CS11-711 Advanced NLP

Pretraining

Sean Welleck

https://cmu-l3.github.io/anlp-fall2025/

https://github.com/cmu-l3/anlp-fall2025-code

Slides adapted from Graham Neubig’s Fall 2024 course

Recap

- Classification, language modeling, sequence architectures

- So far:

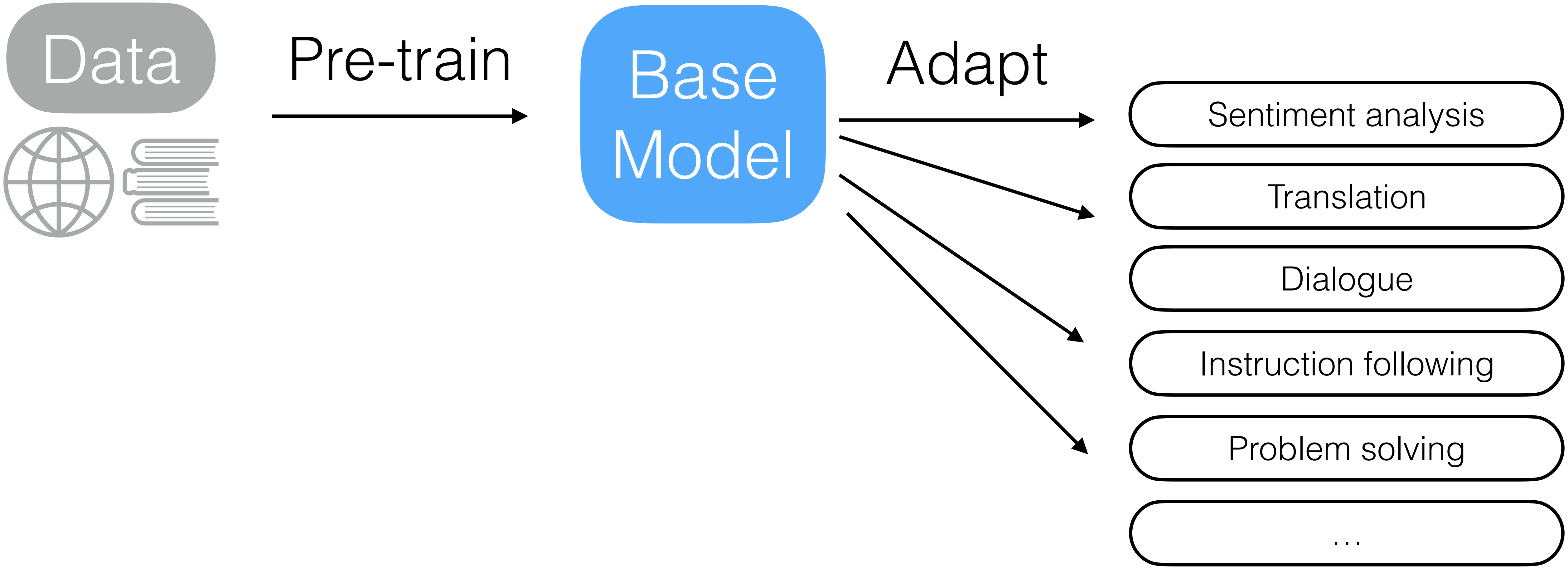
  - Train from scratch

  - 1 model, 1 task

- Today:

  - *Pretrain* a *single model*, adapt it to *many tasks*

Basic idea



Data

Pre-train

Base Model

Adapt

Sentiment analysis

Translation

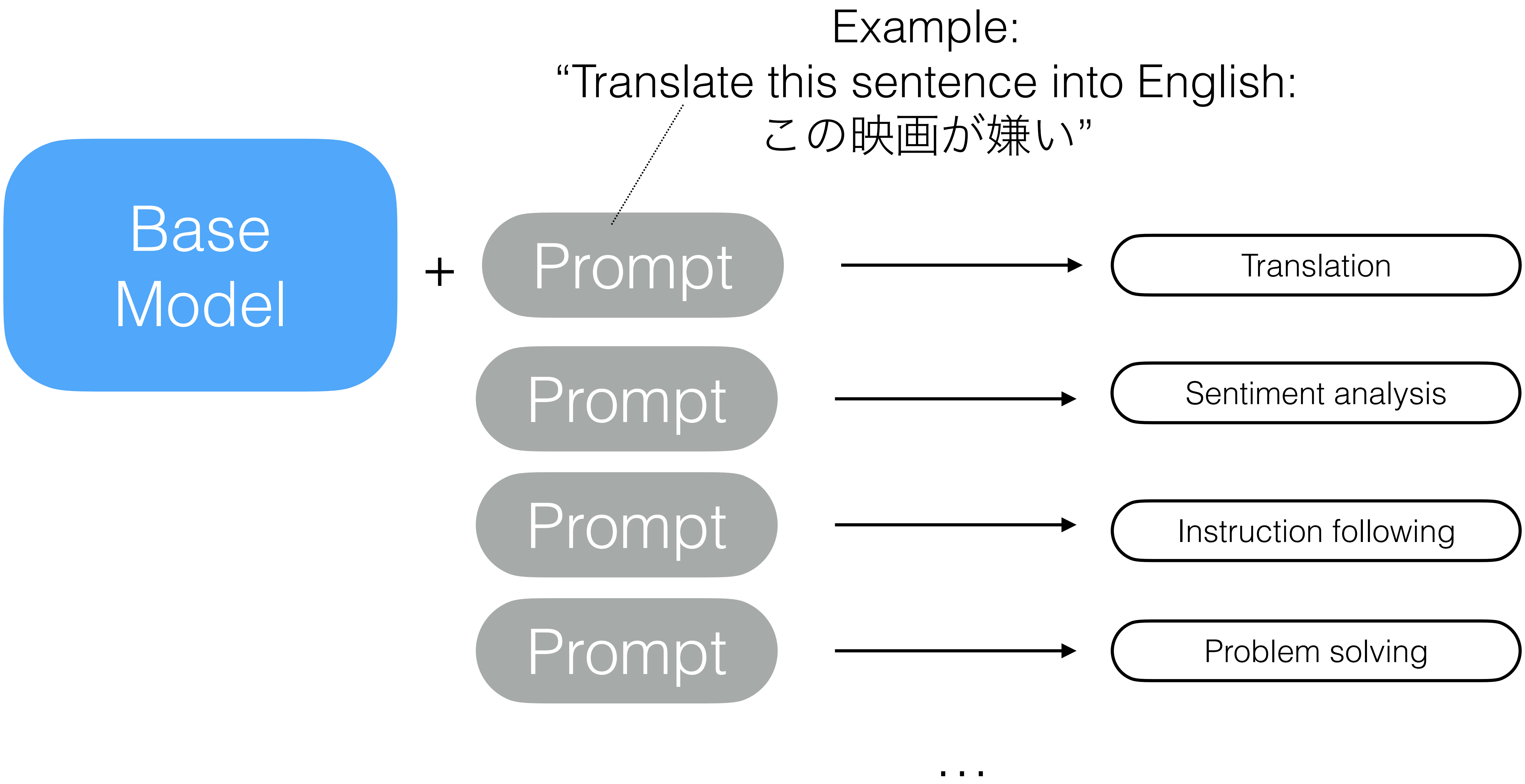
Dialogue

Instruction following

Problem solving

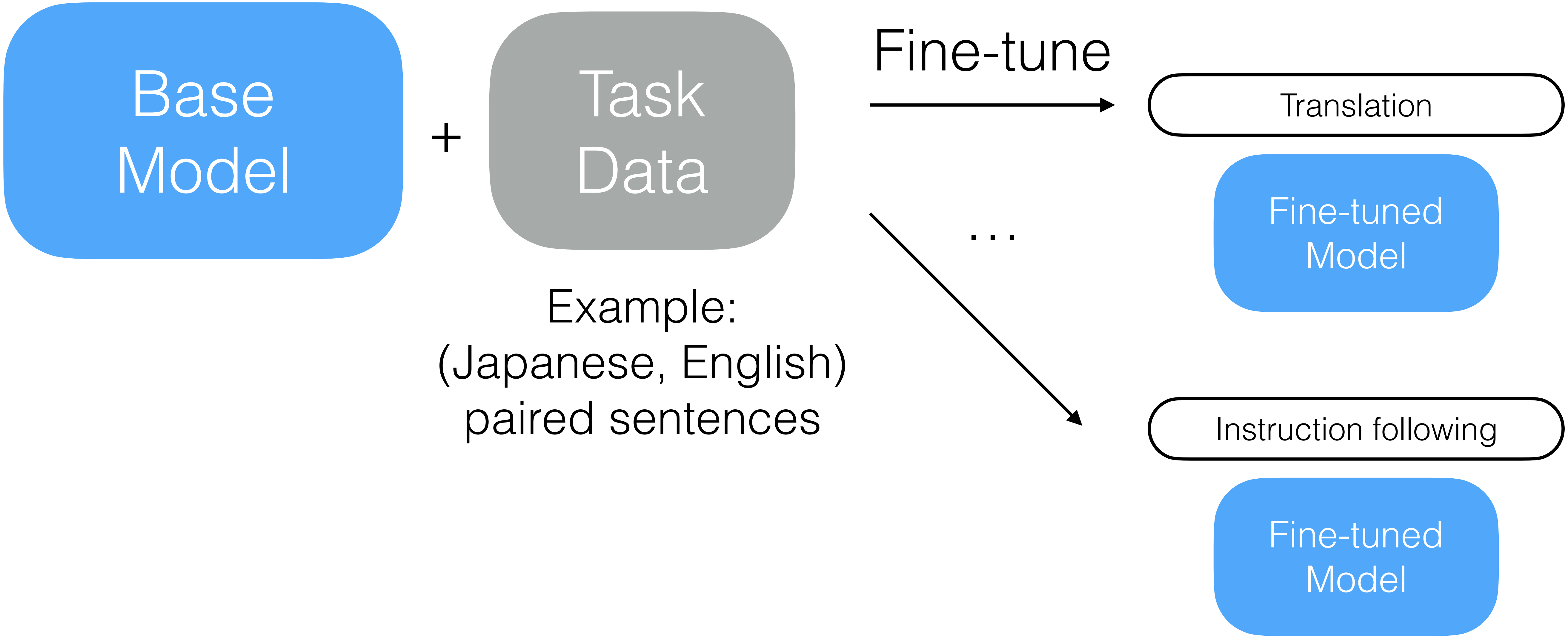
...

Adaptation: prompting [Lecture 7]



Example:
“Translate this sentence into English:
この映画が嫌い”

Adaptation: fine-tune [Lecture 8]



Base Model

+

Task Data

Example: (Japanese, English) paired sentences

Fine-tune

Translation

Fine-tuned Model

…

Instruction following

Fine-tuned Model

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, …

Major factors

- Pre-trained models have names like BERT, GPT-3, Llama, Deepseek-v3, …

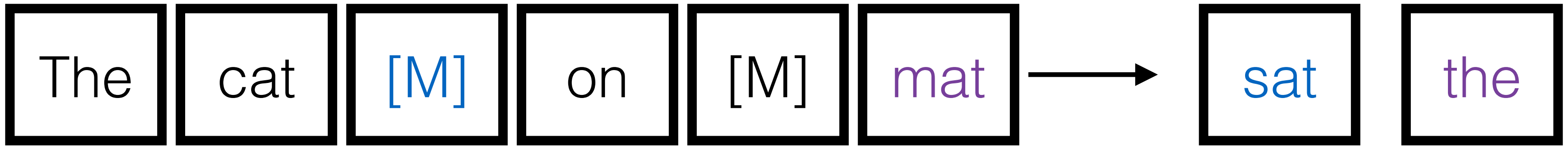
- Each model is influenced by 4 major factors:
  - **Architecture:** neural network architecture
  - **Task:** what the model predicts (e.g. next-token)
  - **Data:** the data used to train the model
  - **Hyper-parameters**: e.g. learning rate, batch size

Today's lecture

- Tasks
  - Masked language modeling objective
  - Autoregressive language modeling objective
- Data: sources, quality, and quantity
- Thinking about pretraining
  - Tokens, model size, compute
  - Scaling laws

Masked Language Modeling

- Predict masked tokens x_M given visible tokens $x_{\neg M}$



The cat [M] on [M] mat → sat the

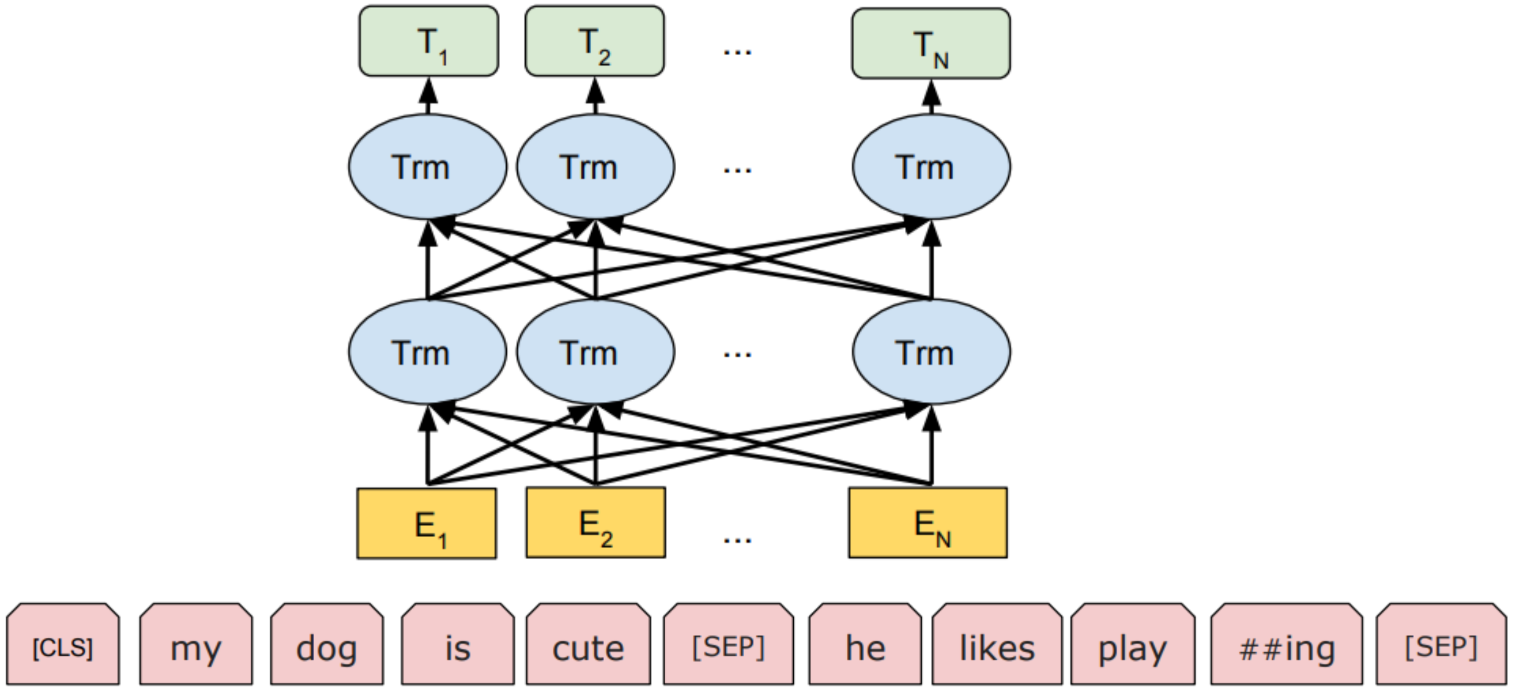
$$\mathcal{L}_{\mathrm{MLM}}(\theta; D) = -\frac{1}{|D|}\sum_{x\in D}\mathbb{E}_{M\sim\mathrm{corrupt}(x)}\sum_{t\in M}\log p_{\theta}(x_t \mid x_{\neg M})$$

- View as *denoising*: corrupt x → reconstruct x
- Maximizes *pseudo-likelihood*

Example: BERT

(Devlin et al. 2018)

- **Model:** Transformer



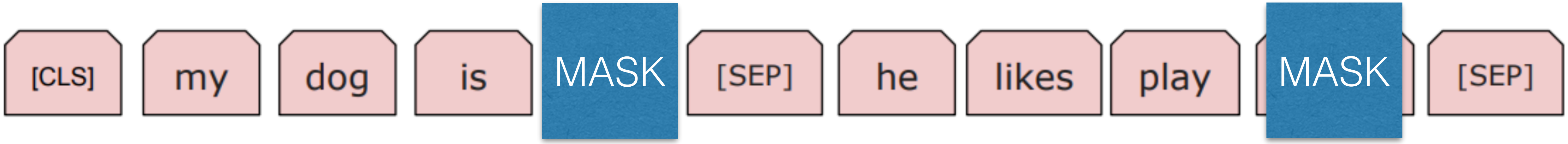
- **Data:** BooksCorpus + English Wikipedia
- **Task:** Masked language modeling

Example: BERT

(Devlin et al. 2018)

Predict a masked token

- 80%: substitute input token with [MASK]
- 10%: substitute input token with random token
- 10%: no change



Adapting a masked language model

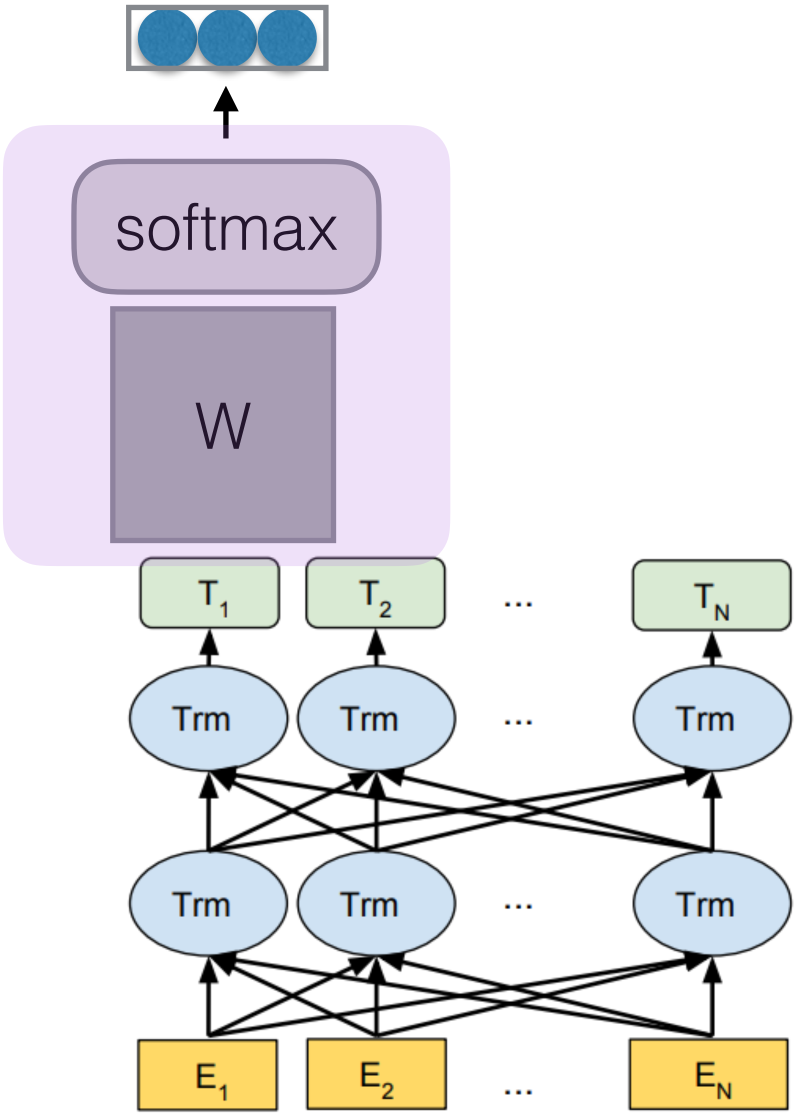
- Add an output layer that maps a hidden vector to scores

- Fine-tune the weights (either just W, or all weights). Example:

    - Data: (movie review, {positive, neutral, negative})

    - Initialize the model with BERT

    - Minimize cross-entropy loss with gradient-based optimization

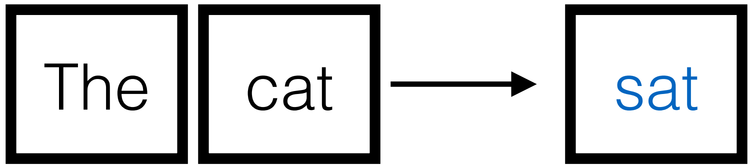


Today’s lecture

- Tasks
  - Masked language modeling
  - **Autoregressive language modeling**

Autoregressive language modeling

- Predict next token x_t given previous tokens $x_{<t}$



The cat → sat

$$\mathscr{L}_{\text{MLE}}(\theta; D) = -\frac{1}{|D|}\sum_{x\in D}\sum_{t=1}^{|x|}\log p_\theta(x_t\,|\,x_{<t})$$

- Maximizes likelihood
  - Fits a data distribution p_*
  - Learns to *compress* data generated by p_*

Maximum likelihood: fits a data distribution

- Makes p_θ match the data distribution p_{data} (p_* for brevity)

$$\begin{aligned}
\min_\theta D_{KL}(p_* || p_\theta) &= \min_\theta - \sum_{x\in\mathcal{X}} p_*(x)\log\frac{p_\theta(x)}{p_*(x)} \\
&\equiv \min_\theta - \sum_{x\in\mathcal{X}} p_*(x)\log p_\theta(x) + \text{const} \\
&= \min_\theta - \mathbb{E}_{x\sim p_*}\log p_\theta(x) \\
&\approx \min_\theta - \frac{1}{|D|}\sum_{x\in D}\log p_\theta(x) \\
&\equiv \max_\theta \sum_{x\in D}\log p_\theta(x)
\end{aligned}$$

Dataset:
samples from p_*

Maximum
likelihood!

Maximum likelihood: learns to compress

- Goal: compress data from a distribution p_* into a binary code, $c(x_{1:n}) \rightarrow \{0,1\}^*$

- *Arithmetic coding* turns a distribution p into a code $c(\,\cdot\,)$

- Minimum expected code length is the entropy [Shannon 1948]:

$$H(p_*) = \mathbb{E}_{x\sim p_*}[-\sum_{i=1}^{N}\log_2 p_*(x_i \,|\, x_{<i})]$$

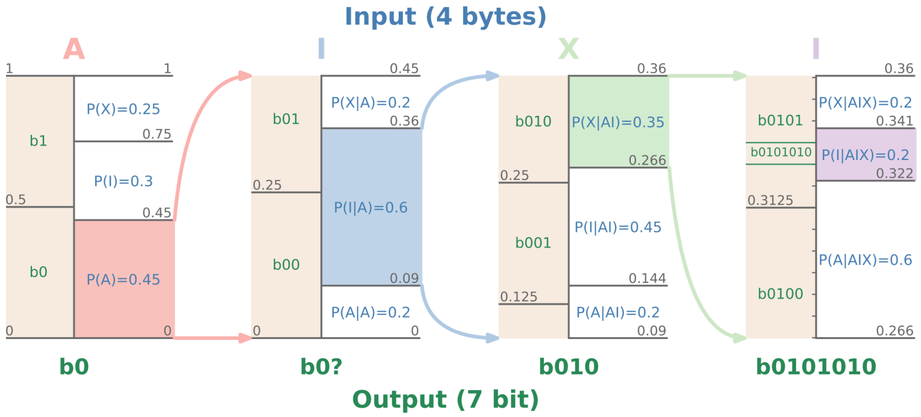


Figure: [Deletang et al 2024]

- When we use a model p_θ, the expected code length is the cross-entropy:

$$H(p_*, p_\theta) = \mathbb{E}_{x\sim p}[-\sum_{i=1}^{N}\log_2 p_\theta(x_i \,|\, x_{<i})]$$

$$H(p_*, p_\theta) = H(p_*) + KL(p_* \| p_\theta)$$

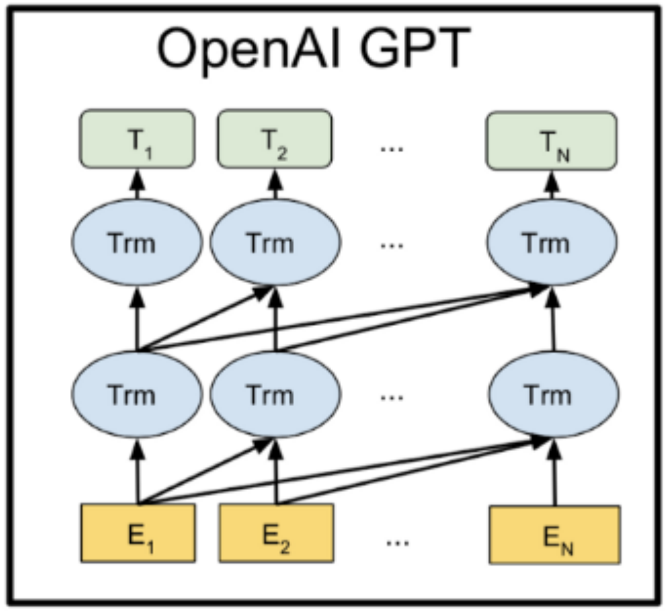
To achieve the minimum expected code length $H(p_*)$, minimize KL divergence via MLE

Key factors

$$\mathscr{L}_{\mathrm{MLE}}(\theta; D) = -\mathbb{E}_{x \sim D} \sum_{t=1}^{|x|} \log p_\theta(x_t \mid x_{<t})$$

- Things we can change:
  - θ: model architecture and size
  - D: training data
  - Optimization hyper-parameters, e.g. learning rate, batch size

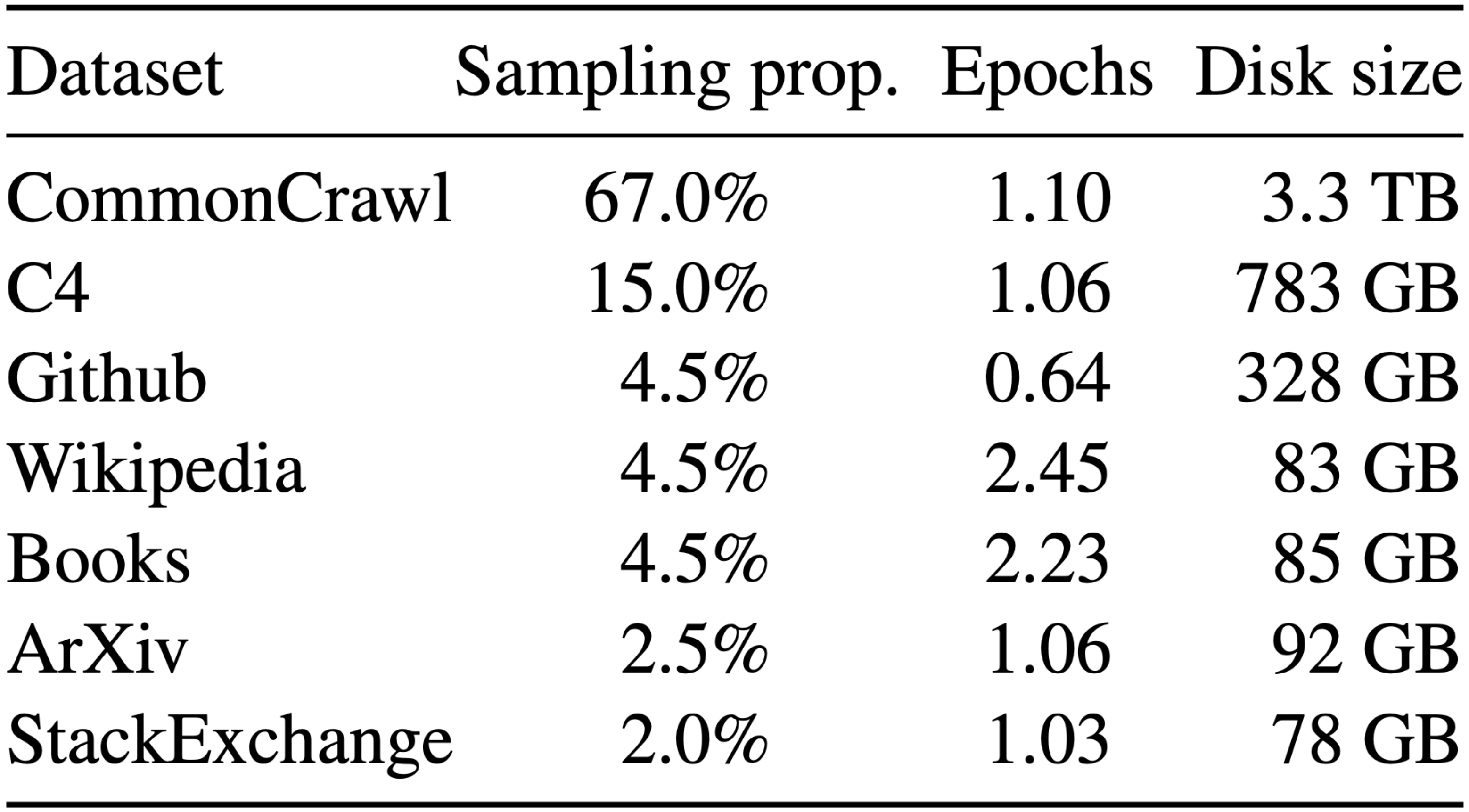
Example: GPT-2



- **Model:** Transformer (1.5B)
- **Data:** WebText (millions of web pages)

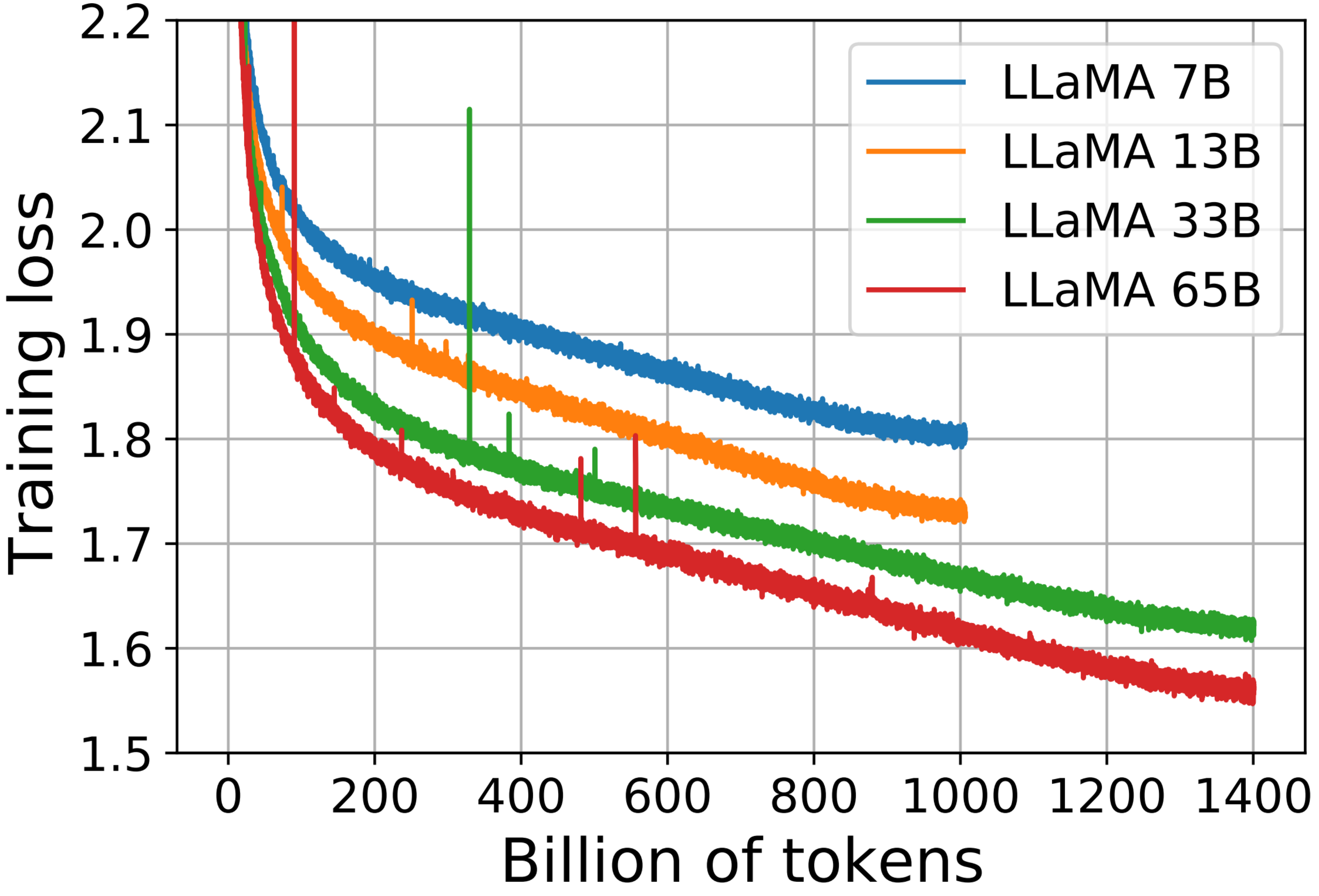
Example: Llama

- **Model:** Transformer, {6.7B, 13B, 32B, 65B}
- **Data:** 1.4 trillion tokens, sources:



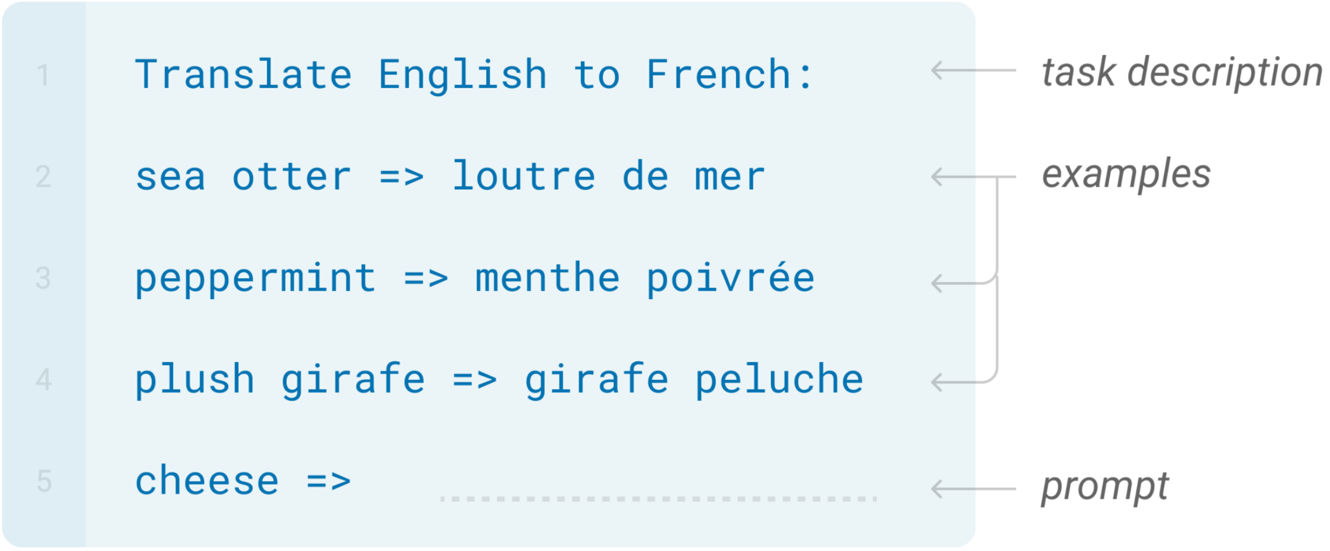
Dataset	Sampling prop.	Epochs	Disk size
CommonCrawl	67.0%	1.10	3.3 TB
C4	15.0%	1.06	783 GB
Github	4.5%	0.64	328 GB
Wikipedia	4.5%	2.45	83 GB
Books	4.5%	2.23	85 GB
ArXiv	2.5%	1.06	92 GB
StackExchange	2.0%	1.03	78 GB

Llama: training loss



Evaluating a model

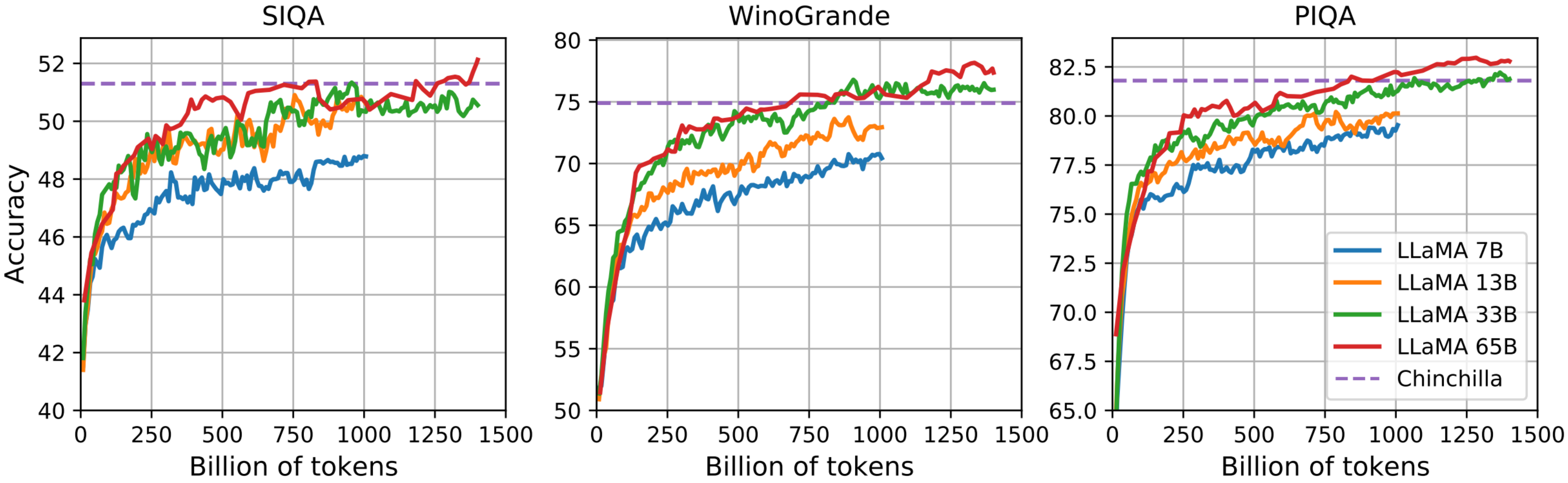
- Loss (training, validation, test)
  - Diagnose training trajectory, compare models in the same family
- Few-shot prompting



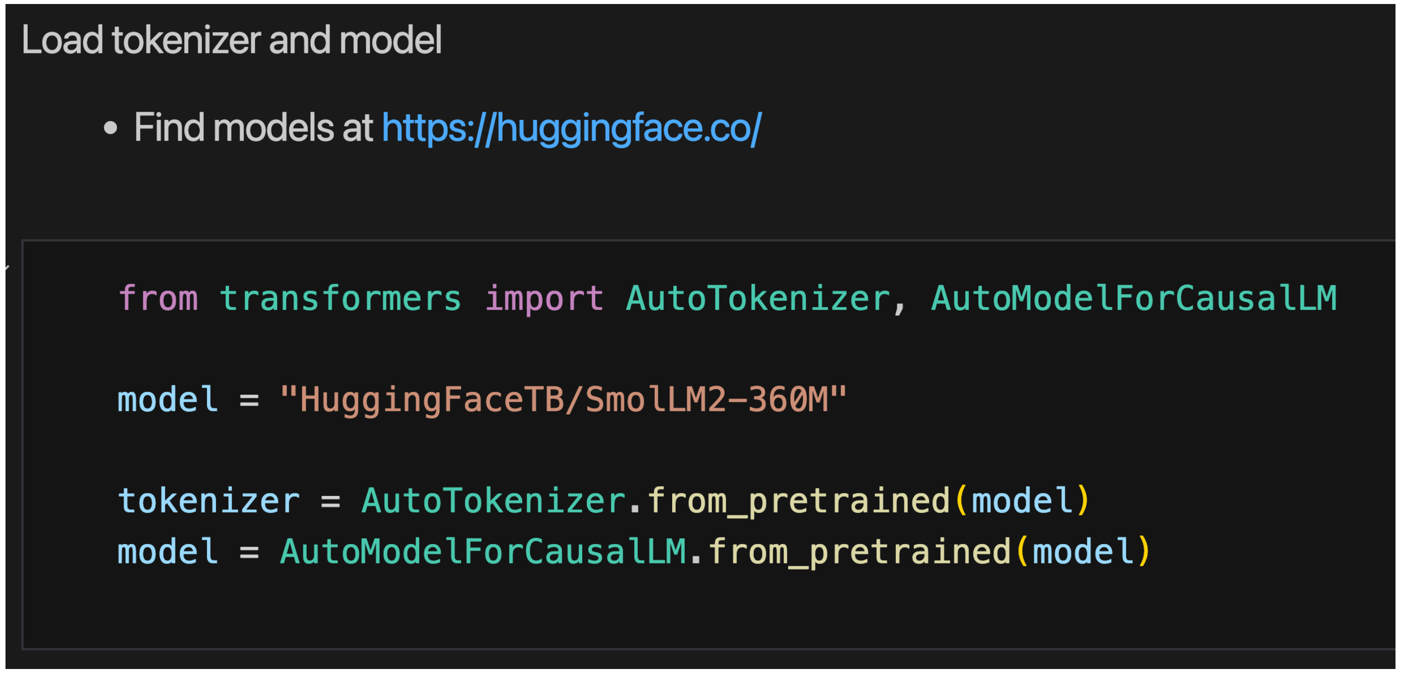
```
Translate English to French:
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese => ..............................
```

- Fine-tuning

Llama: few-shot performance trajectory



Practical tools: HuggingFace



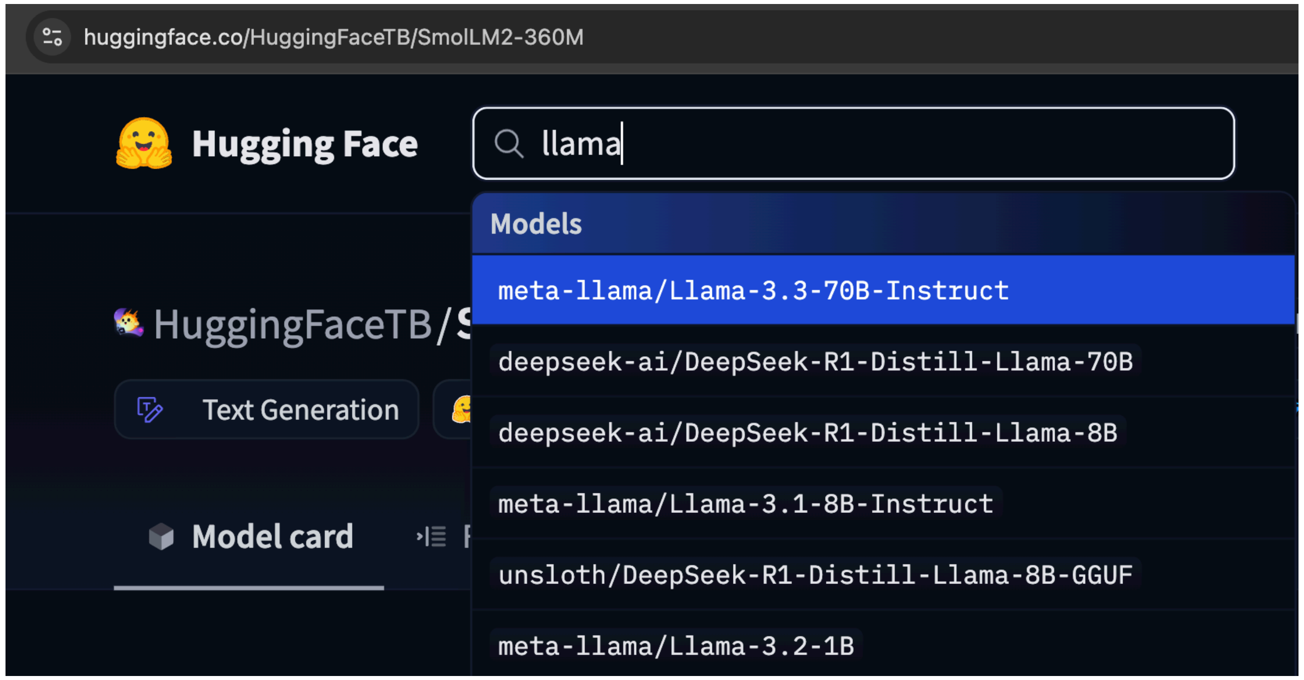
Load tokenizer and model

- 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)
```



huggingface.co/HuggingFaceTB/SmolLM2-360M

llama

Models

meta-llama/Llama-3.3-70B-Instruct

deepseek-ai/DeepSeek-R1-Distill-Llama-70B

deepseek-ai/DeepSeek-R1-Distill-Llama-8B

meta-llama/Llama-3.1-8B-Instruct

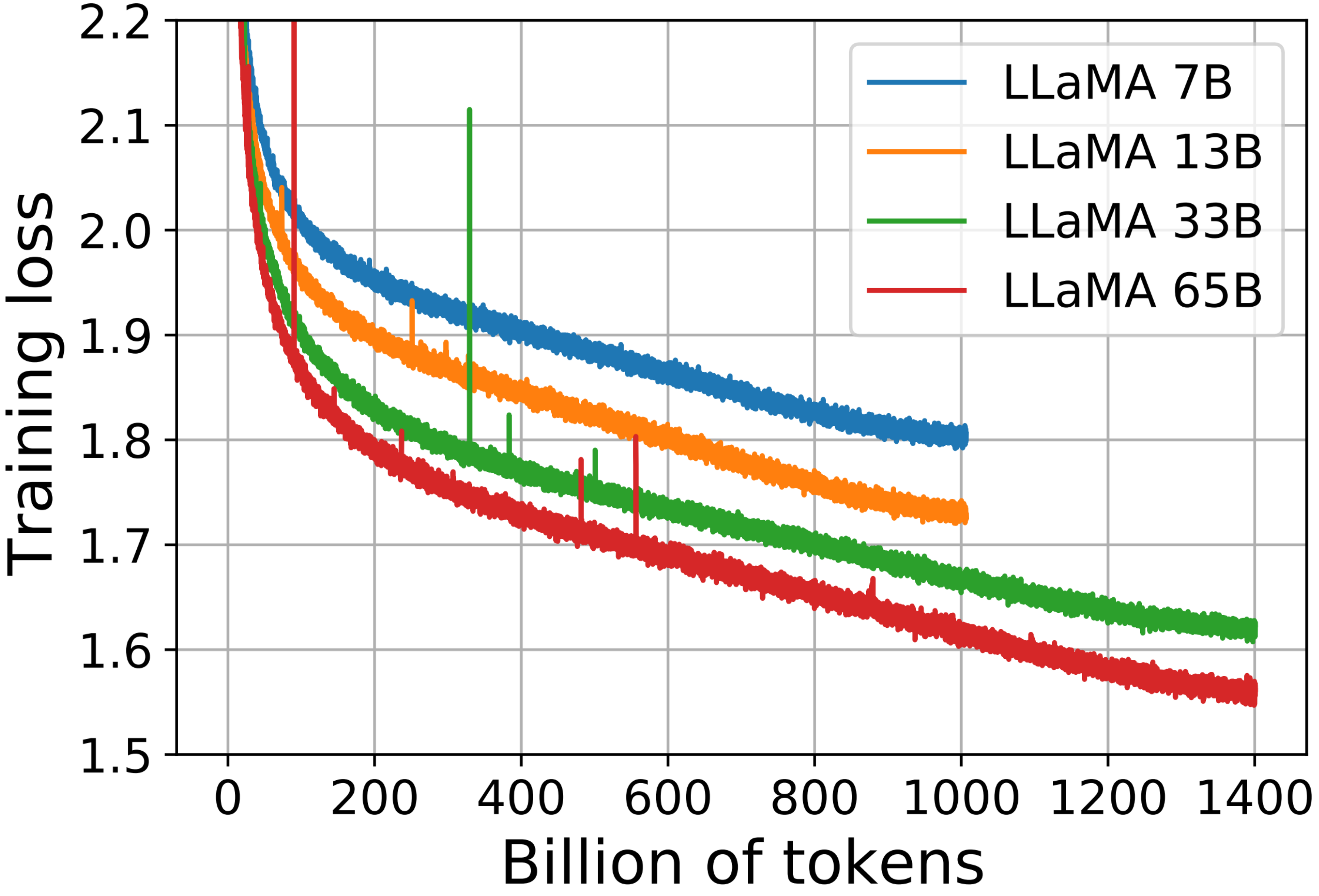
unsloth/DeepSeek-R1-Distill-Llama-8B-GGUF

meta-llama/Llama-3.2-1B

https://github.com/cmu-l3/anlp-fall2025-code/blob/main/06_pretraining/pretraining.ipynb

Today’s lecture

- Tasks
- **Data**: sources, quality, and quantity



Better Loss

More data

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?

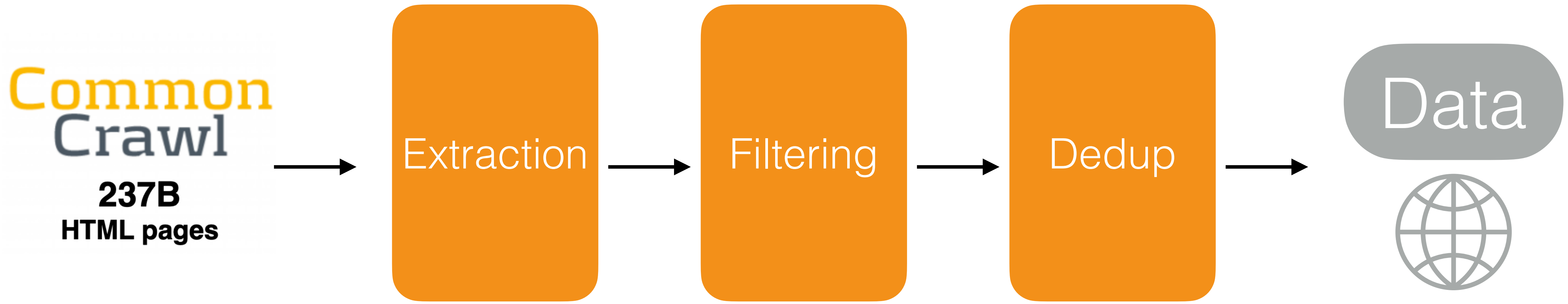
Data quantities

	Tokens of training data
Llama 1	1.4 trillion
Llama 2	1.8 trillion
Llama 3	15 trillion
Deepseek 3	15 trillion

Wikipedia: < 10 billion

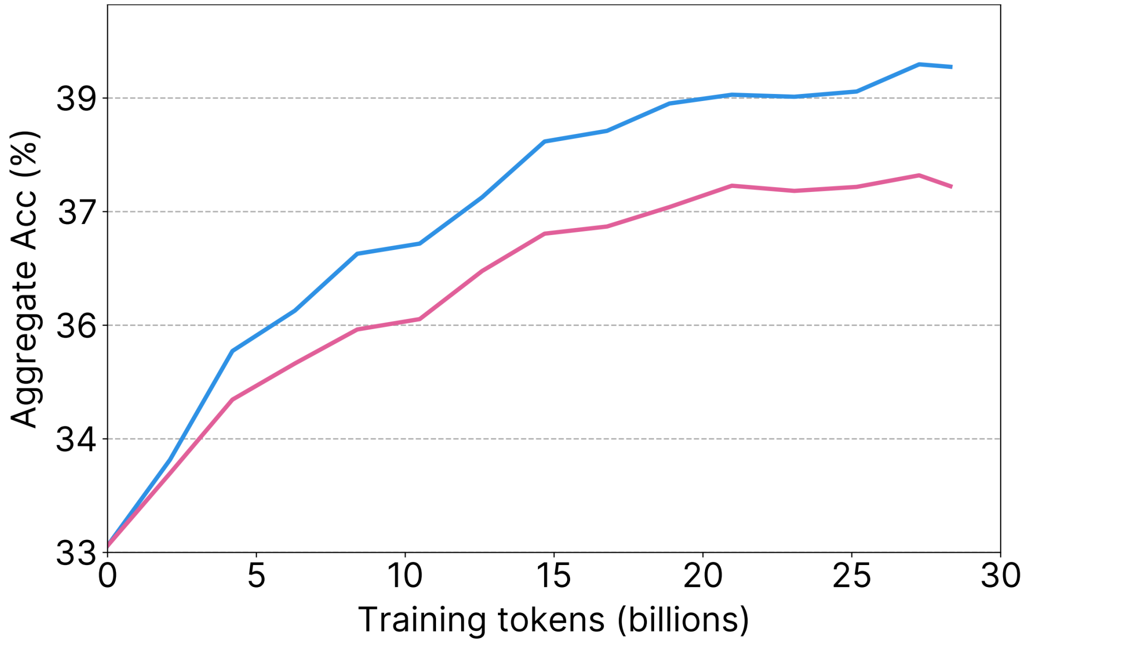
Web data: common crawl

- Large snapshots of web pages.
  - Extraction: HTML to text
  - Filtering: filter out unwanted pages
  - Deduplication: many duplicate web pages

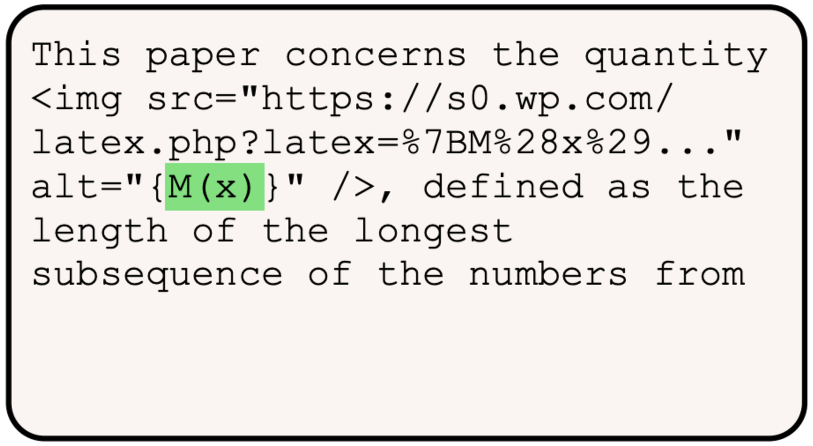


Quality: Extraction

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

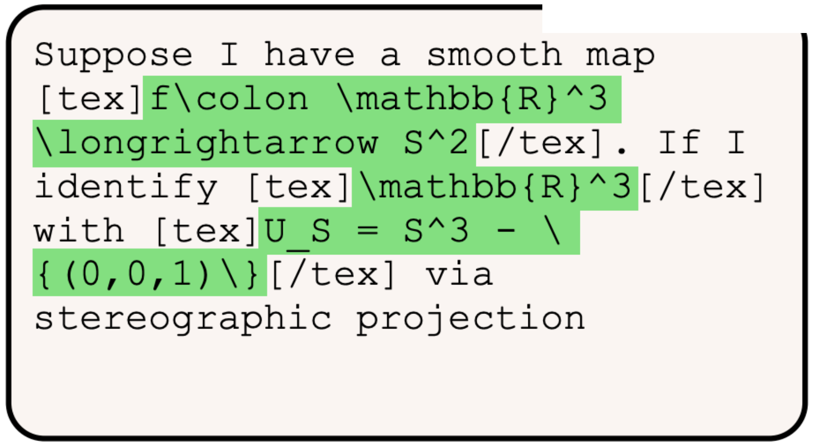


Penedo et al 2024



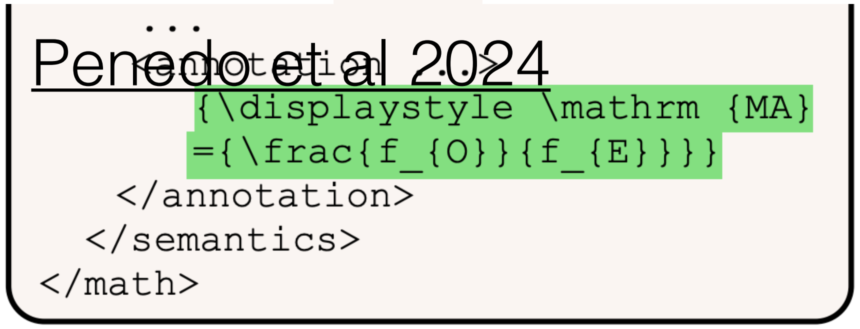
```
This paper concerns the quantity
<img src="https://s0.wp.com/
latex.php?latex=%7BM%28x%29..."
alt="{M(x)}" />, defined as the
length of the longest
subsequence of the numbers from
```

Image Equations



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

Delimited Math



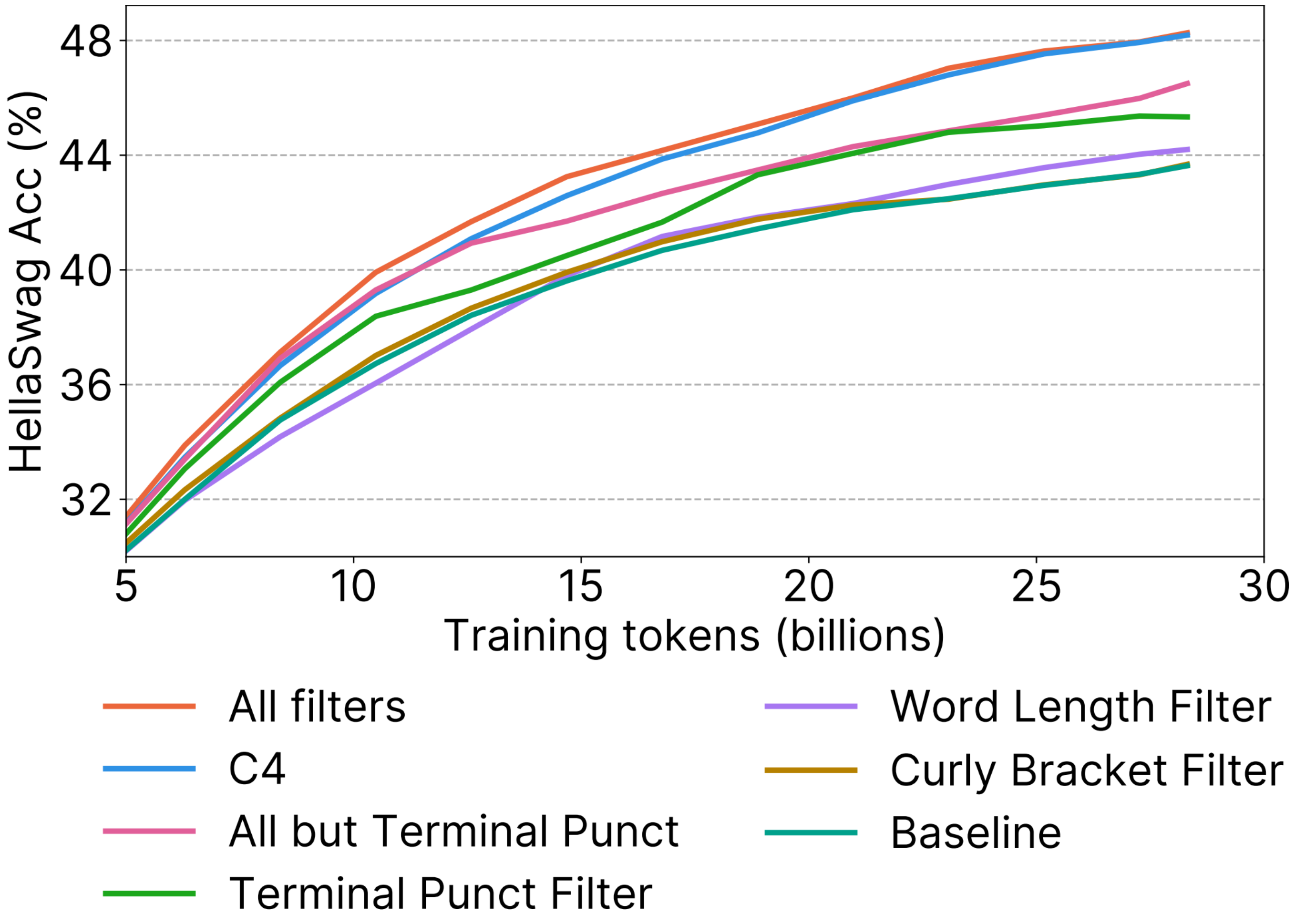
```
      ...
     <annotation
          {\displaystyle \mathrm {MA}
          ={\frac{f_{O}}{f_{E}}}}
     </annotation>
   </semantics>
</math>
```

Special Tags

Paster et al 2023

Quality: Filtering

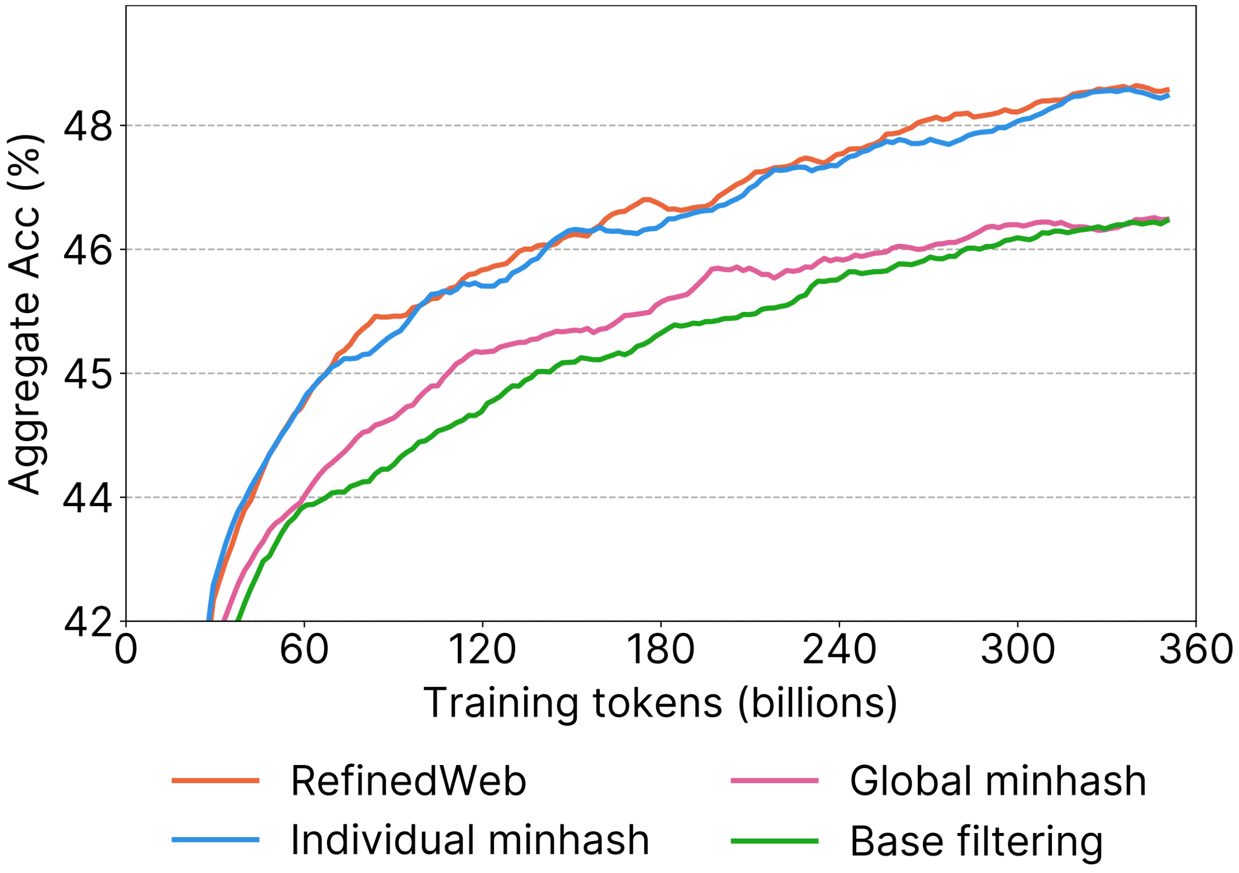
- Filter out unwanted text
  - Language filter
  - Repetitions
  - Too many short lines
  - …



Penedo et al 2024

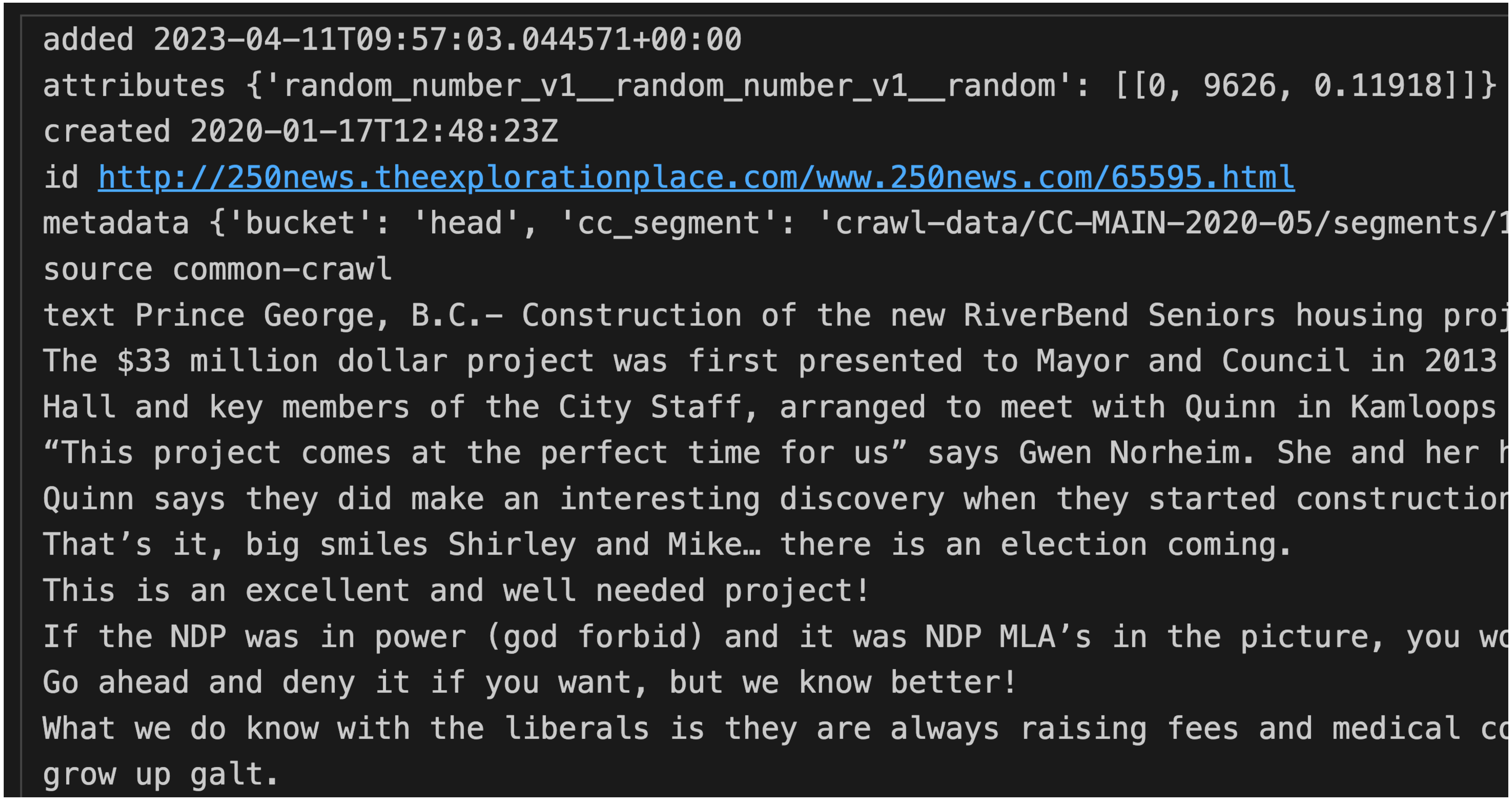
Quality: Deduplication

- 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

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 proj
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 h
Quinn says they did make an interesting discovery when they started constructior
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 wo
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 co
grow up galt.
```

https://github.com/cmu-l3/anlp-fall2025-code/blob/main/06_pretraining/pretraining.ipynb

Data factors

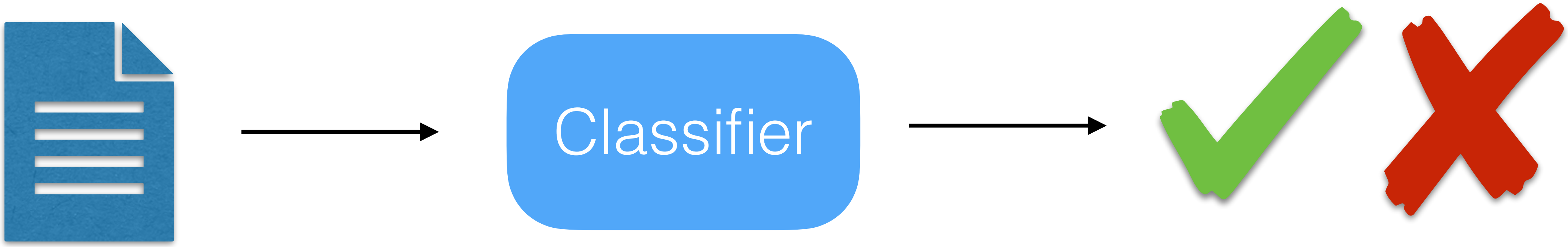
- Quantity: How much data do I have?
- Quality: Is it beneficial for training?
- **Coverage**: Does the data cover the domain(s) I care about, and in the right proportions?

Coverage

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

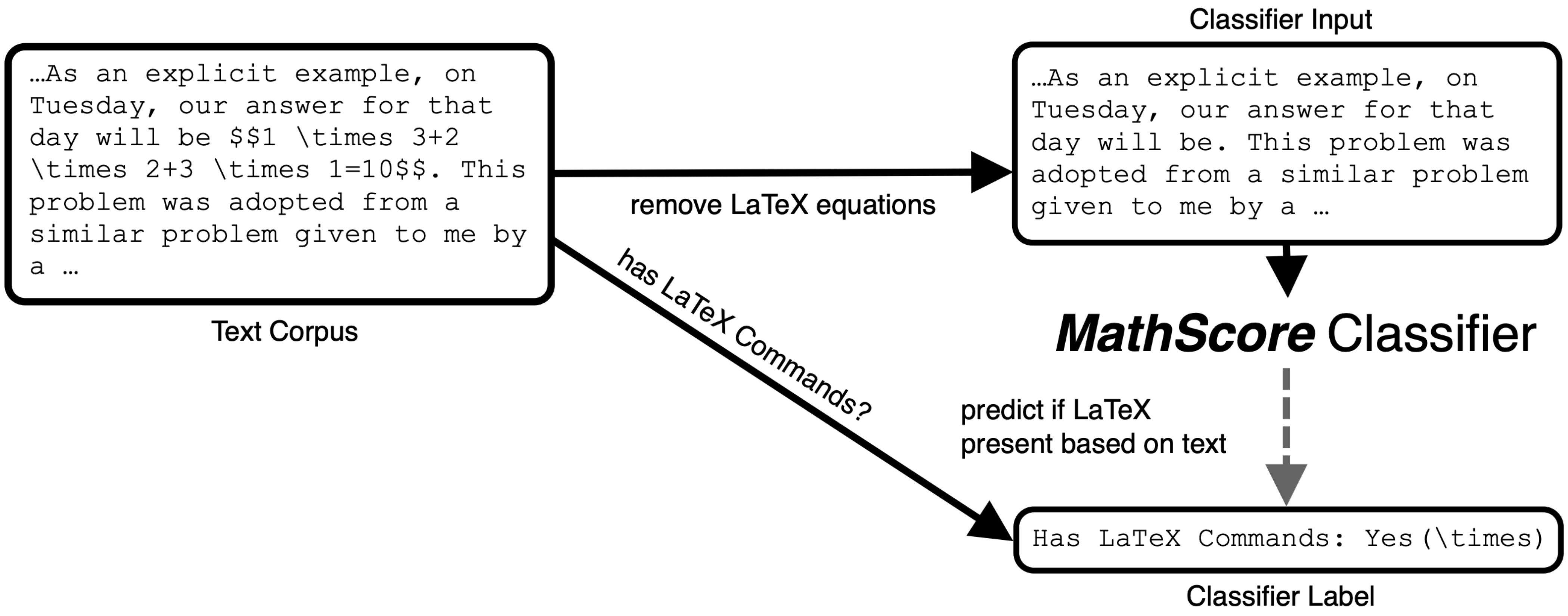
Approach: classifier filtering

- 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



…As an explicit example, on Tuesday, our answer for that day will be $$1 \times 3+2 \times 2+3 \times 1=10$$. This problem was adopted from a similar problem given to me by a …

Text Corpus

remove LaTeX equations

Classifier Input

…As an explicit example, on Tuesday, our answer for that day will be. This problem was adopted from a similar problem given to me by a …

MathScore Classifier

has LaTeX Commands?

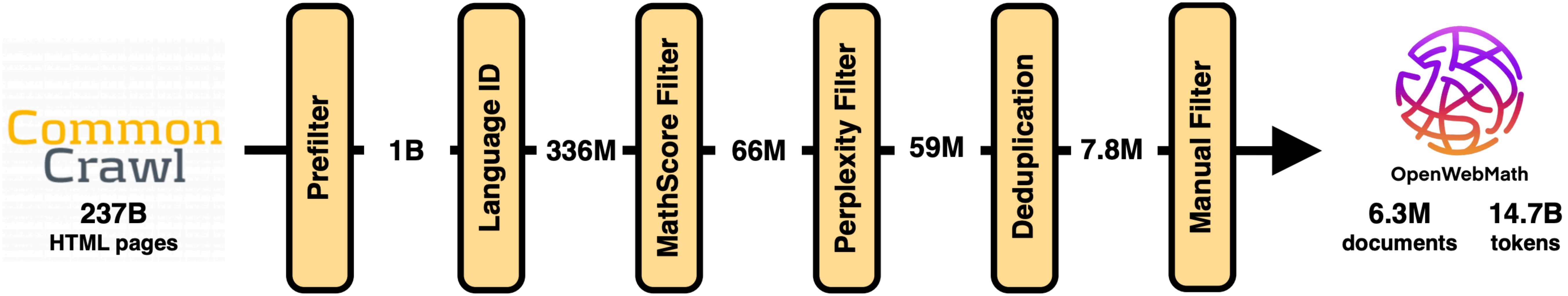
predict if LaTeX present based on text

Has LaTeX Commands: Yes (\times)

Classifier Label

Approach: classifier filtering

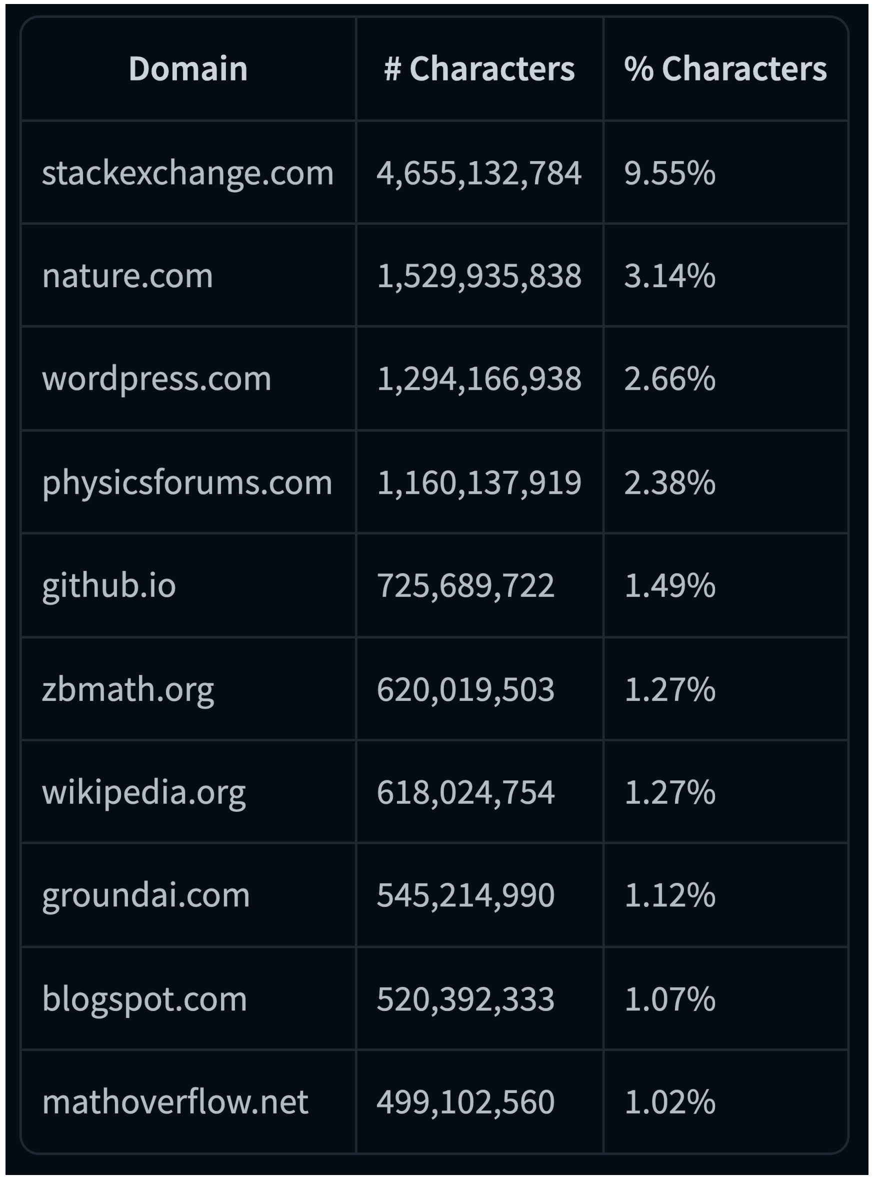
- Example: **OpenWebMath** [Paster et al 2023]
    - MathScore classifier detects math content



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

Approach: classifier filtering

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

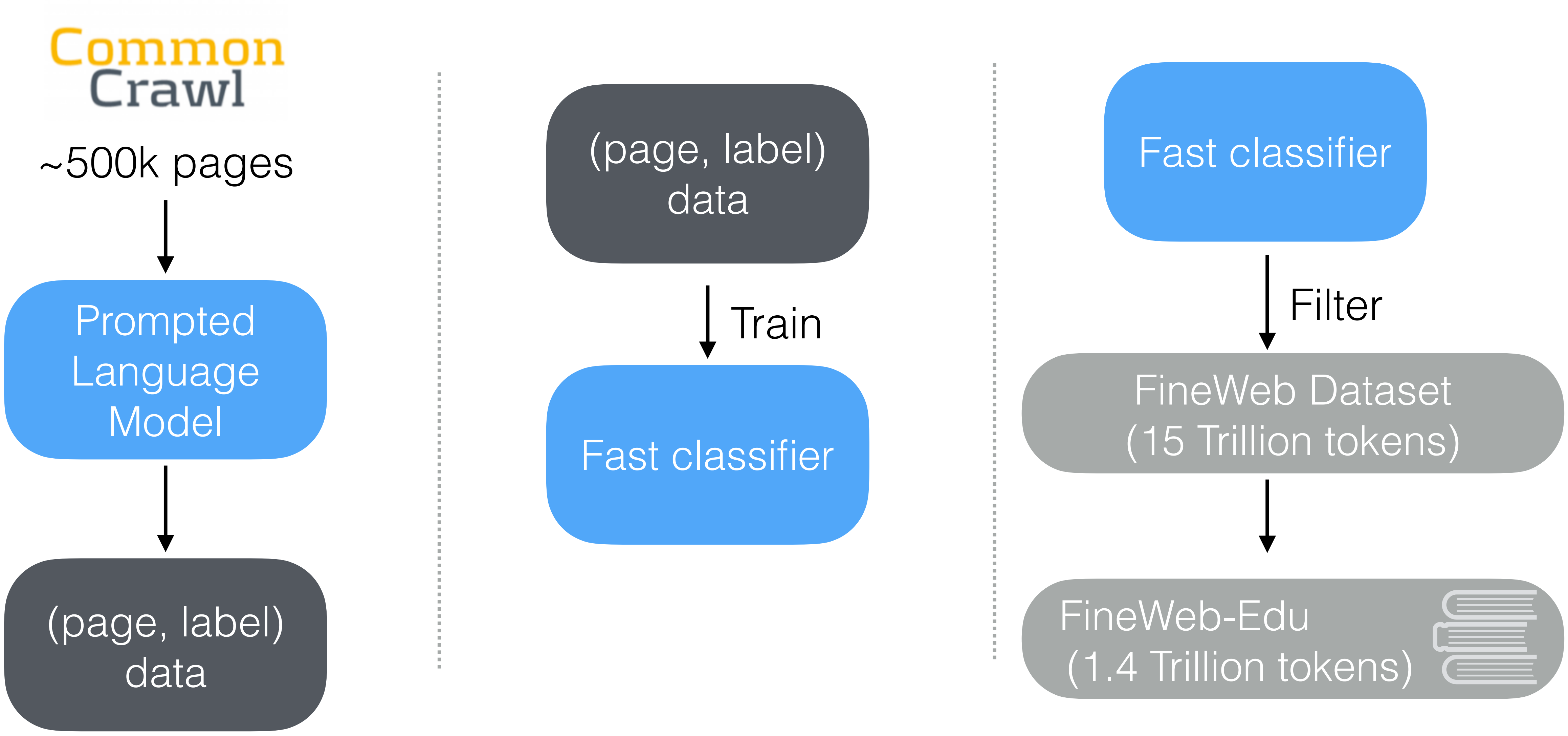


Domain	# Characters	% Characters
stackexchange.com	4,655,132,784	9.55%
nature.com	1,529,935,838	3.14%
wordpress.com	1,294,166,938	2.66%
physicsforums.com	1,160,137,919	2.38%
github.io	725,689,722	1.49%
zbmath.org	620,019,503	1.27%
wikipedia.org	618,024,754	1.27%
groundai.com	545,214,990	1.12%
blogspot.com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

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

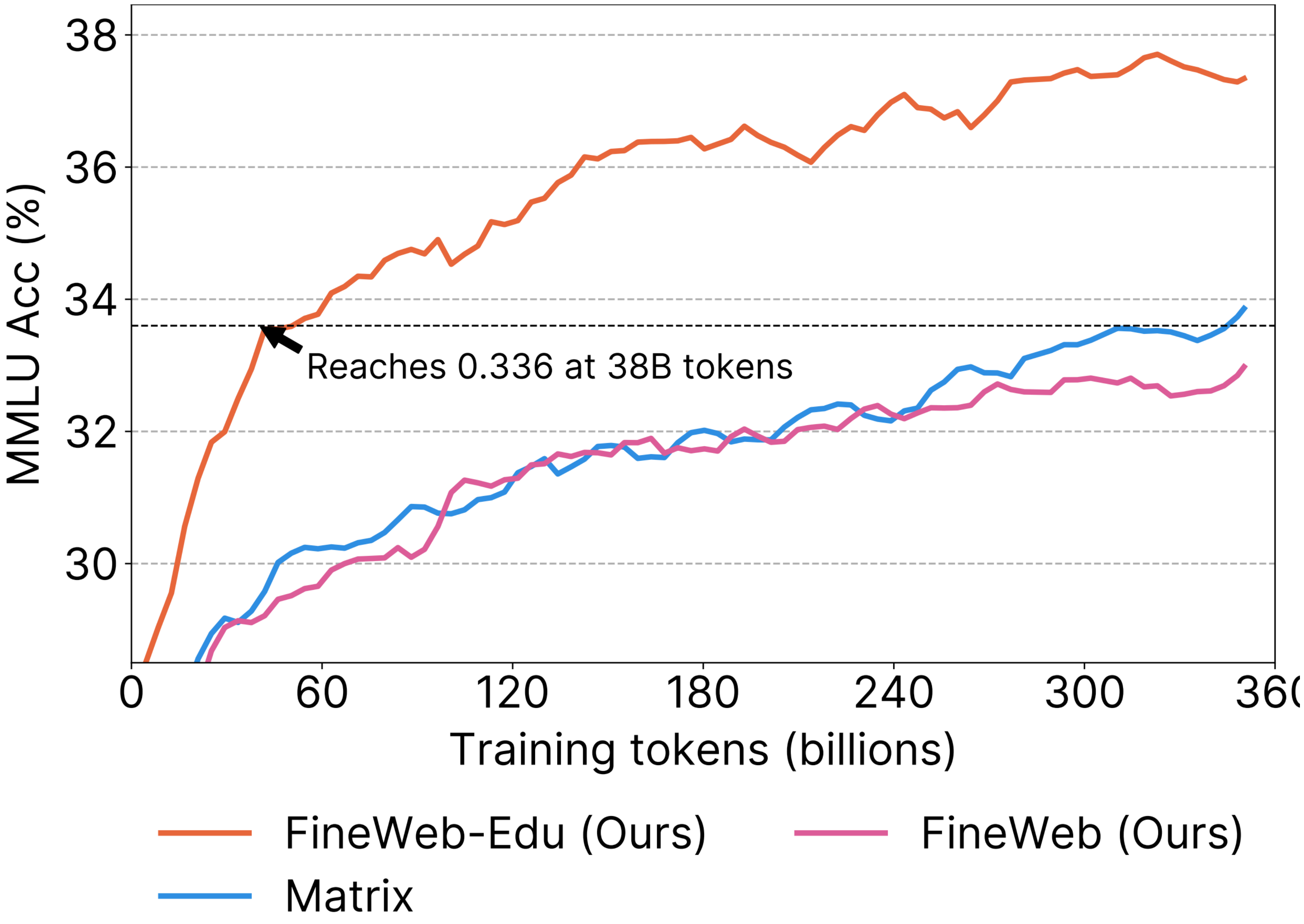
Approach: classifier filtering

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



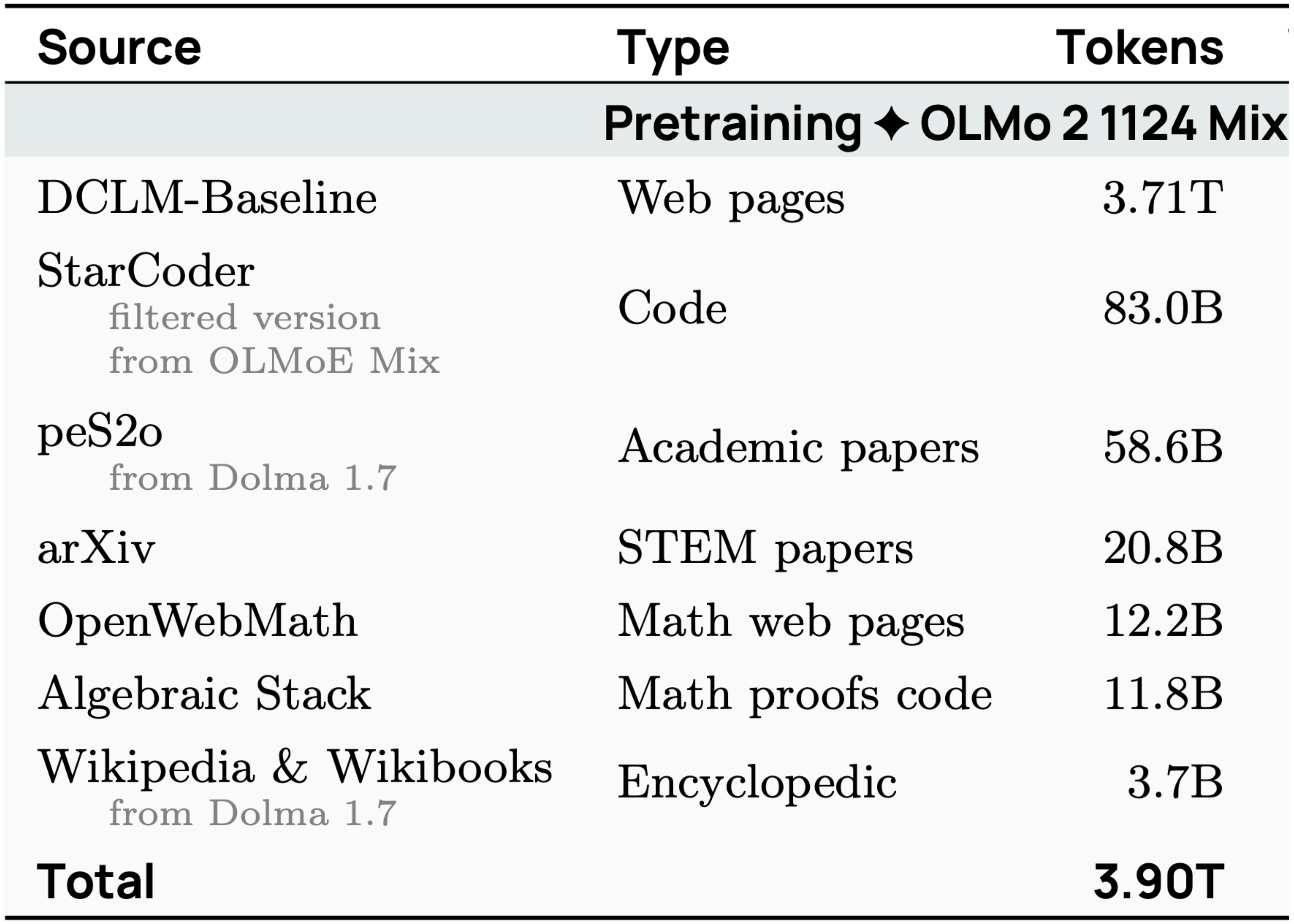
Approach: classifier filtering

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



Mixtures

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



Source	Type	Tokens
	Pretraining ✦ OLMo 2 1124 Mix	
DCLM-Baseline	Web pages	3.71T
StarCoder filtered version from OLMoE Mix	Code	83.0B
peS2o from Dolma 1.7	Academic papers	58.6B
arXiv	STEM papers	20.8B
OpenWebMath	Math web pages	12.2B
Algebraic Stack	Math proofs code	11.8B
Wikipedia & Wikibooks from Dolma 1.7	Encyclopedic	3.7B
Total		**3.90T**

Recap

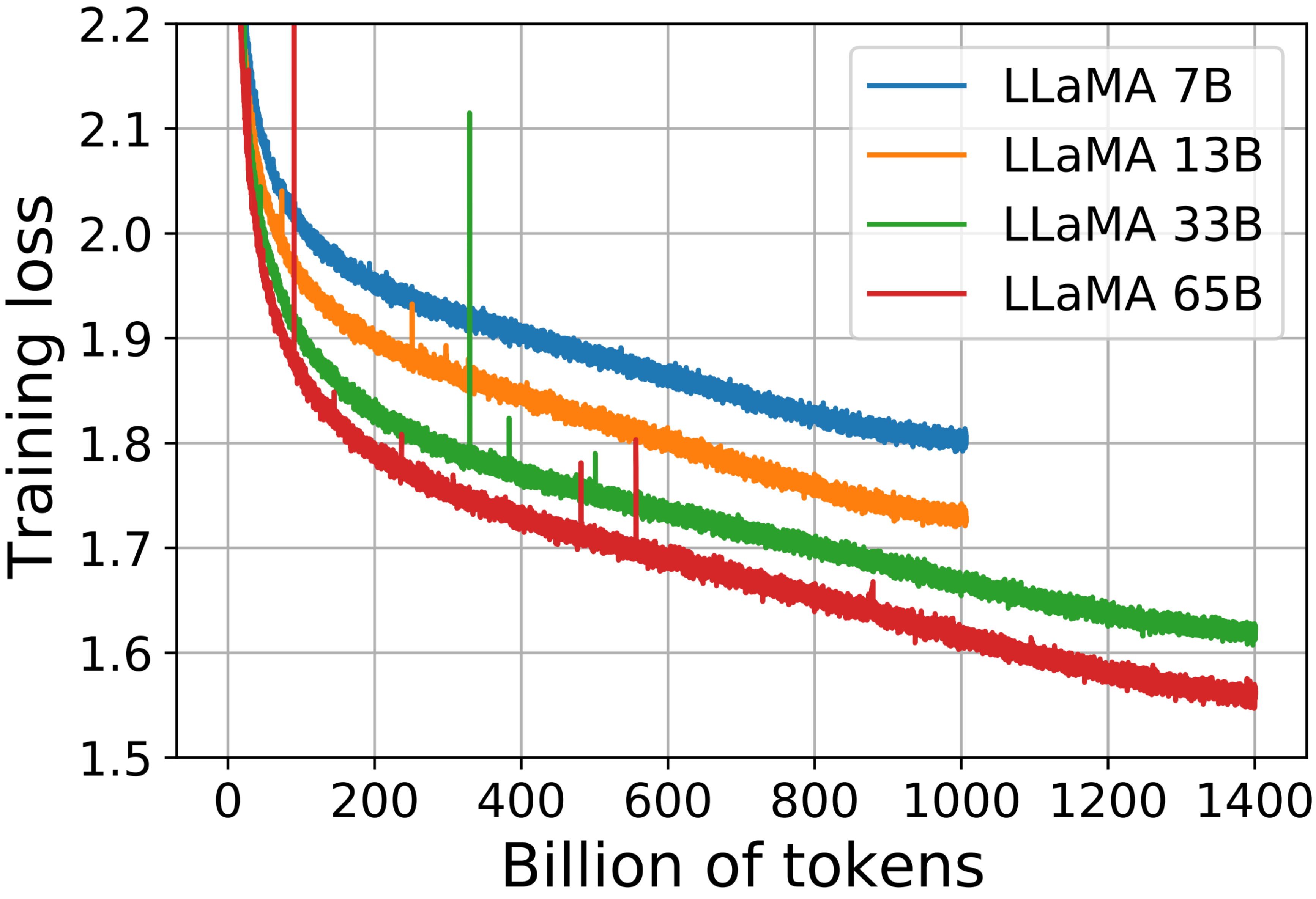
- Web data: large quantities of data
  - Extract, filter, deduplicate to improve quality
  - Filter to cover desired domain(s)
- Mix together web data and other sources to make a pre-training dataset

Recent examples

	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
OpenWebMath	2023	Math web pages	15 billion
AlgebraicStack	2023	Math code	11 billion
FineWeb-Edu	2024	Educational (middle-school)	1.4 trillion

Today’s lecture

- Tasks
- Data
- **Thinking about pretraining**
    - Tokens, model size, compute
    - Scaling laws



Bigger model

Better Loss

More data

Pretraining and compute

- Goal: get a better pretrained model by “adding more compute”
  - *“The biggest lesson that can be read from 70 years of AI research is that general methods that leverage computation are ultimately the most effective, and by a large margin.”*

    - *The Bitter Lesson*, Richard Sutton 2019

What is compute?

- We spend **compute** by performing forward and backward passes on training sequences

- An approximation for transformer language models:

$$C \approx 6ND$$

N: number of model parameters
D: number of tokens
C: compute; floating point operations (FLOPs)

Approximation from [Kaplan et al 2020]

What is compute?

- We spend **compute** by performing forward and backward passes on training sequences

- For example, Llama 2:

$$C \approx 6 \times 7 \text{ billion} \times 2 \text{ trillion}$$

$$= 8.4 \times 10^{22} \text{FLOPs}$$

N: number of model parameters
D: number of tokens
C: compute; floating point operations (FLOPs)

What is compute?

- We spend **compute** by performing forward and backward passes on training sequences

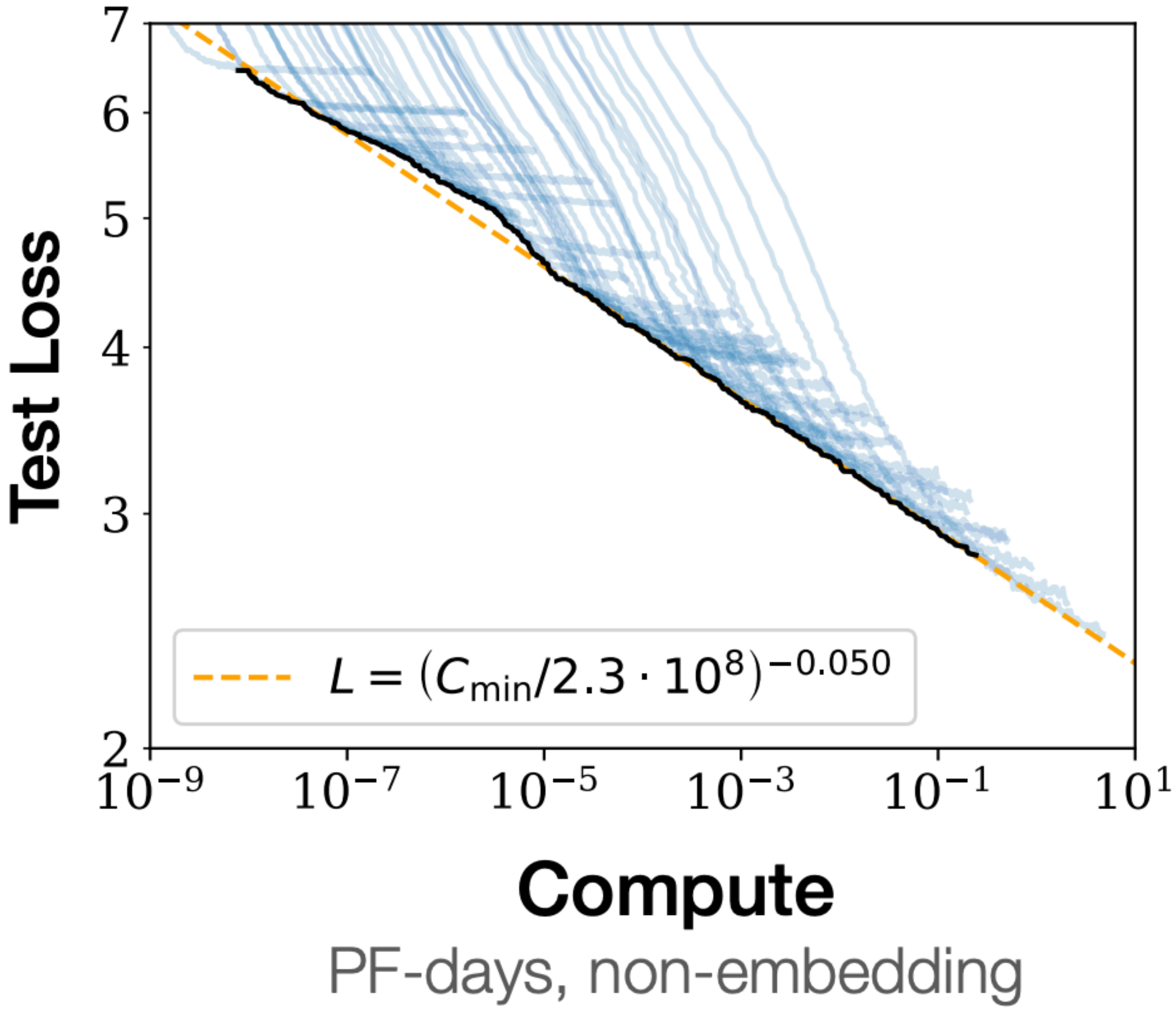
- **Increase compute:**

  - increase the **number of parameters** ($\uparrow N$)

  - train on **more tokens** $(\uparrow D)$

Scaling laws

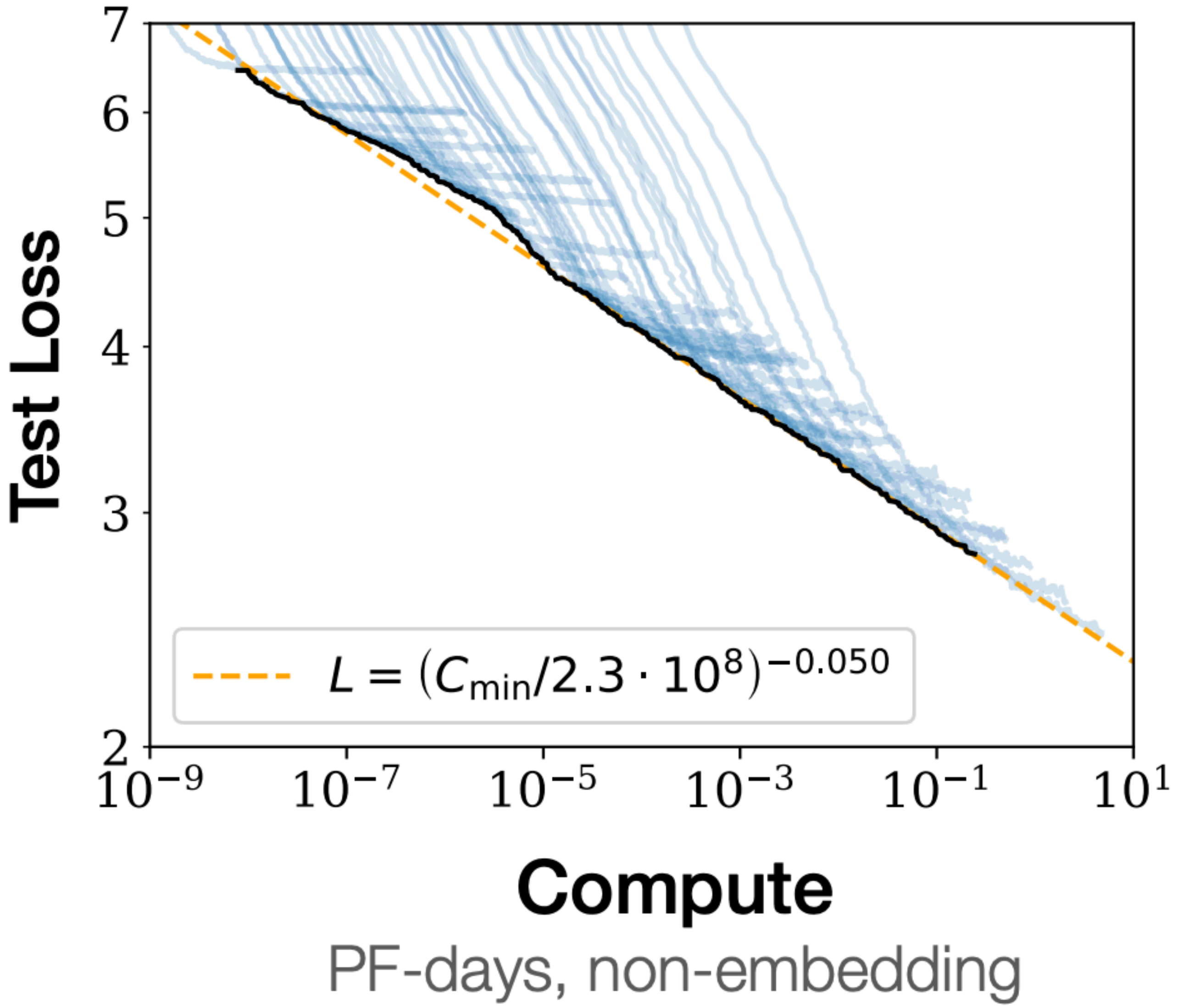
- Observed relationships between a variable (e.g., amount of compute) and loss



Scaling Laws for Neural Language Models

Scaling laws

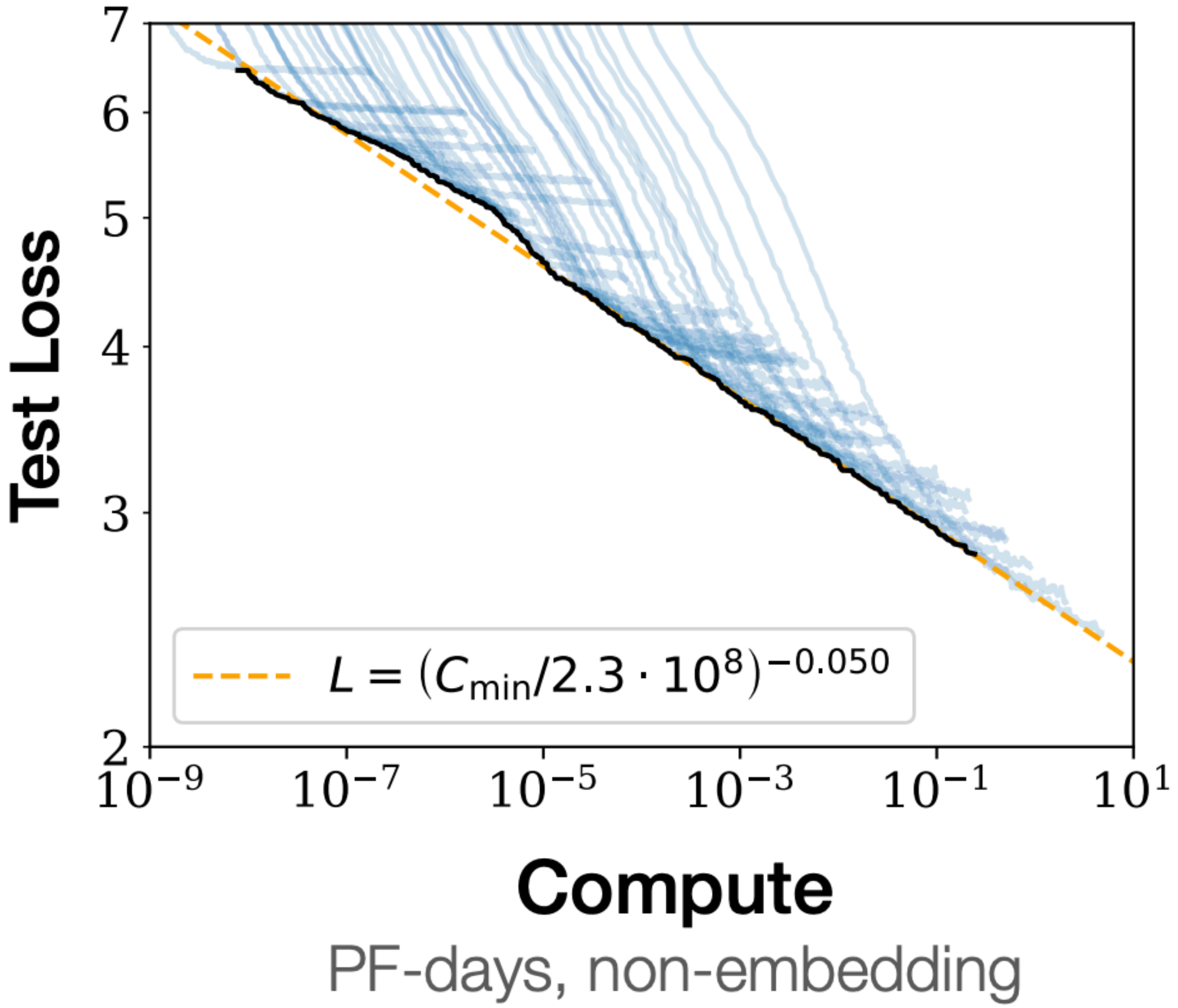
- **Basic idea**:
  - Train models of different sizes and numbers of tokens
  - Plot loss at each step of training [light blue]
  - Pick minimum loss at each amount of compute [black]
  - Run linear regression on the resulting (loss, compute) pairs [orange]



Scaling laws

Terminology:

- **Compute optimal**: black
- **Scaling law**: orange
  - E.g. $L(C) \propto 1/C^{0.05}$



Recap

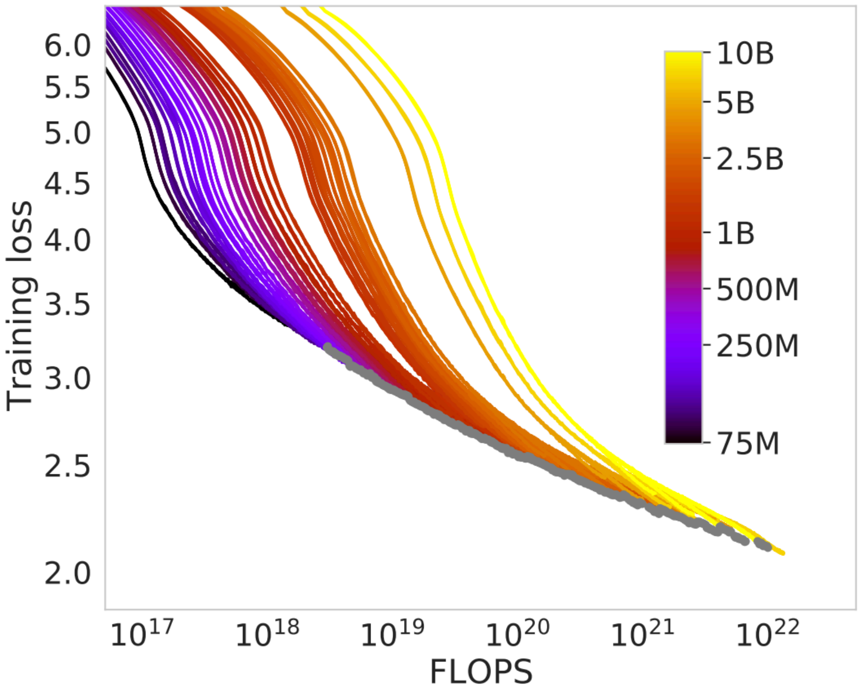
- We can think of pre-training in terms of *compute*, which is determined by *model size* and *number of tokens*

- *Scaling laws* are observed relationships between a variable (e.g., compute) and loss

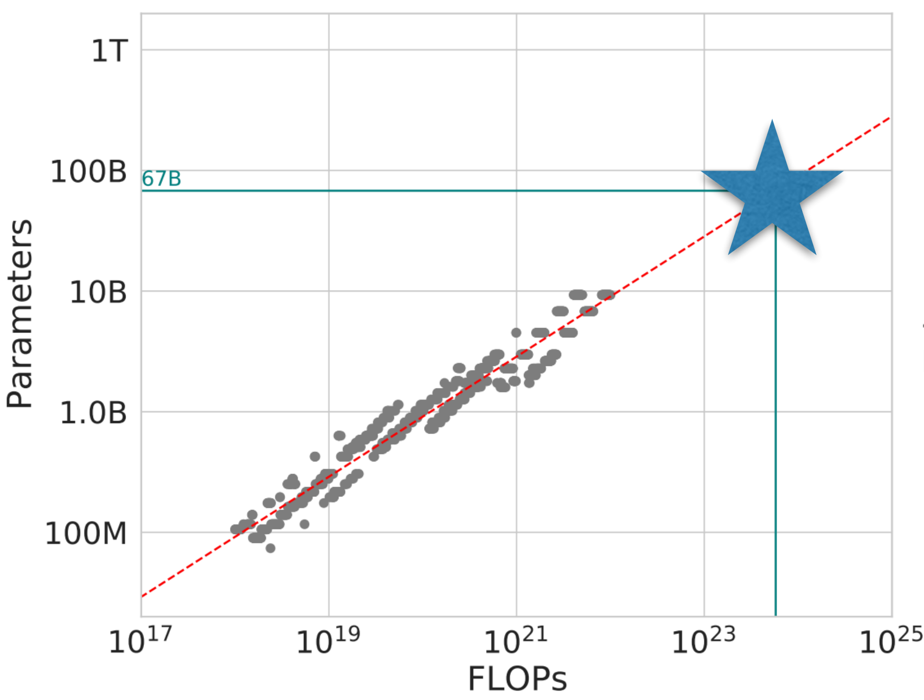
Using scaling laws

- Scaling laws are also used to choose hyper parameters
- Basic idea:
    - Run many experiments at a small scale
    - Use a scaling law to estimate the best hyper parameter for a large-scale model / training run

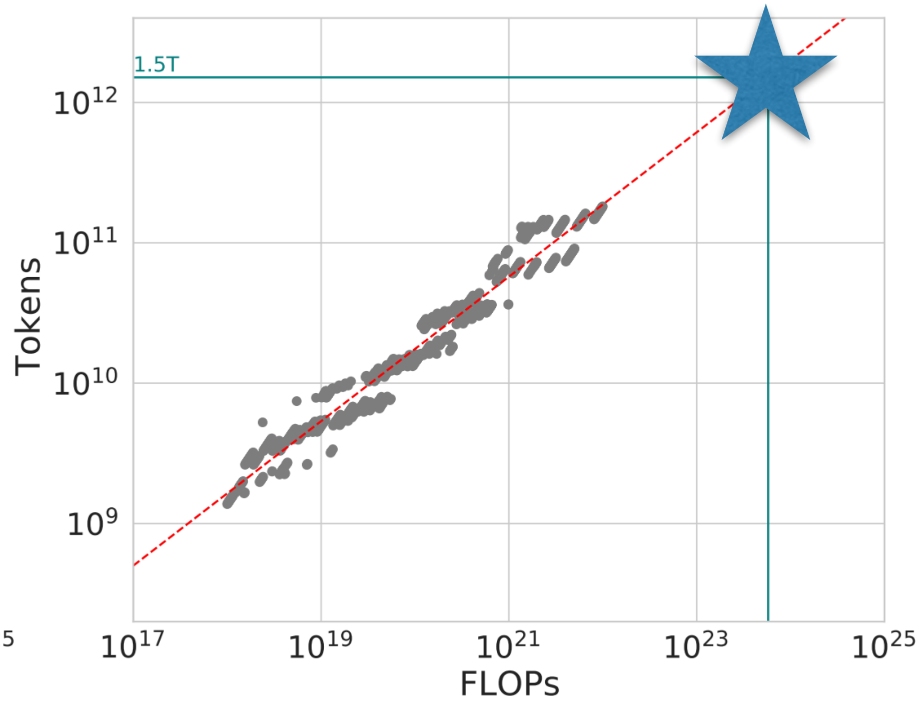
Example: choose model size and # of tokens



Run experiments



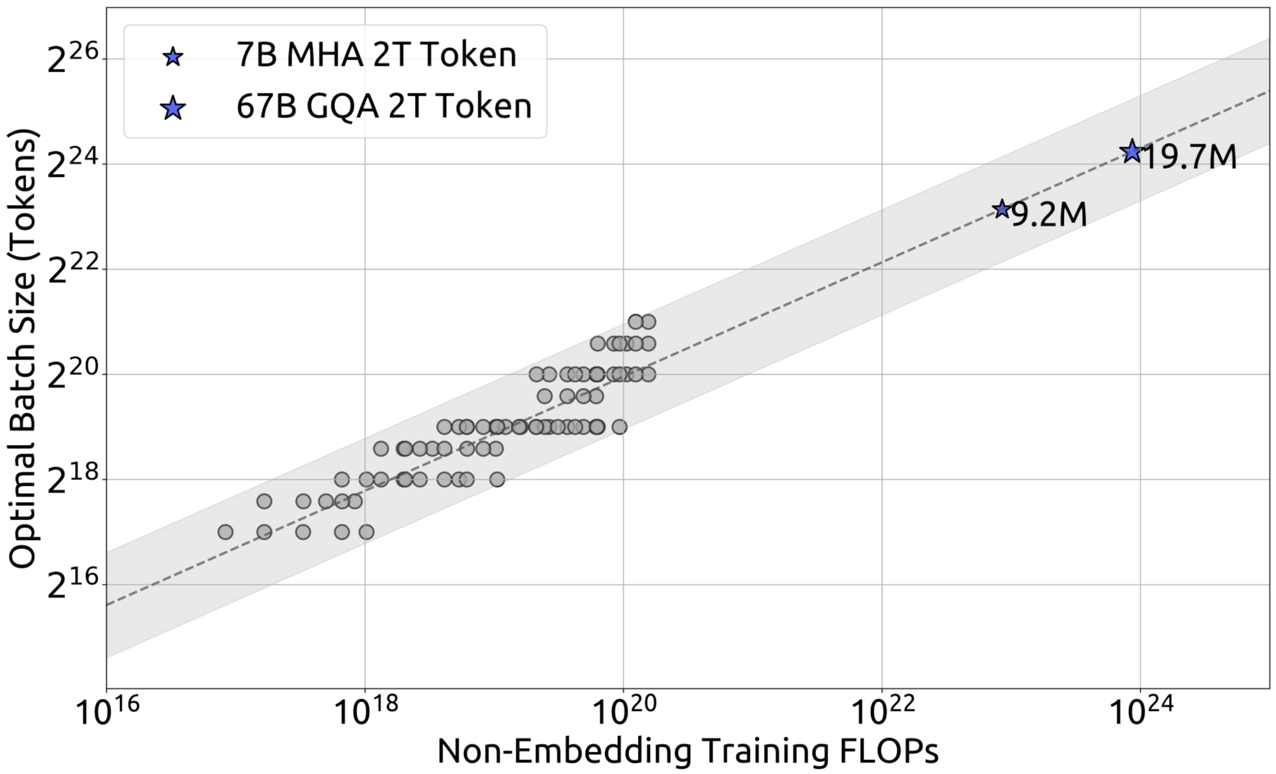
Fit a line and predict optimal model size



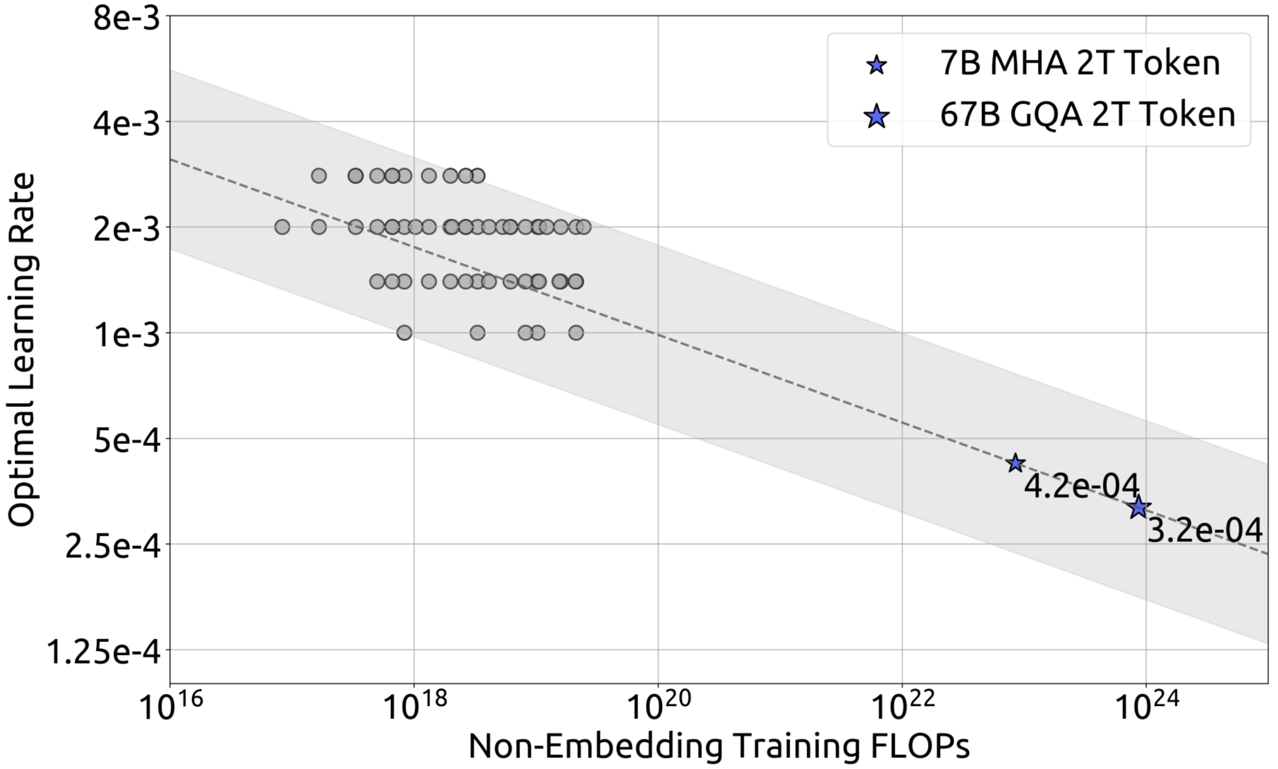
Fit a line and predict optimal # of tokens

Training Compute-Optimal Large Language Models

Example: choose batch size, learning rate



Optimal batch size



Optimal learning rate

DeepSeek LLM: Scaling Open-Source Language Models with Longtermism

Today’s lecture

- Pretraining tasks
  - Masked language modeling
  - Autoregressive language modeling
- Pretraining data: sources, quality, and quantity
- Thinking about pretraining
  - Tokens, model size, compute
  - Scaling laws

Thank you