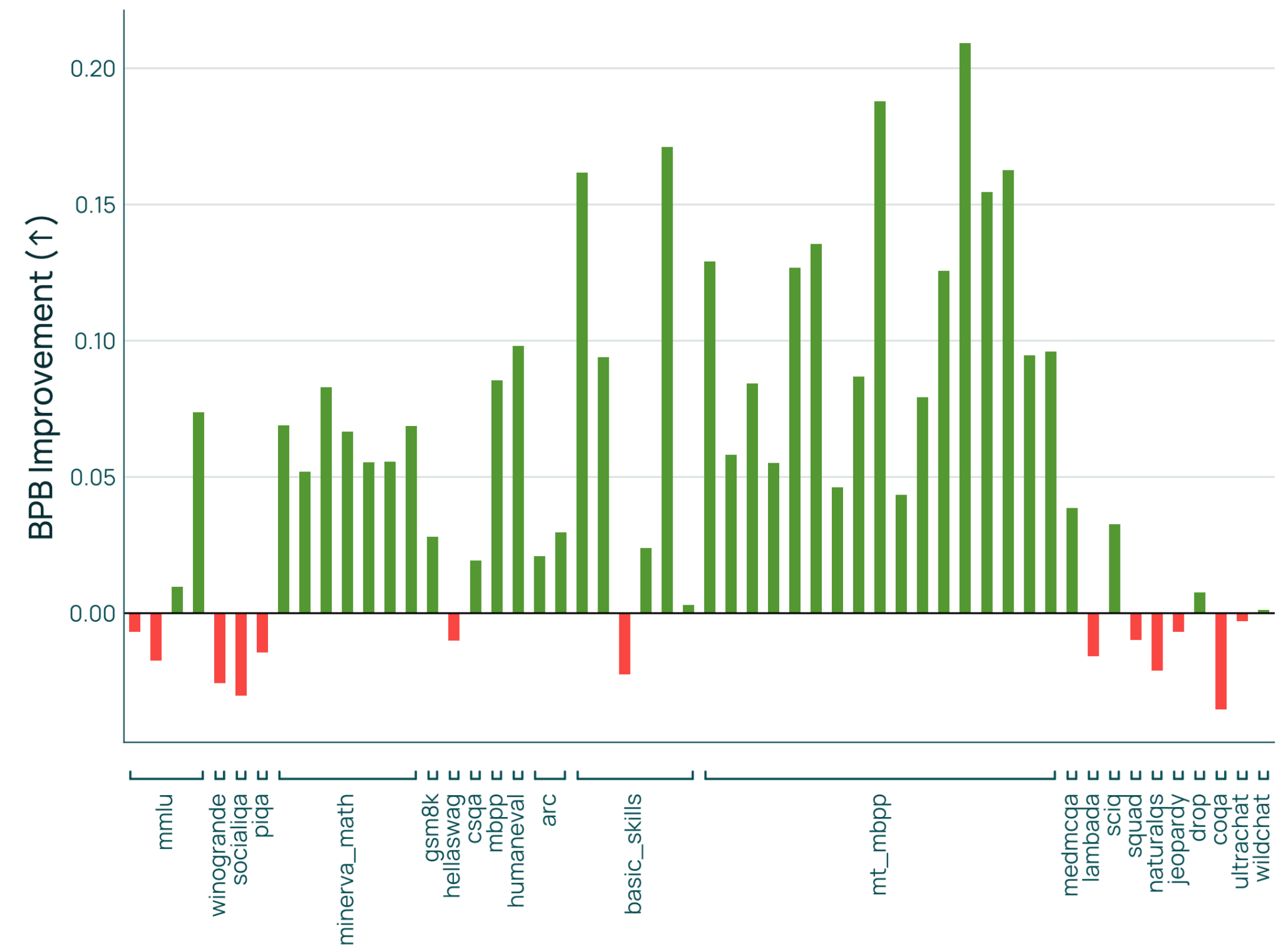
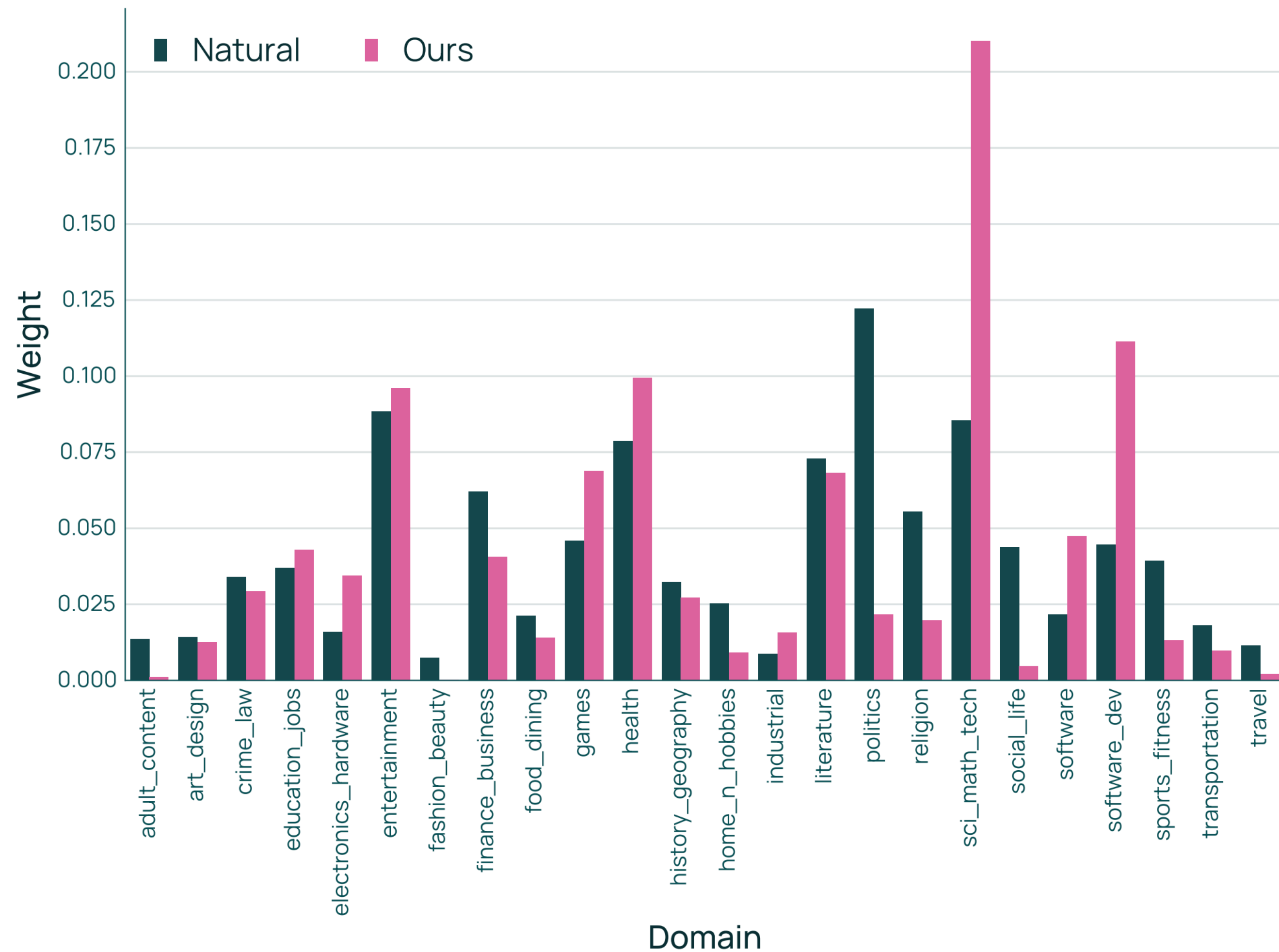


How to choose the mixture?



Wettig et al. 2025. "Organize the Web: Constructing Domains Enhances Pre-Training Data Curation."

How to choose the mixture?



Recent examples of public pre-training datasets

	Year	Domain	Tokens
FineWeb	2024	Web	15 trillion
RedPajama v2	2024	Web	30 trillion
Dolma	2024	Mix	3 trillion
OLMO2 Mix	2025	Mix	4 trillion
DOLMA 3	2025	Mix	6 trillion
OpenWebMath	2023	Math web pages	15 billion
AlgebraicStack	2023	Math code	11 billion
FineWeb-Edu	2024	Educational (middle-school)	1.4 trillion

Importance of Coverage

Coverage: Does the data cover the domain(s) I care about, and in the right proportions?

- The data determines the data distribution
 - And hence the model, $p_{\theta} \approx p_{data}$
- Web data $\neq$ math data
- Web data $\neq$ educational data
- Web data $\neq$ code data
- ...

Cleaning, filtering, and quality control

Dolma 3's data prep pipeline

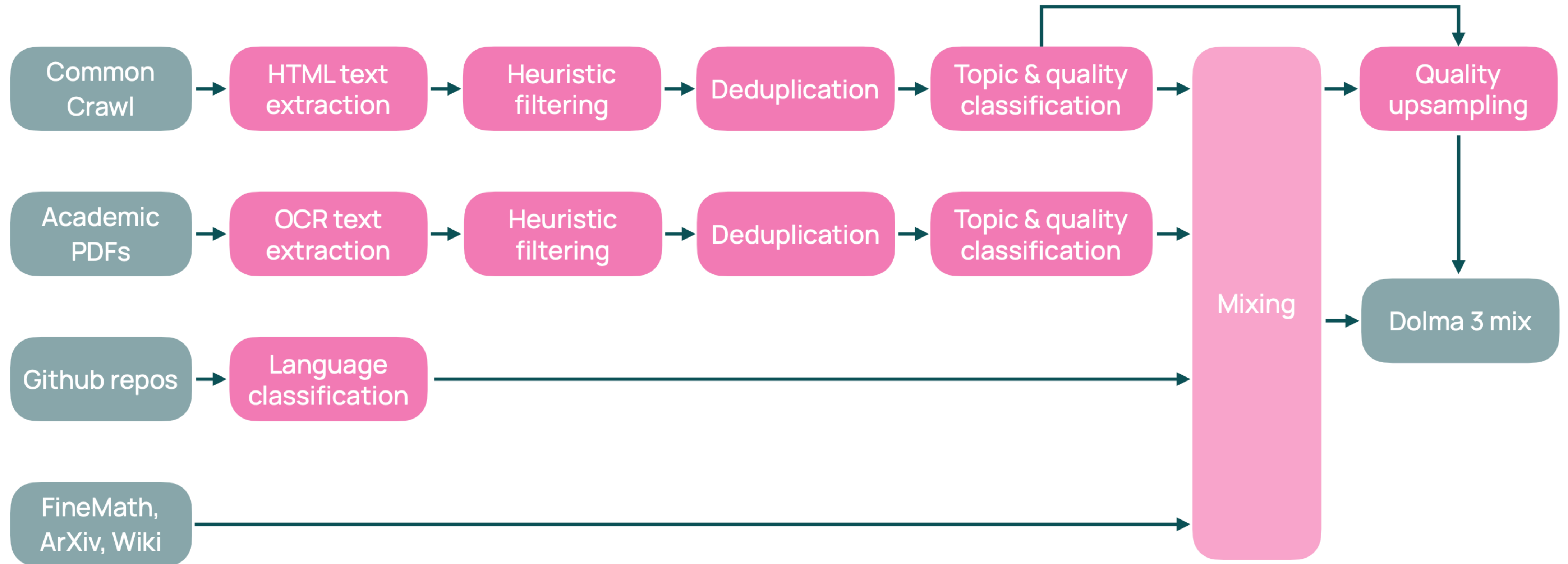


Figure 8 Data curation flow for pretraining data sources in DOLMA 3 MIX.

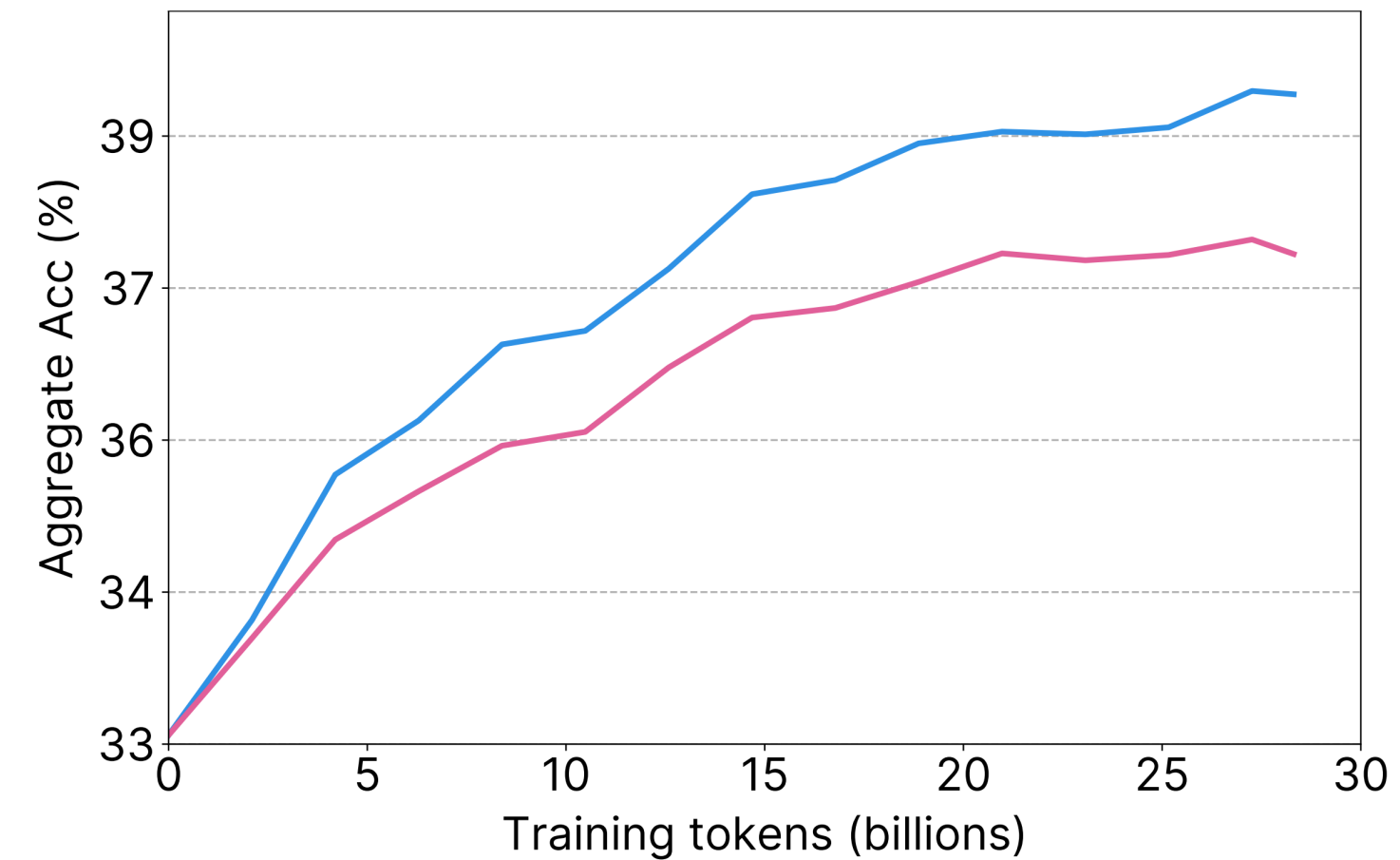
Next class

- How do we turn websites, PDFs, and other complex data formats into training data?
- Deduplication and other filtering
- Scaling laws of language models
- In-context learning

extra slides

Text Extraction

- Extraction: HTML to text
- Remove boilerplate
- Retain Latex, code, etc.



```
This paper concerns the quantity
, defined as the
length of the longest
subsequence of the numbers from
```

Image Equations

```
Suppose I have a smooth map
[ $f\colon \mathbb{R}^3$ 
 $\rightarrow S^2$ ]
. If I
identify [ $\mathbb{R}^3$ ]
with [ $U_S = S^3 - \setminus$ 
 $\{(0,0,1)\}$ ]
via
stereographic projection
```

Delimited Math

```
...
Penedo et al 2024
<math display="block">\mathrm{MA}
= \frac{f_0}{f_E} >
</math>
</semantics>
</math>
```

Special Tags

Paster et al 2023

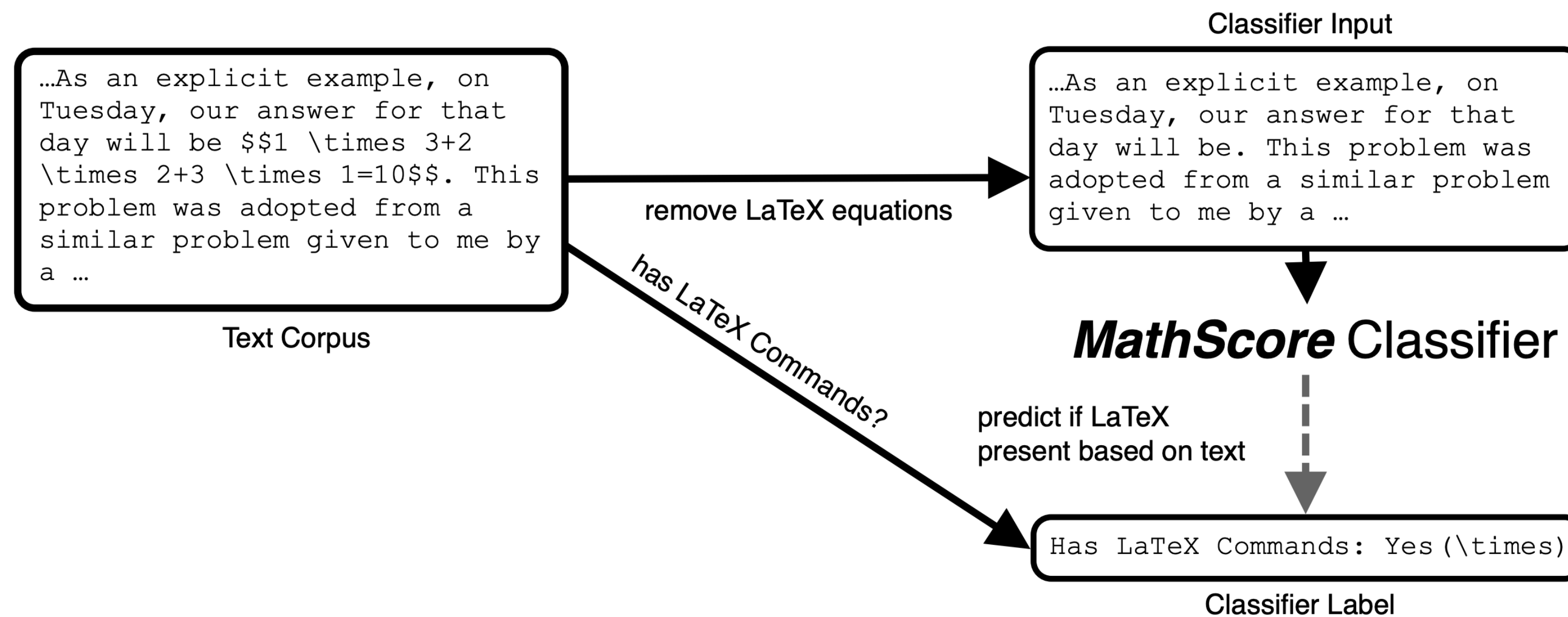
Approach: classifier filtering

- Train a classifier to detect desired data
- Use it to filter out undesired data



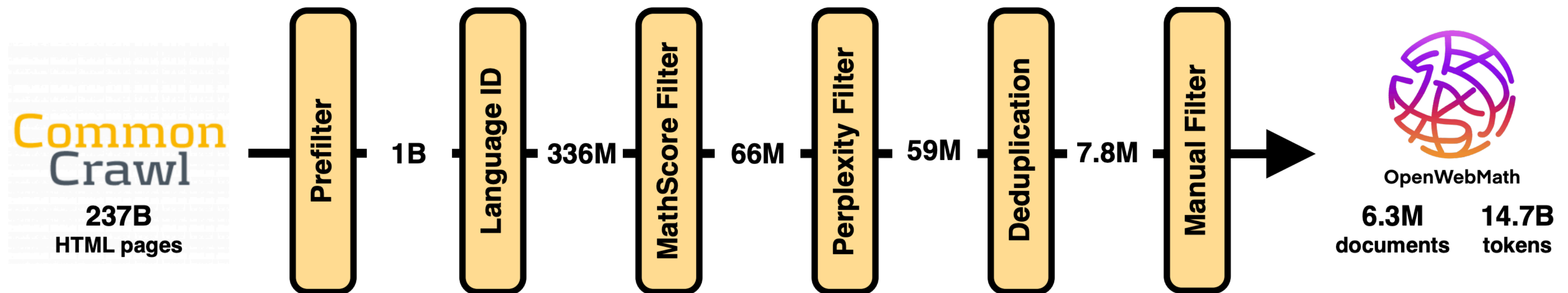
Approach: classifier filtering

- Example: **OpenWebMath** [Paster et al 2023]
- MathScore classifier detects math content



Approach: classifier filtering

- Example: **OpenWebMath** [Paster et al 2023]
- MathScore classifier detects math content



Approach: classifier filtering

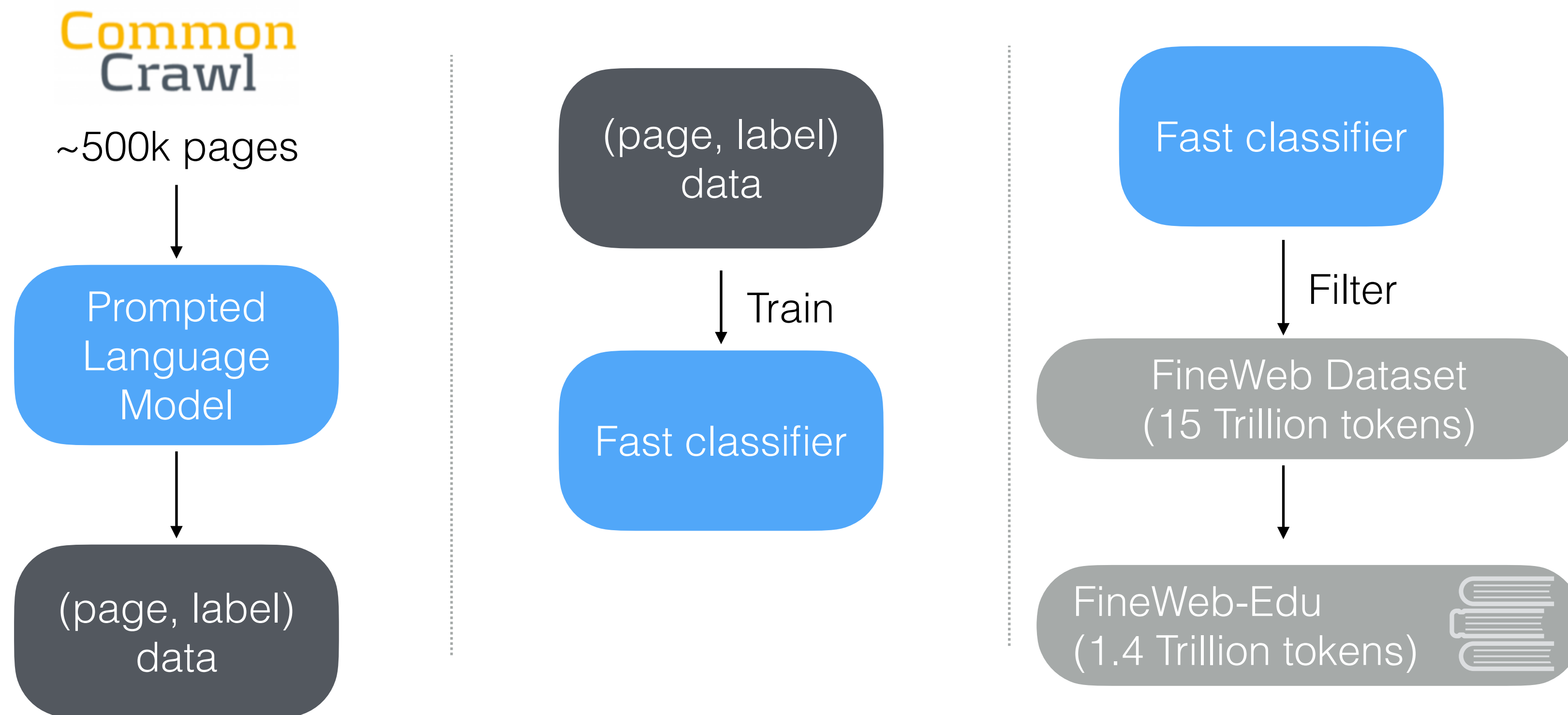
- Example: **OpenWebMath** [Paster et al 2023]

Domain	# Characters	% Characters
stackexchange.com	4,655,132,784	9.55%
nature.com	1,529,935,838	3.14%
wordpress.com	1,294,166,938	2.66%
physicsforums.com	1,160,137,919	2.38%
github.io	725,689,722	1.49%
zbmath.org	620,019,503	1.27%
wikipedia.org	618,024,754	1.27%
groundai.com	545,214,990	1.12%
blogspot.com	520,392,333	1.07%
mathoverflow.net	499,102,560	1.02%

<https://huggingface.co/datasets/open-web-math/open-web-math>

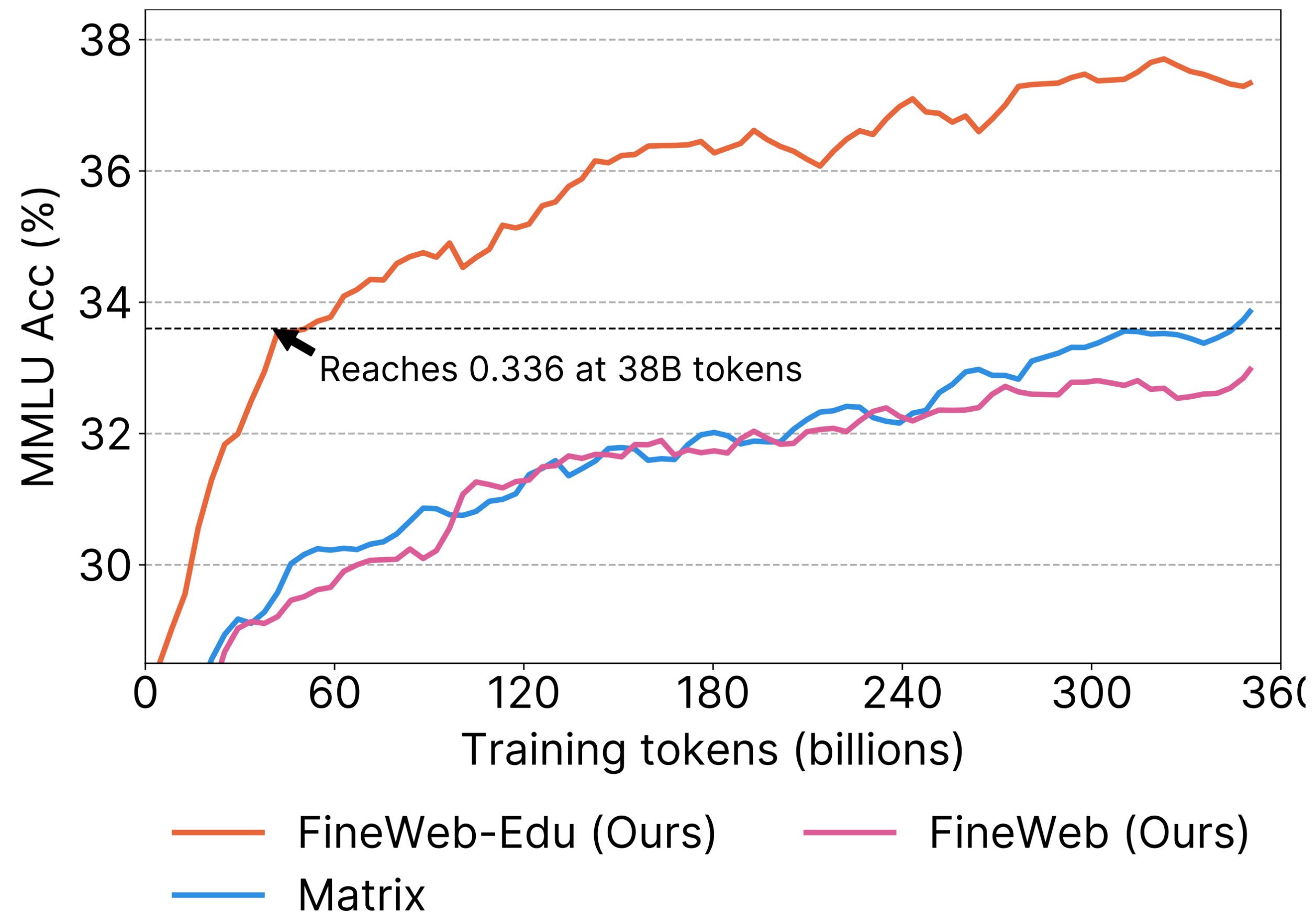
Approach: classifier filtering

- Example: **FineWeb-Edu** [Penedo et al 2024]
- Classifier to classify pages as “educational”



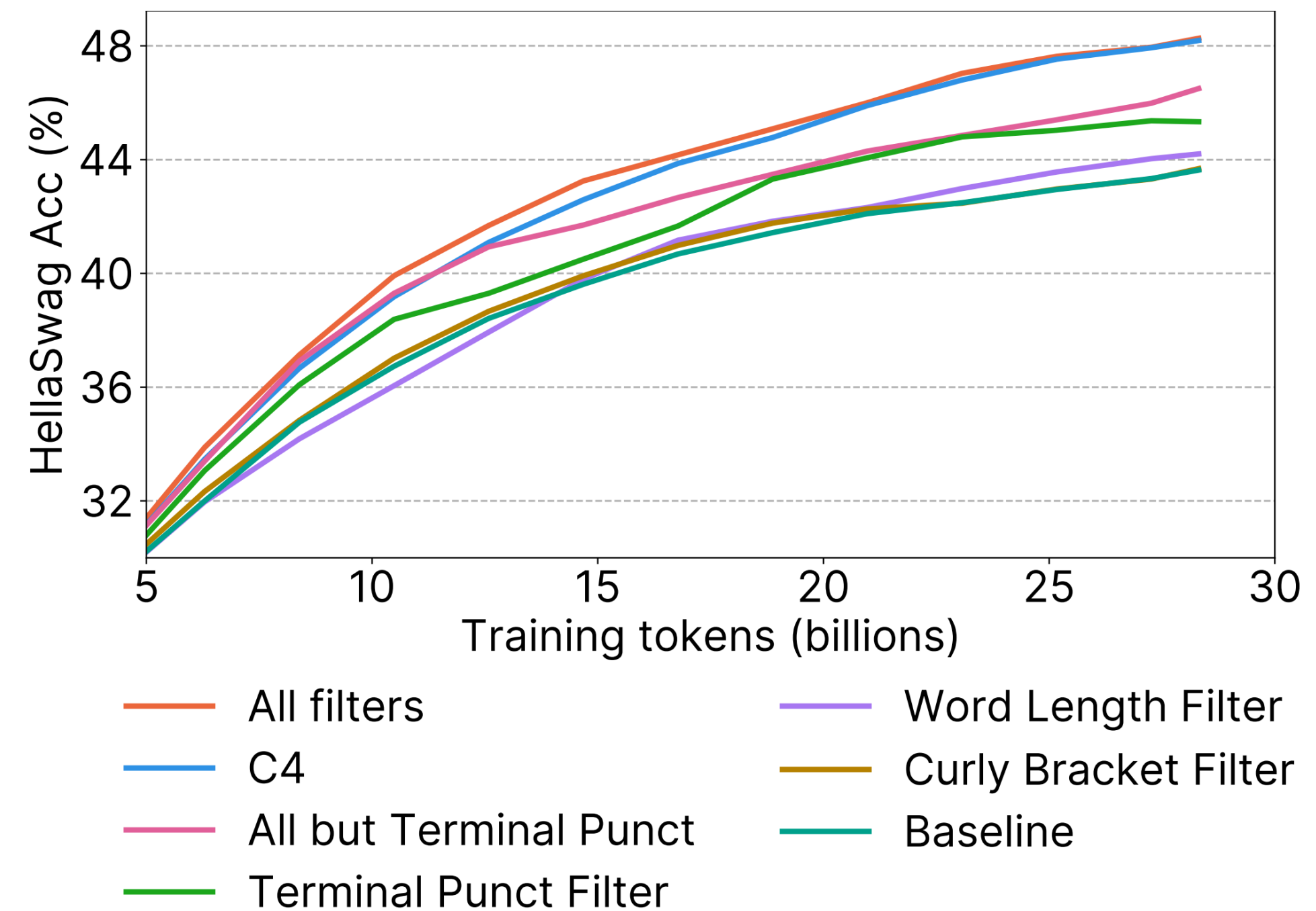
Approach: classifier filtering

- Example: **FineWeb-Edu** [Penedo et al 2024]



Quality: Filtering

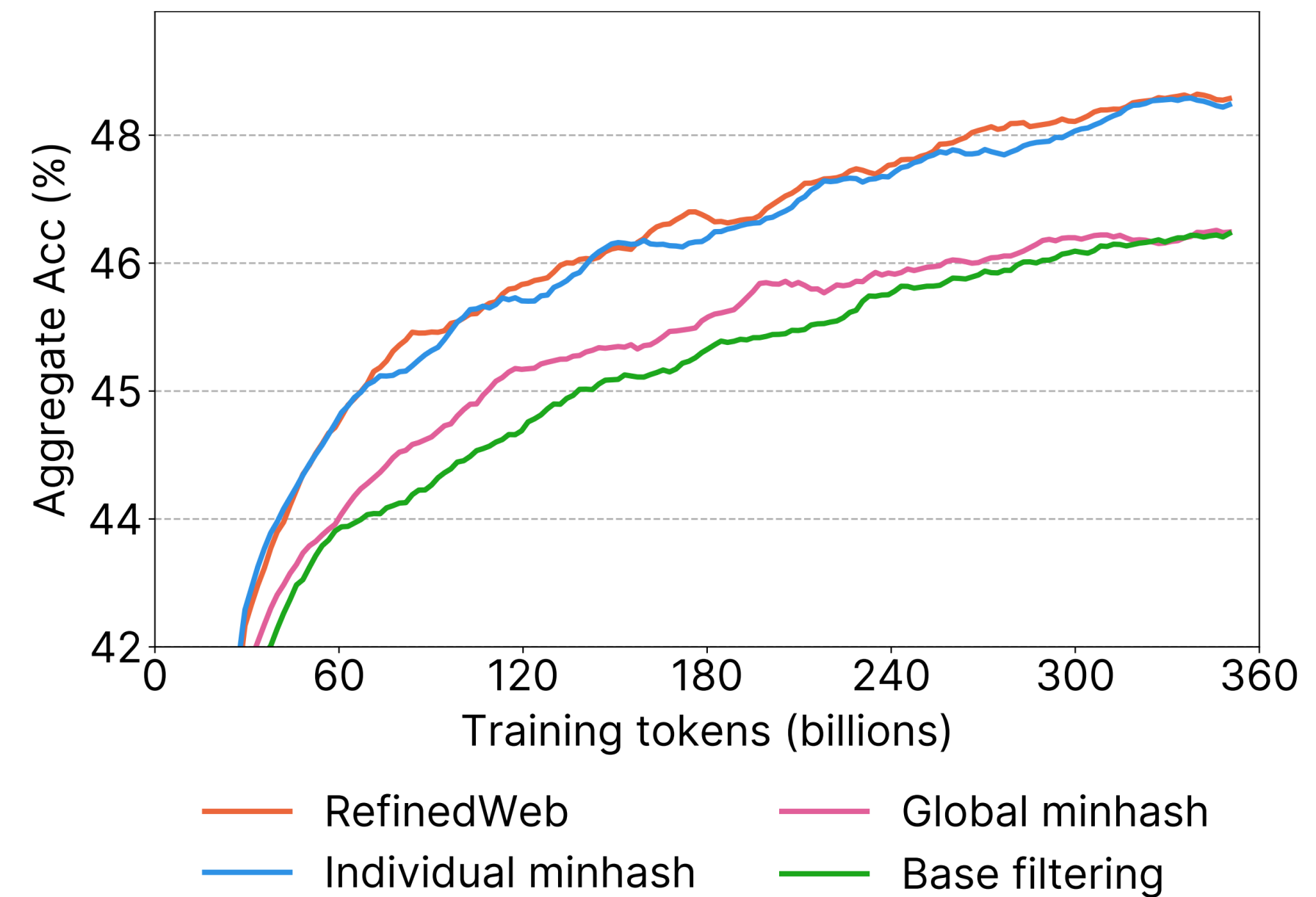
- Filter out unwanted text
- Language filter
- Repetitions
- Too many short lines
- ...



Penedo et al 2024

Quality: Deduplication

- Remove duplicate content
- Fuzzy strategy: *minhash*
- Too much deduplication can be harmful
- [Penedo et al 2024]:
Deduplicate per-snapshot
rather than globally



Penedo et al 2024

Recap

- Data factors:
 - **Quantity**: How much data do I have?
 - **Quality**: Is it beneficial for training?
 - **Coverage**: Does the data cover the domain(s) I care about, and in the right proportions?
- We mix together filtered, cleaned web data and other sources to make a pre-training dataset

end

Practical tools: HuggingFace

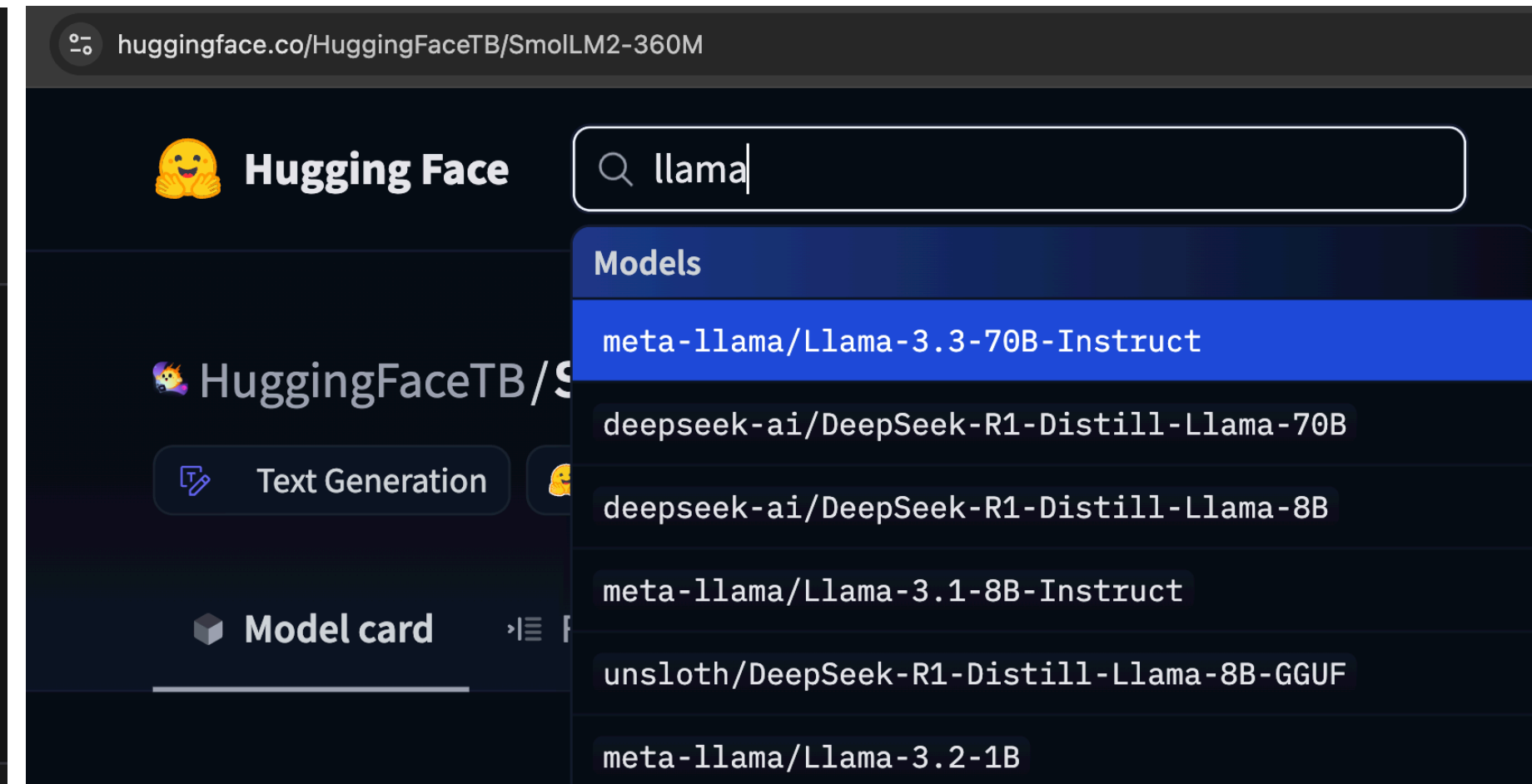
Load tokenizer and model

- Find models at <https://huggingface.co/>

```
from transformers import AutoTokenizer, AutoModelForCausalLM

model = "HuggingFaceTB/SmolLM2-360M"

tokenizer = AutoTokenizer.from_pretrained(model)
model = AutoModelForCausalLM.from_pretrained(model)
```



https://github.com/cmu-l3/anlp-spring2026-code/blob/main/06_pretraining/pretraining.ipynb

Example (Dolma)

```
added 2023-04-11T09:57:03.044571+00:00
attributes {'random_number_v1__random_number_v1__random': [[0, 9626, 0.11918]]}
created 2020-01-17T12:48:23Z
id http://250news.theexplorationplace.com/www.250news.com/65595.html
metadata {'bucket': 'head', 'cc_segment': 'crawl-data/CC-MAIN-2020-05/segments/1'}
source common-crawl
text Prince George, B.C.– Construction of the new RiverBend Seniors housing project
The $33 million dollar project was first presented to Mayor and Council in 2013
Hall and key members of the City Staff, arranged to meet with Quinn in Kamloops
“This project comes at the perfect time for us” says Gwen Norheim. She and her husband
Quinn says they did make an interesting discovery when they started construction
That’s it, big smiles Shirley and Mike... there is an election coming.
This is an excellent and well needed project!
If the NDP was in power (god forbid) and it was NDP MLA’s in the picture, you would
Go ahead and deny it if you want, but we know better!
What we do know with the liberals is they are always raising fees and medical costs
grow up galt.
```

https://github.com/cmu-l3/anlp-spring2026-code/blob/main/06_pretraining/pretraining.ipynb

Thank you