CMPU 366 · Natural Language Processing

Large Language Models

Part 2

29 October 2025

Where are we?

A ***language model*** gives the probability for the next token given some prefix: $P(w_i \mid w_{<i})$

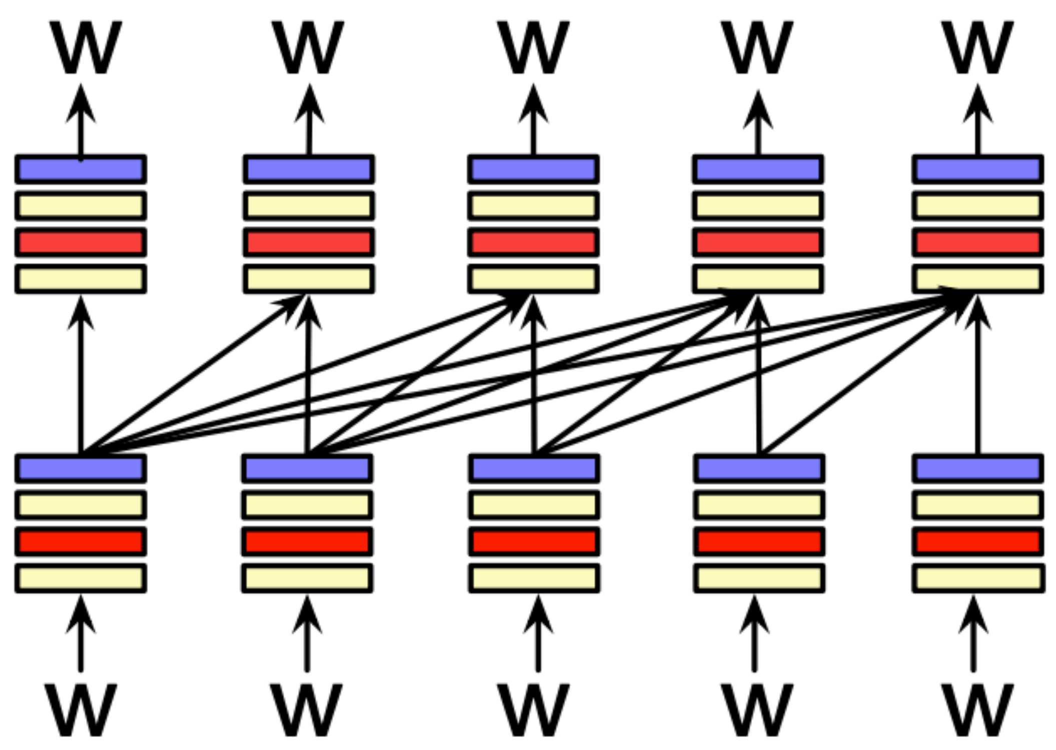
Using these probabilities, we can also compute the probability of an entire sequence of tokens (using the chain rule) – or to generate new text.

A *large language model* is distinguished from traditional language models by

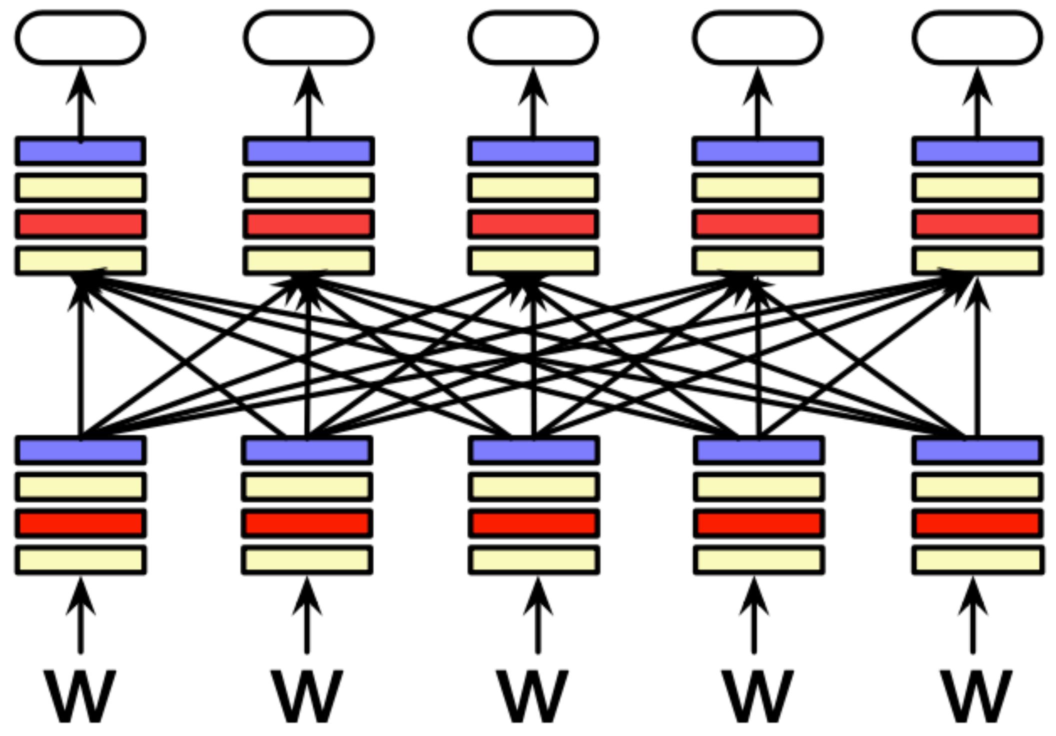
the use of neural networks to learn to predict the next word given a variable-length context rather than counting words seen after a fixed-length prefix and

the ("large") number of parameters in the network.

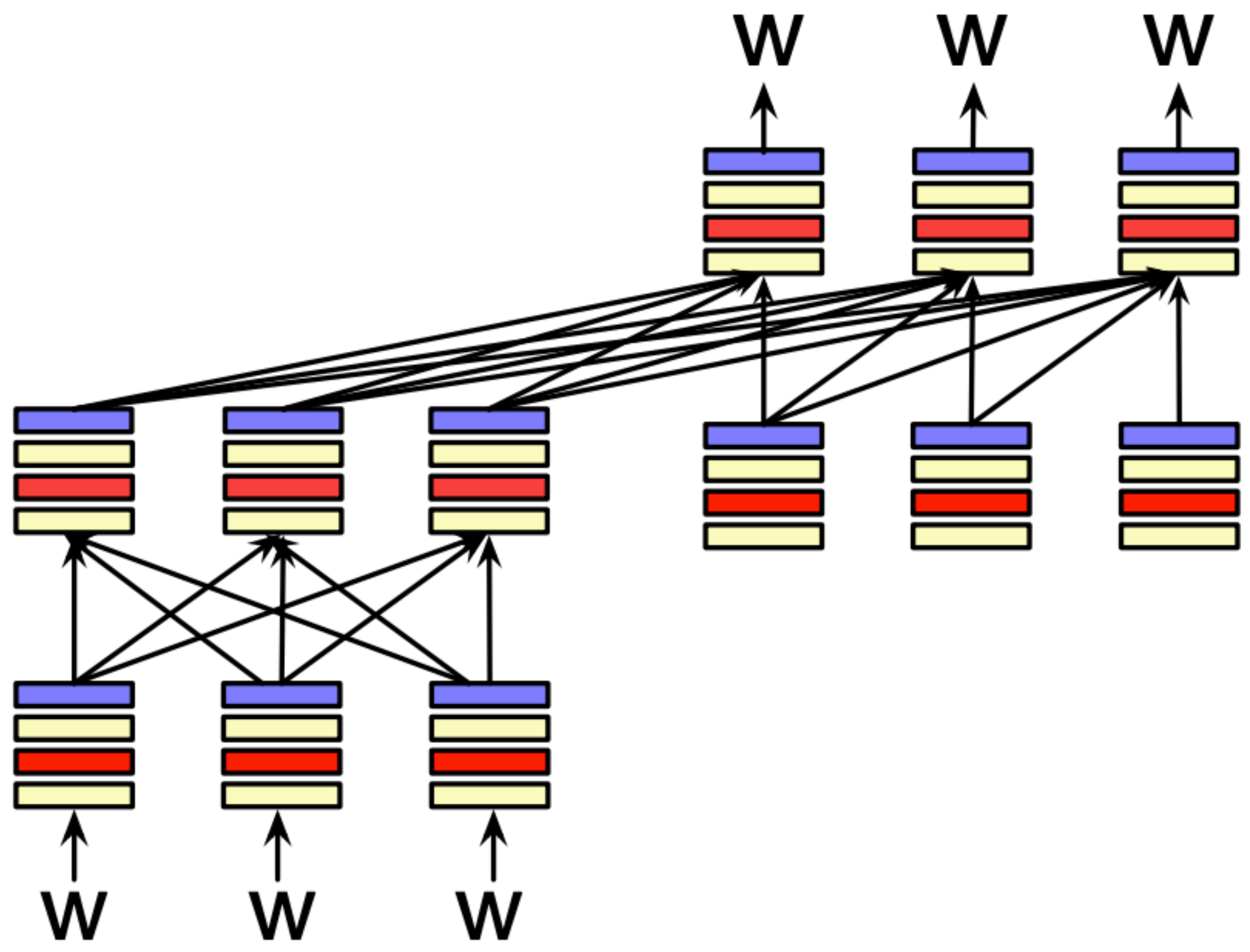
Three architectures for LLMs



Decoders



Encoders

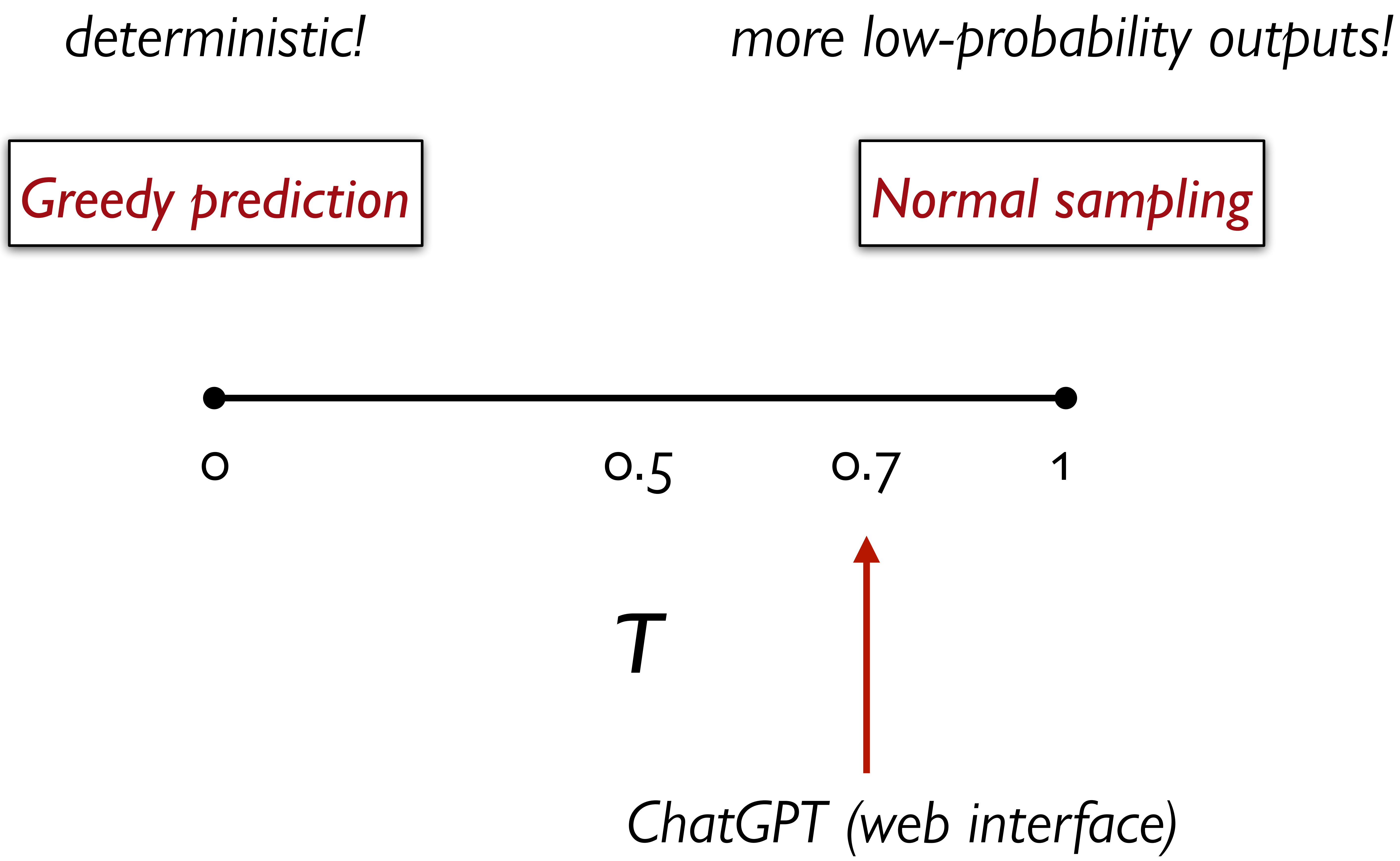


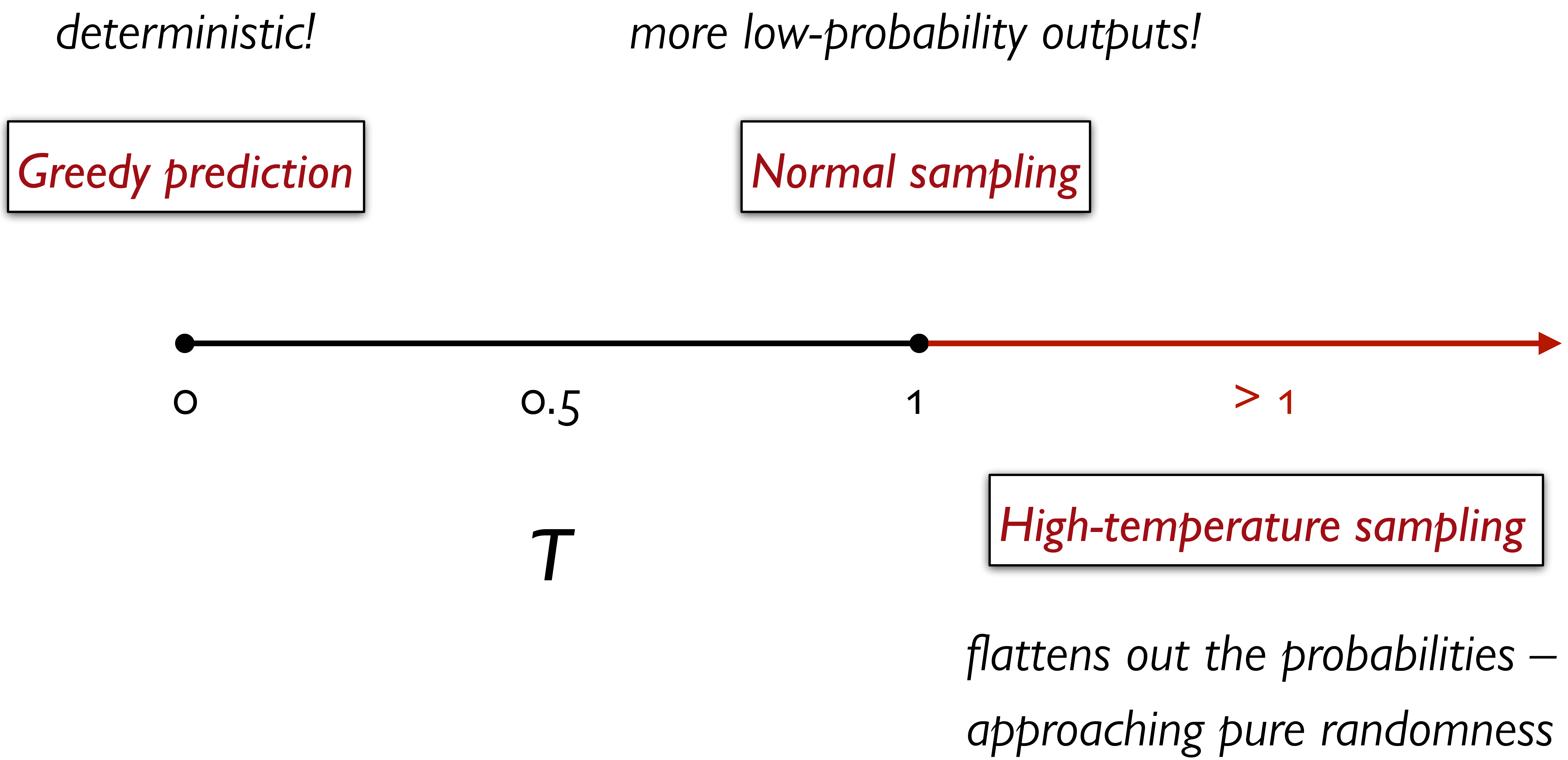
Encoder–decoders

Text generation using LLMs depends on the choice of decoding method:

greedy decoding or

sampling.

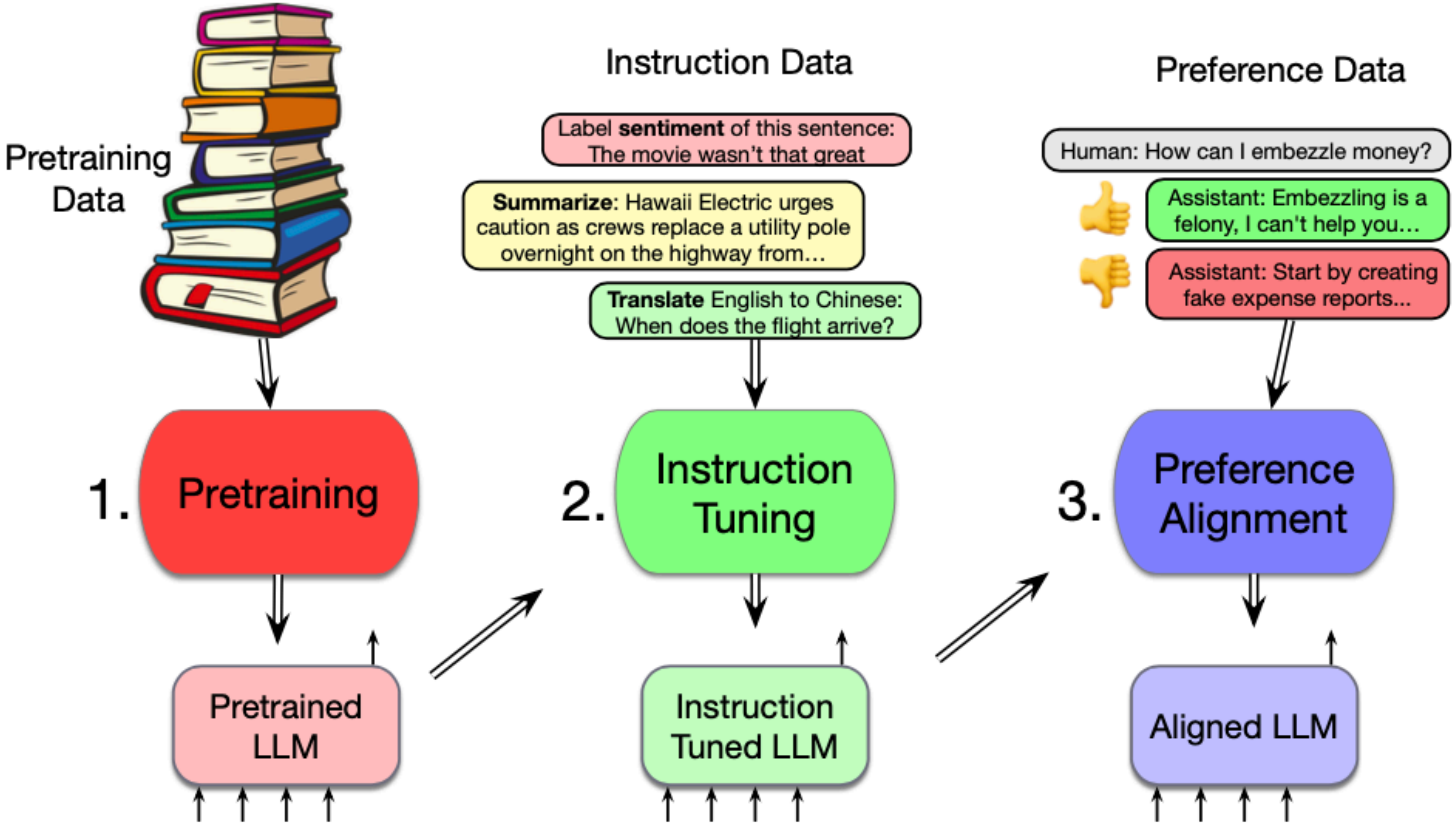

deterministic!
more low-probability outputs!
Greedy prediction
Normal sampling
0
0.5
0.7
1
T
ChatGPT (web interface)


deterministic!
Greedy prediction
more low-probability outputs!
Normal sampling
0
0.5
1
> 1
T
High-temperature sampling
flattens out the probabilities –
approaching pure randomness

Notebook: *Decoding with LLMs*

Pretraining large language models (*continued*)

Three stages of training in LLMs



We train the model to predict the next word:

1. Take a corpus of text
2. At each time step t,

   ask the model to predict the next word

   train the model using gradient descent to minimize the error in this prediction

Since the correct next word is the one that occurs in the text, this is *self-supervised* training.

The *cross-entropy loss* is the negative log probability that the model assigns to the true next word *w*.

We want the loss to be high if the model assigns too low a probability to *w*.

When it does so, we move the model weights in the direction that assigns it a higher probability.

Cross-entropy loss measures the difference between the correct probability distribution and the predicted distribution:

$$L_{CE}(\hat{\mathbf{y}}_t, \mathbf{y}_t) = -\sum_{w \in V} \mathbf{y}_t[w] \log \hat{\mathbf{y}}_t[w]$$

The correct distribution $\mathbf{y}_t$ is 1 for the *actual next word* w_{t+1} and 0 for the others. So all the terms get multiplied by zero except for one – the log probability the model assigns to the correct next word. Therefore it's just

$$L_{CE}(\hat{\mathbf{y}}_t, \mathbf{y}_t) = -\log \hat{\mathbf{y}}_t[w_{t+1}]$$

Cross-entropy loss measures the difference between the correct probability distribution and the predicted distribution:

$$L_{CE}(\hat{\mathbf{y}}_t, \mathbf{y}_t) = -\sum_{w \in V} \underbrace{\boxed{\mathbf{y}_t[w]}}_{} \log \boxed{\hat{\mathbf{y}}_t[w]}$$

correct *predicted*

The correct distribution $\mathbf{y}_t$ is 1 for the *actual next word* w_{t+1} and 0 for the others. So all the terms get multiplied by zero except for one – the log probability the model assigns to the correct next word. Therefore it's just

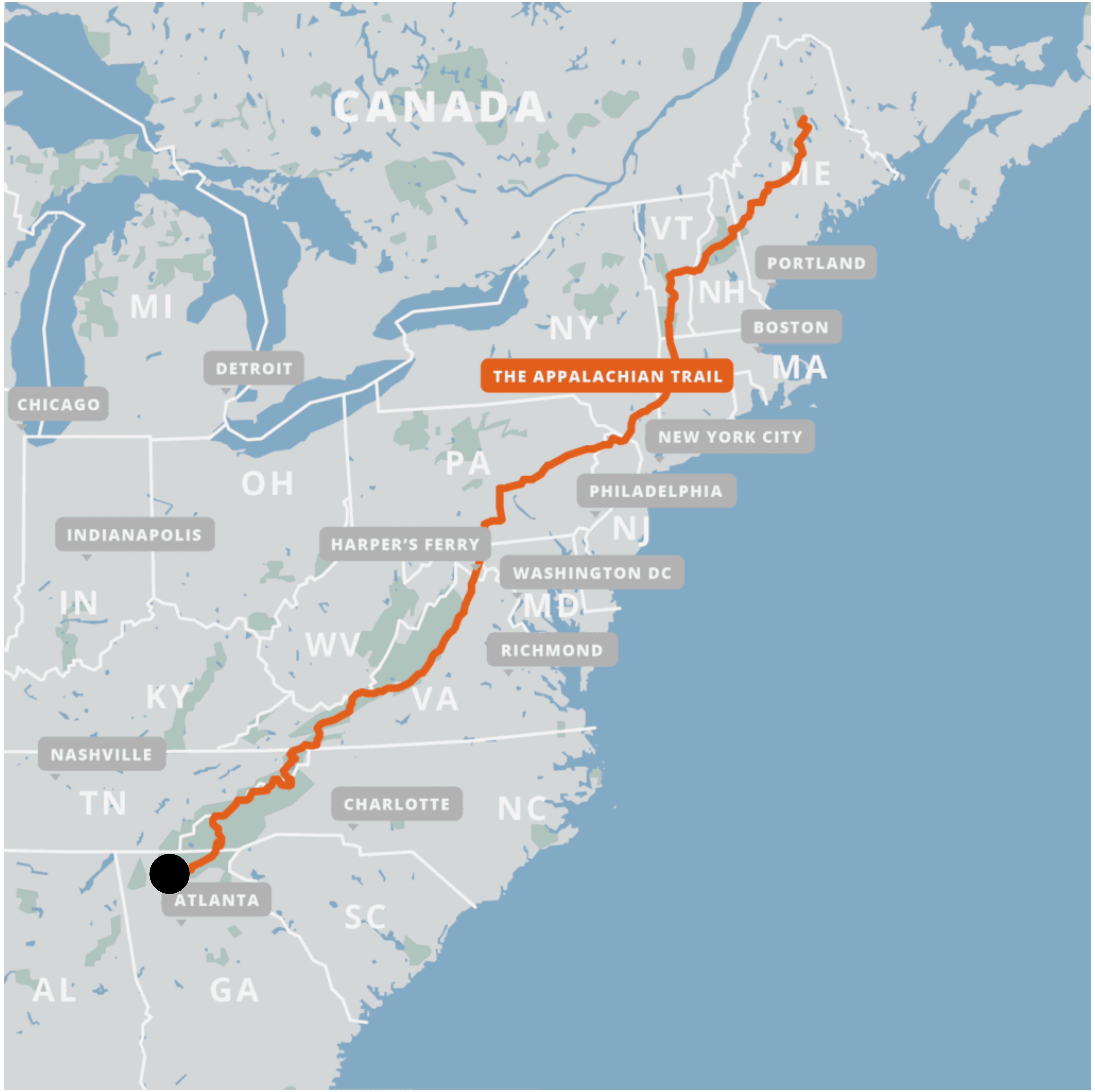
$$L_{CE}(\hat{\mathbf{y}}_t, \mathbf{y}_t) = -\log \hat{\mathbf{y}}_t[w_{t+1}]$$

Teacher forcing

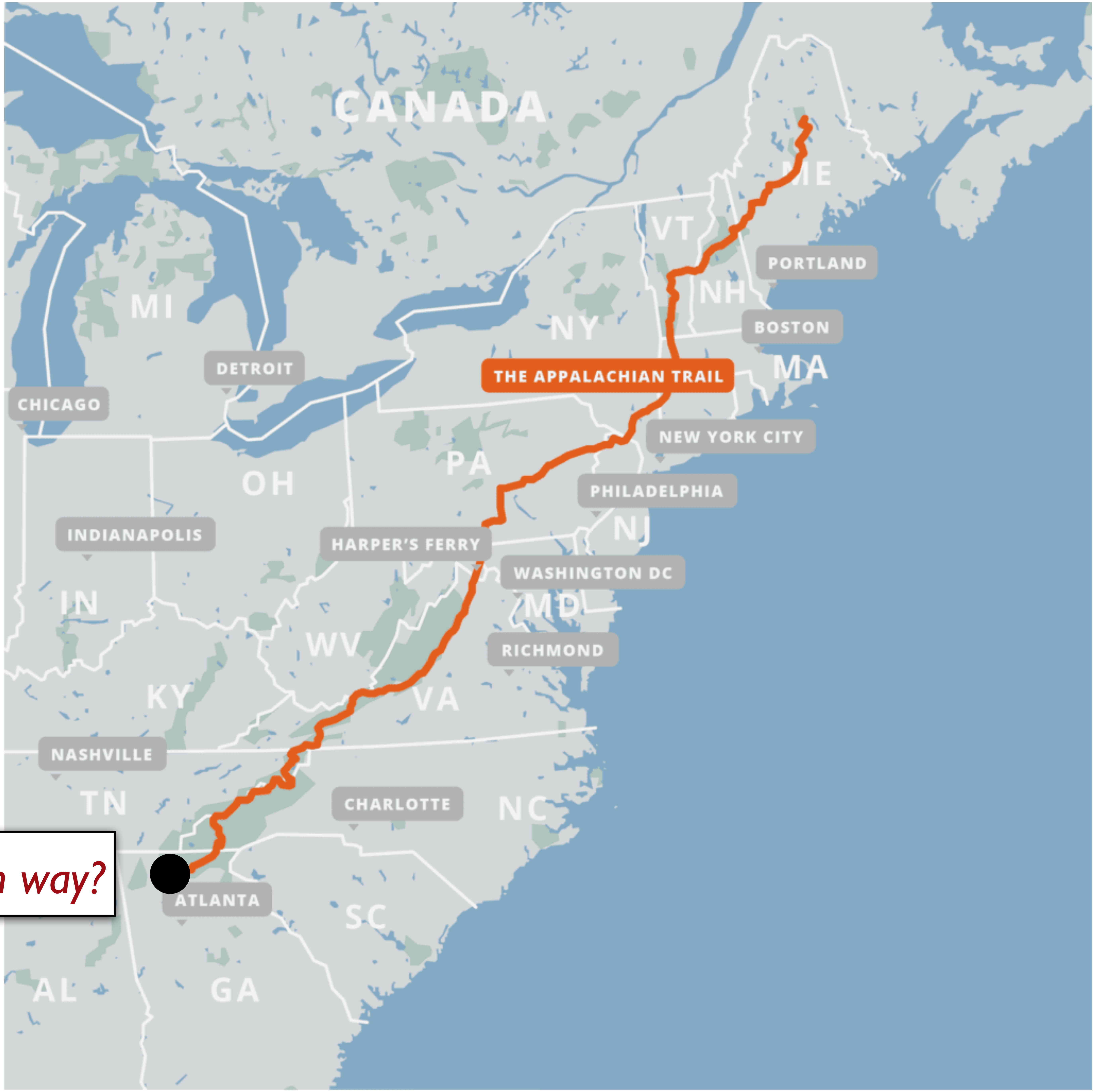
At each token position t, the model sees correct tokens $w_{1:t}$ and computes the loss for the next token w_{t+1}.

At next token position $t+1$, we ignore what model *predicted* for w_{t+1}. Instead, we take the *correct* word w_{t+1}, add it to context, move on.

Teacher forcing

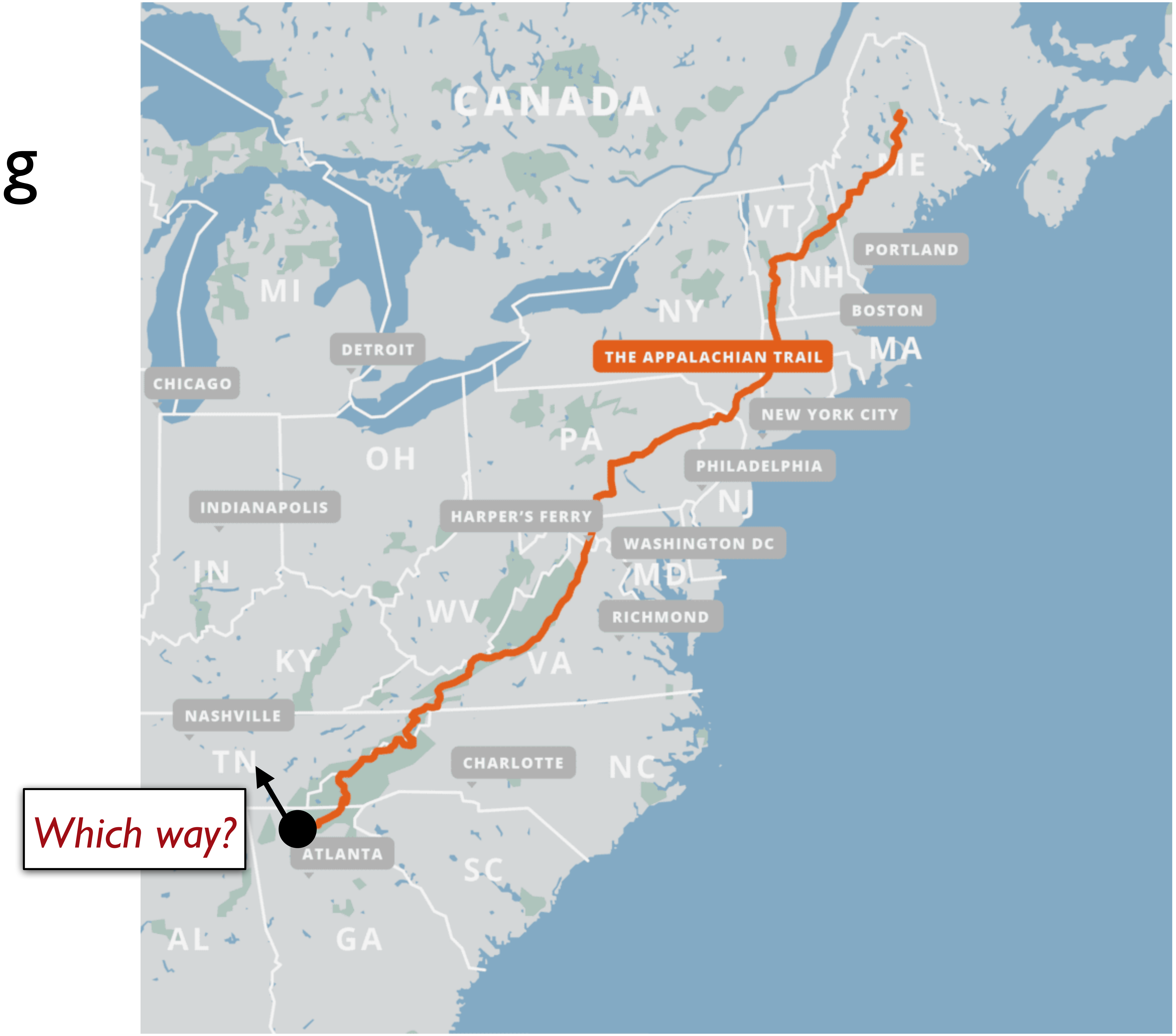


Teacher forcing



Which way?

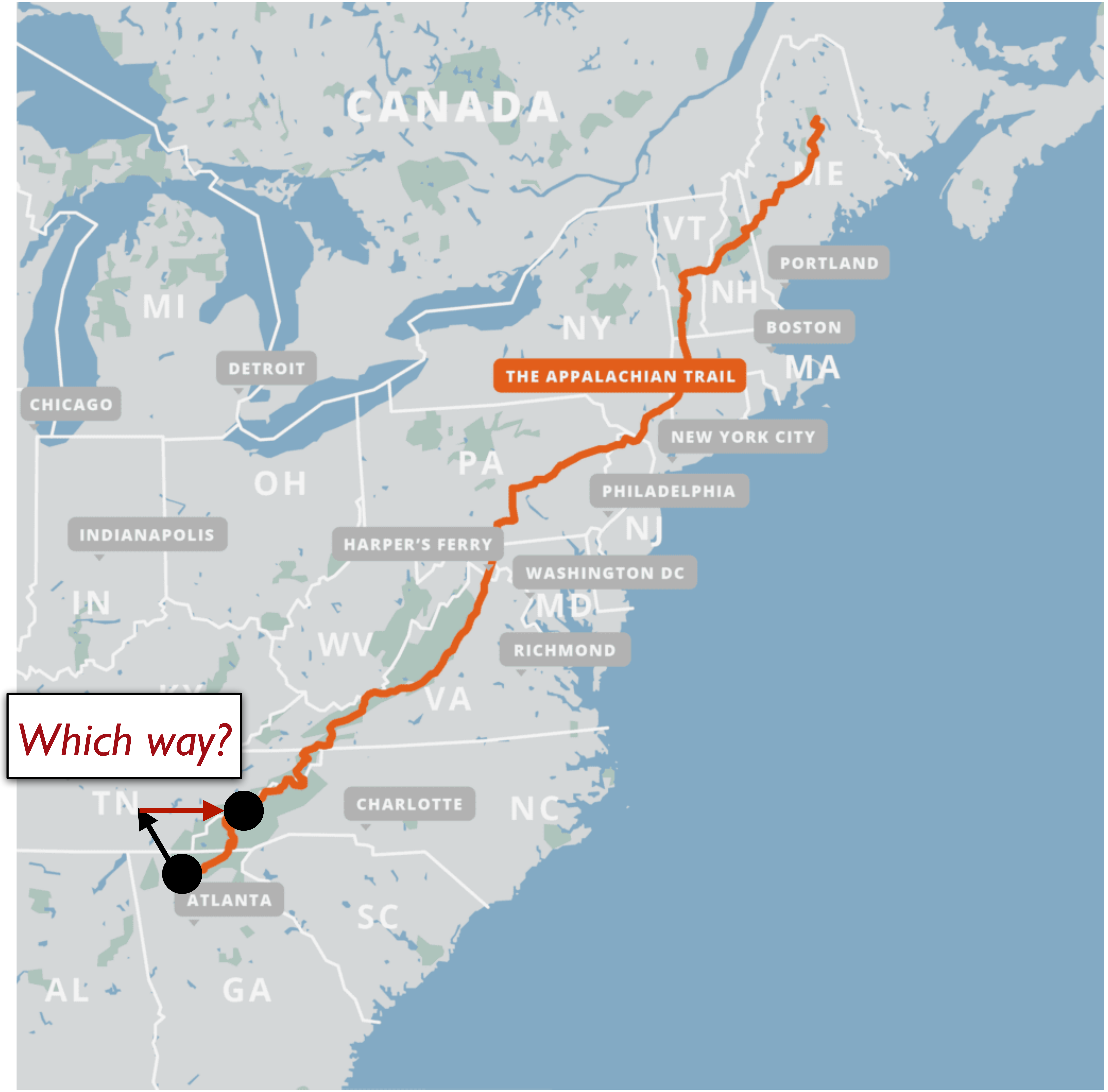
Teacher forcing



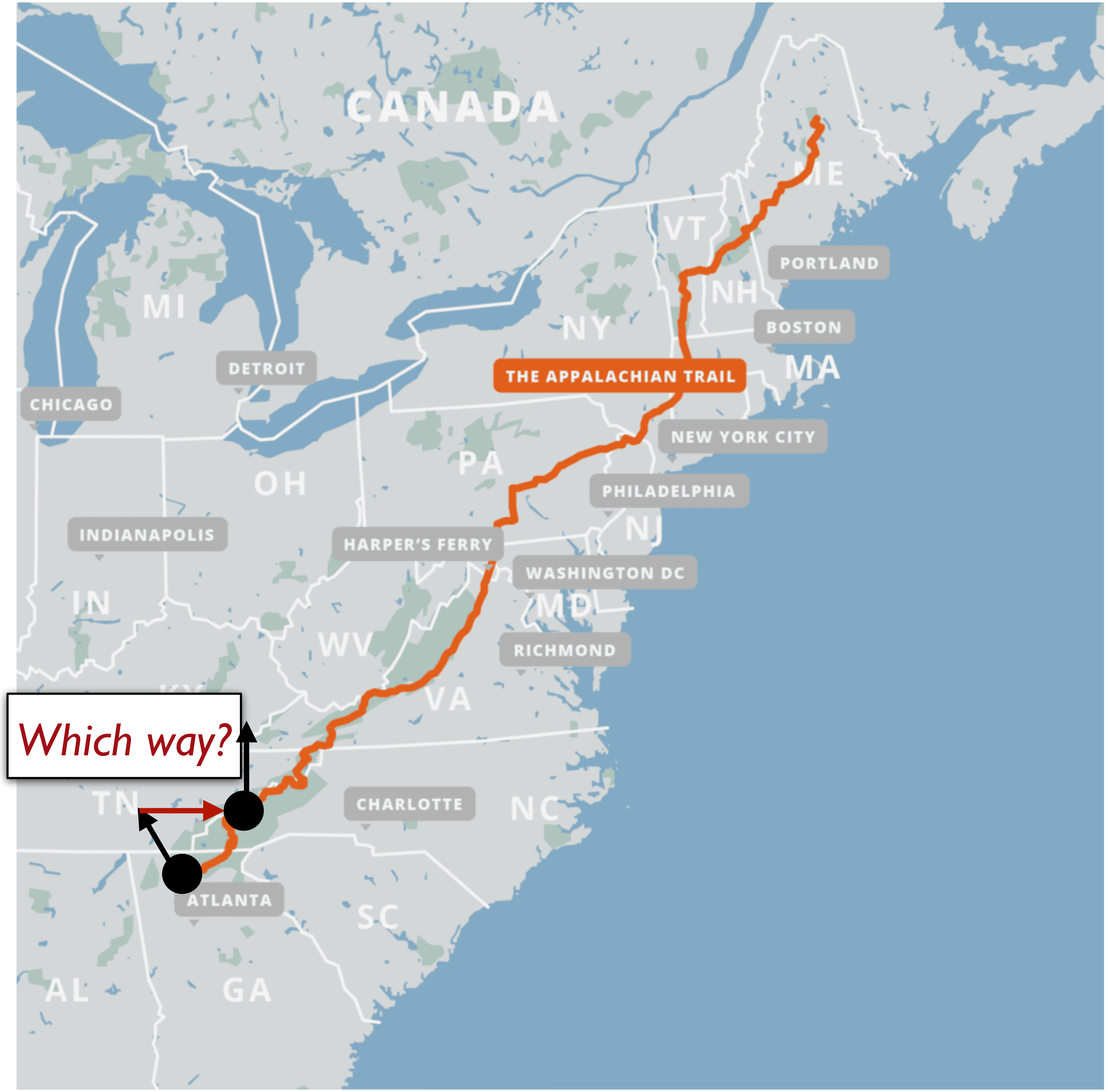
Teacher forcing

Teacher forcing

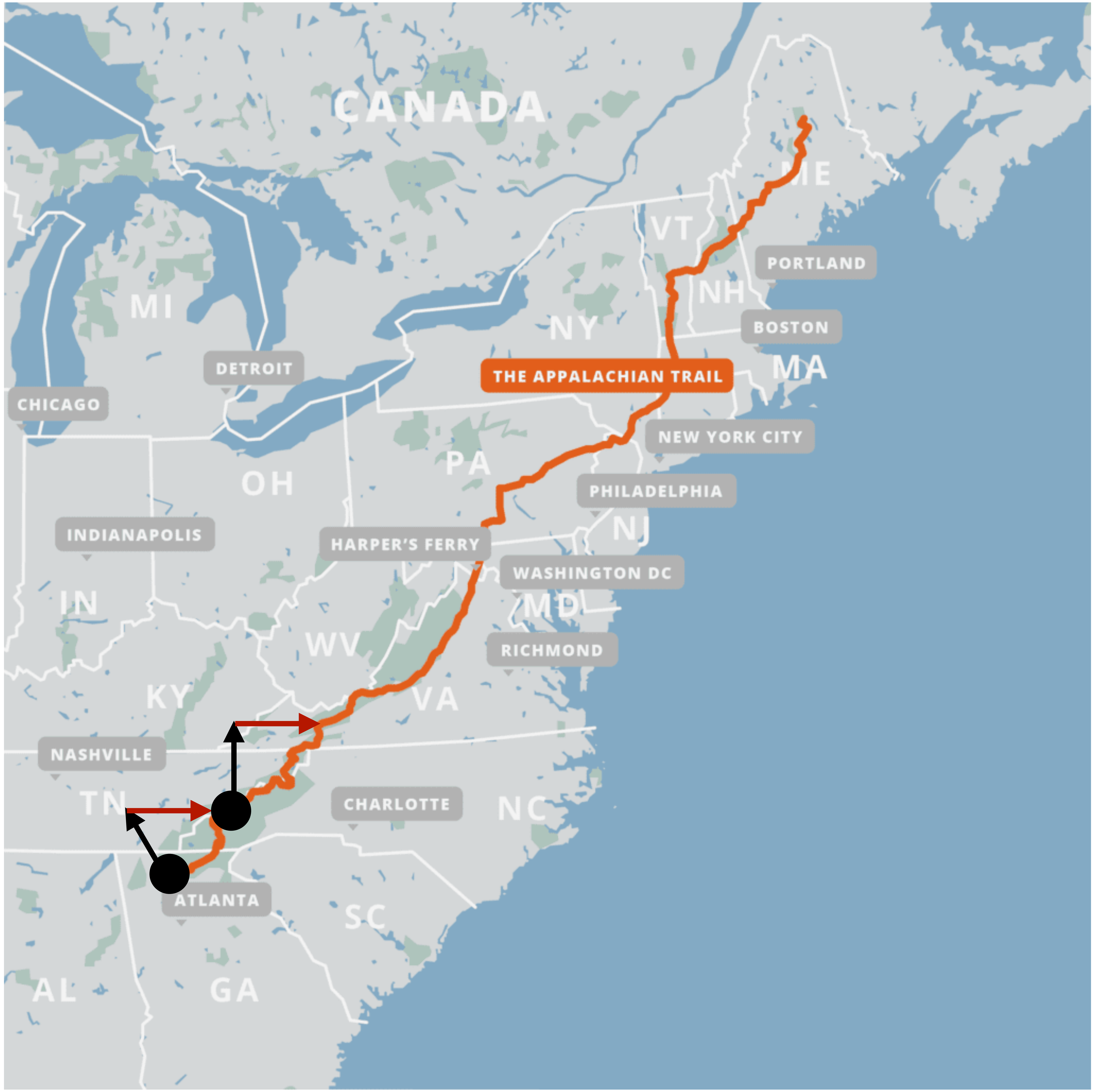
Teacher forcing



Teacher forcing

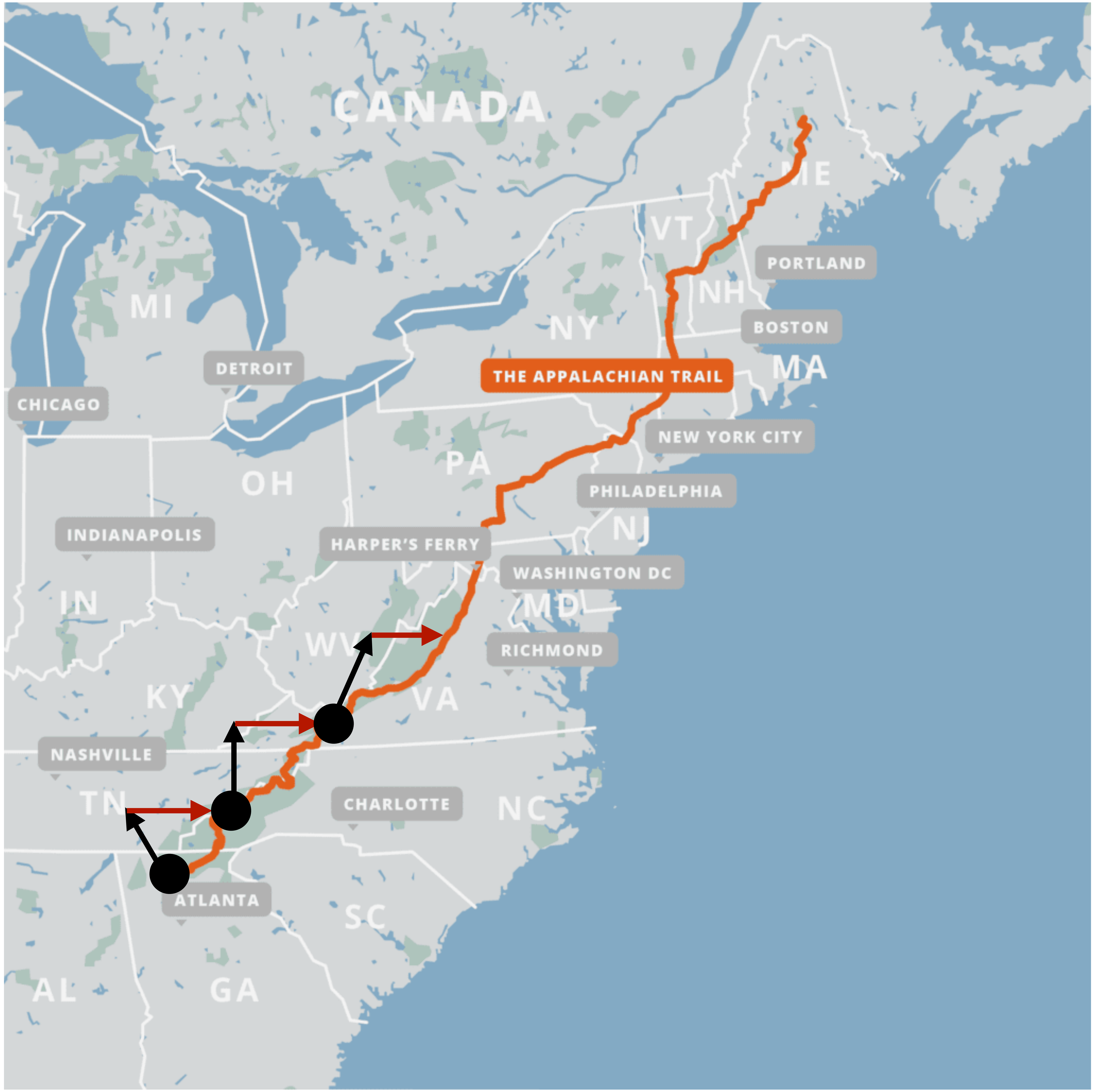


Teacher forcing

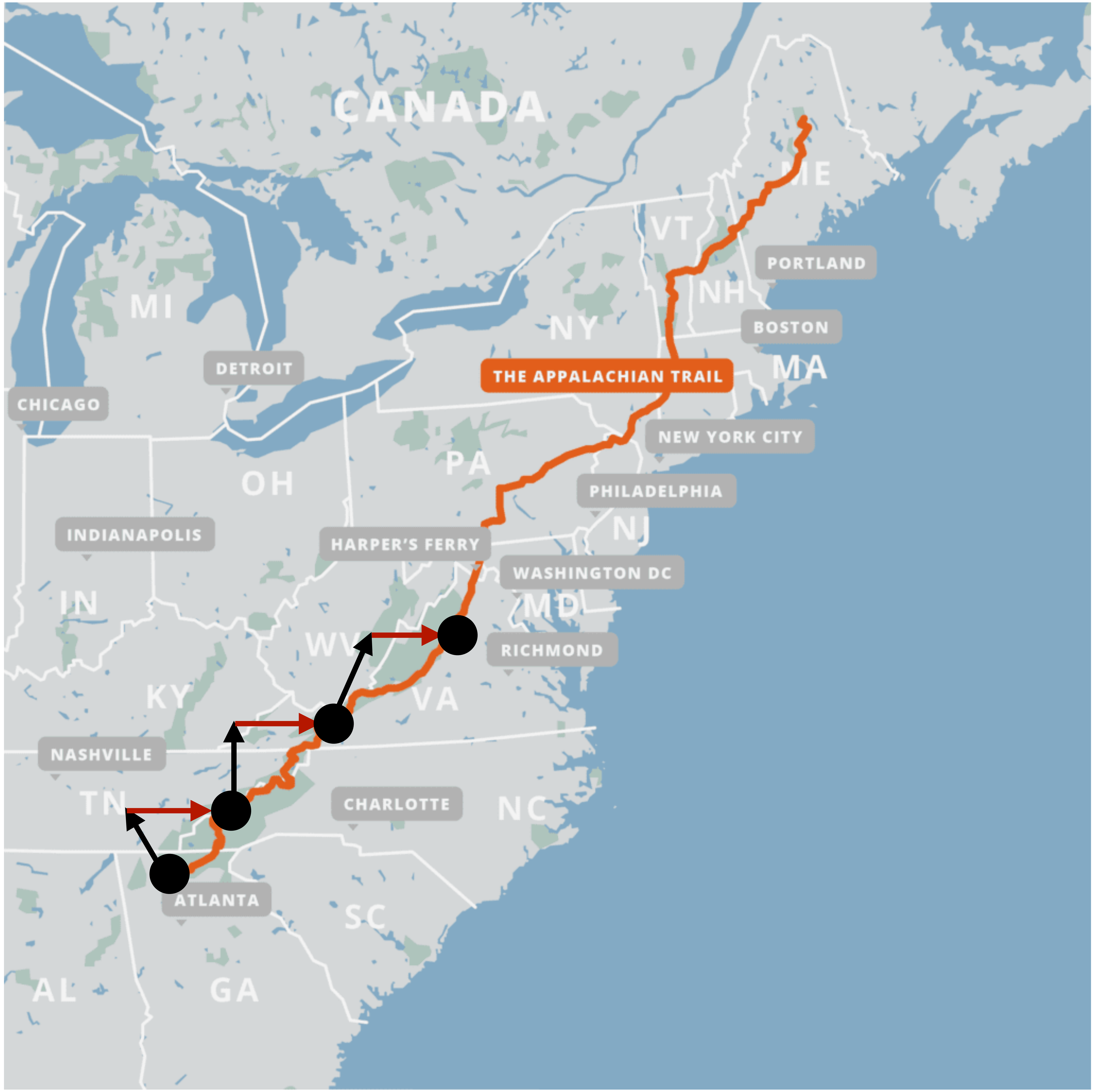


Teacher forcing

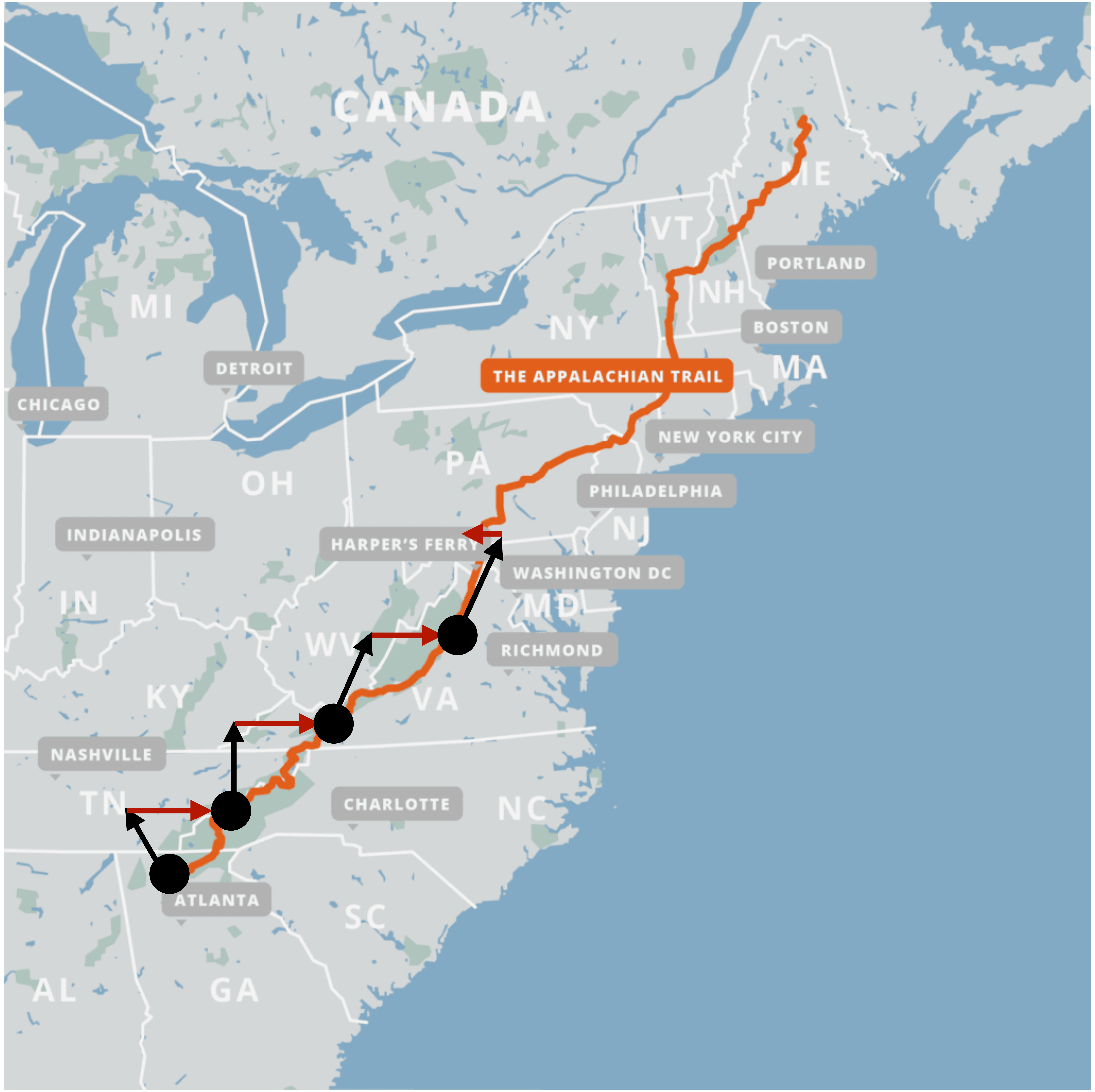
Teacher forcing



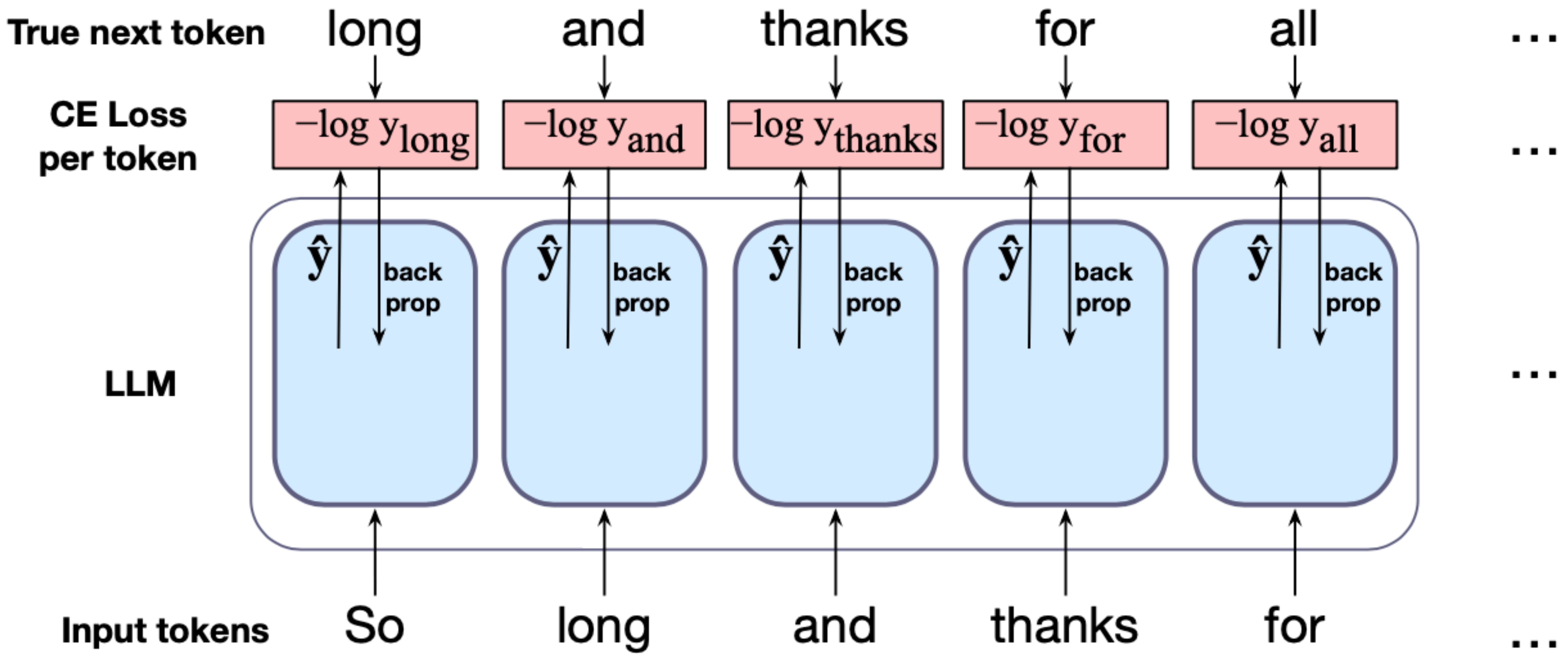
Teacher forcing



Teacher forcing



Training a transformer language model



What to read?

What to read?

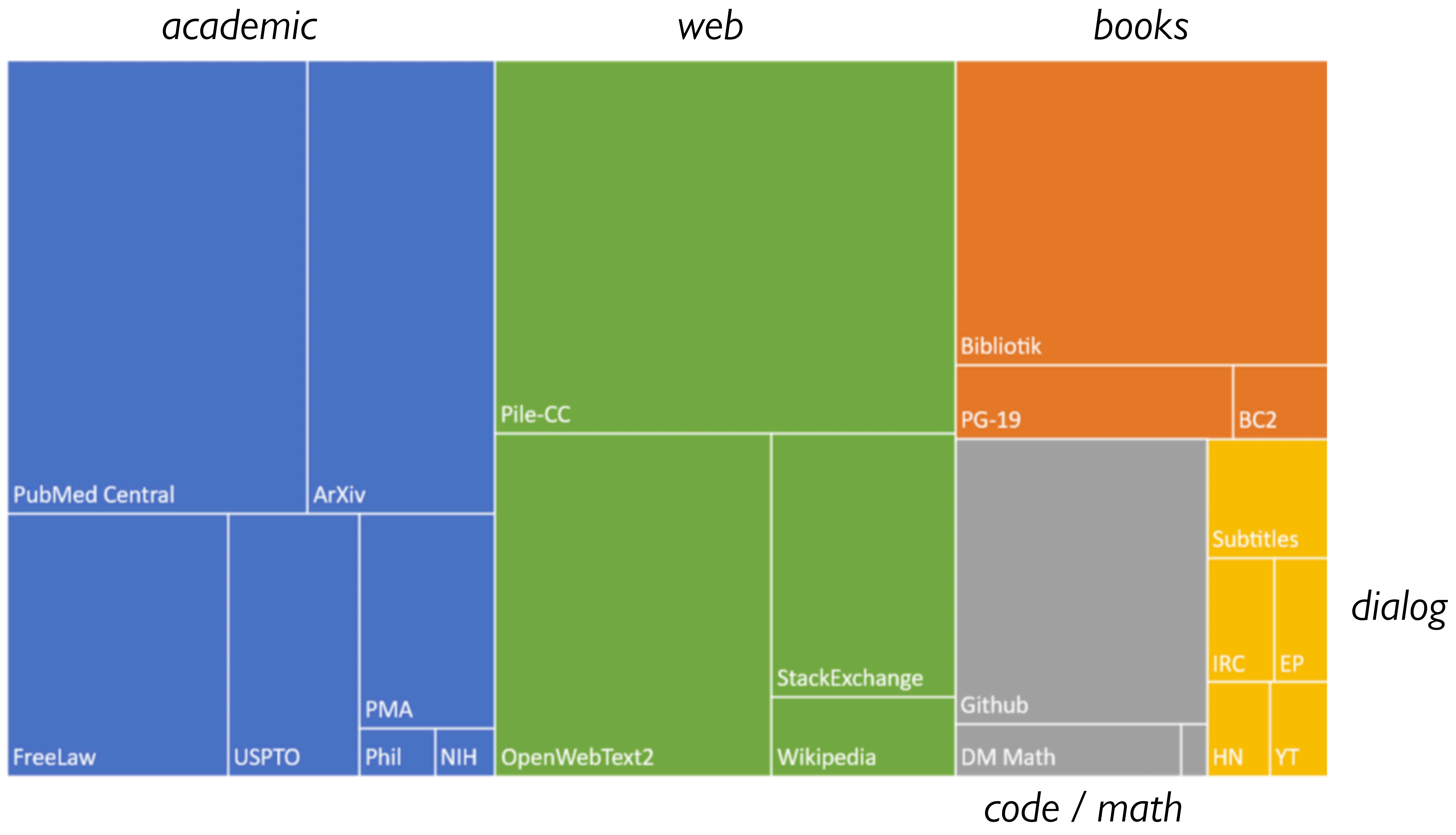
LLMs are mainly trained on text from the Web.

Common crawl consists of snapshots of billions of webpages produced by the non-profit Common Crawl.

Colossal Clean Crawled Corpus (C4) is a filtered corpus of common crawl data, consisting of 156 billion tokens of English – mostly patent text documents, Wikipedia, and news sites.

Raffel et al., 2020

The Pile: a pretraining corpus



The Library of Congress contains more than 32 million books on Twitter that "I cannot imagine how we are supposed to

GPT-4's predecessor, GPT-3, was trained on forty-five terabytes of text data, which, according to its successor, is the word-count equivalent of around ninety million novels. These included Wikipedia entries, journal articles, newspaper punditry, instructional manuals, Reddit discussions, social-media posts, books, and any other text its developers could commandeer, typically without informing or compensating the creators. It is unclear how many more terabytes of data were used to train GPT-4, or where they came from, because OpenAI, despite its name, says only in the technical report that GPT-4 was pre-trained "using both publicly available data (such as internet data) and data licensed from third-party providers" and adds that "given both the competitive landscape and the safety implications of large-scale models like GPT-4, this report contains no further details about the architecture (including model size), hardware, training compute, dataset construction, training method, or similar."

This secrecy matters because, as impressive as GPT-4 and other A.I. models that process everyday, natural language may be, they also can present dangers. As Sam Altman, the C.E.O. of OpenAI, recently told ABC News, "I'm particularly worried that these models could be used for large-scale disinformation."

Filtering

Quality is subjective

Many LLMs attempt to match Wikipedia, books, particular websites

Try to remove boilerplate, adult content

Deduplication at many levels (URLs, documents, even lines)

Safety also subjective

Toxicity detection is important, although that has mixed results

Can mistakenly flag data written in dialects like African American English www.reddit.com/r/microwavegang/top/?t=all

reddit

r/microwavegang
Search in r/microwavegang

Log In

u/Bitter_Gold_5023 · 8 mo. ago

MMMMMMMmmmmmmmmmmMMMMMMMMMMMMMMMMmmmmmmmm
MMMMMMMM

MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMm
mmmmmmmmmmmmmmmmmmmmmmMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMm
mmmmmmmmmmmmmmmmmmmmmmmmmmmmMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
MMMMMMMMMMMMMMMMMMMmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmm
mmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmMMMMMMMMMMMMMMMMMMMMMMM...

240 · 17 · Share

u/Legopeeps · 6 yr. ago

mmm? Jkjk... mmm?

other

microwavegang

welcome to the microwave gang! here we talk about topics such as "MMMMMMMMMMMMMmmMMMMMM" and "BEEP BEEP BEEP BEEP BEEP" we did something to ai too i guess

Created Sep 25, 2019

Public

1.7K Weekly visitors

22 Weekly contributions

R/MICROWAVEGANG RULES

1 must be microwave based

MODERATORS

Moderator list hidden. Learn More

View all moderators

Reddit Rules Privacy Policy User Agreement Accessibility www.nytimes.com/2023/12/27/business/media/new-york-times-open-a

MEDIA

The New York Times

GIVE THE TIMES

Account

The Times Sues OpenAI and Microsoft Over A.I. Use of Copyrighted Work

Millions of articles from The New York Times were used to train chatbots that now compete with it, the lawsuit said.

Share full article

1.3K

A lawsuit by The New York Times could test the emerging legal contours of generative

Other issues with scraping the Web

Data consent

Website owners can indicate they don’t want their site crawled (`robots.txt`)

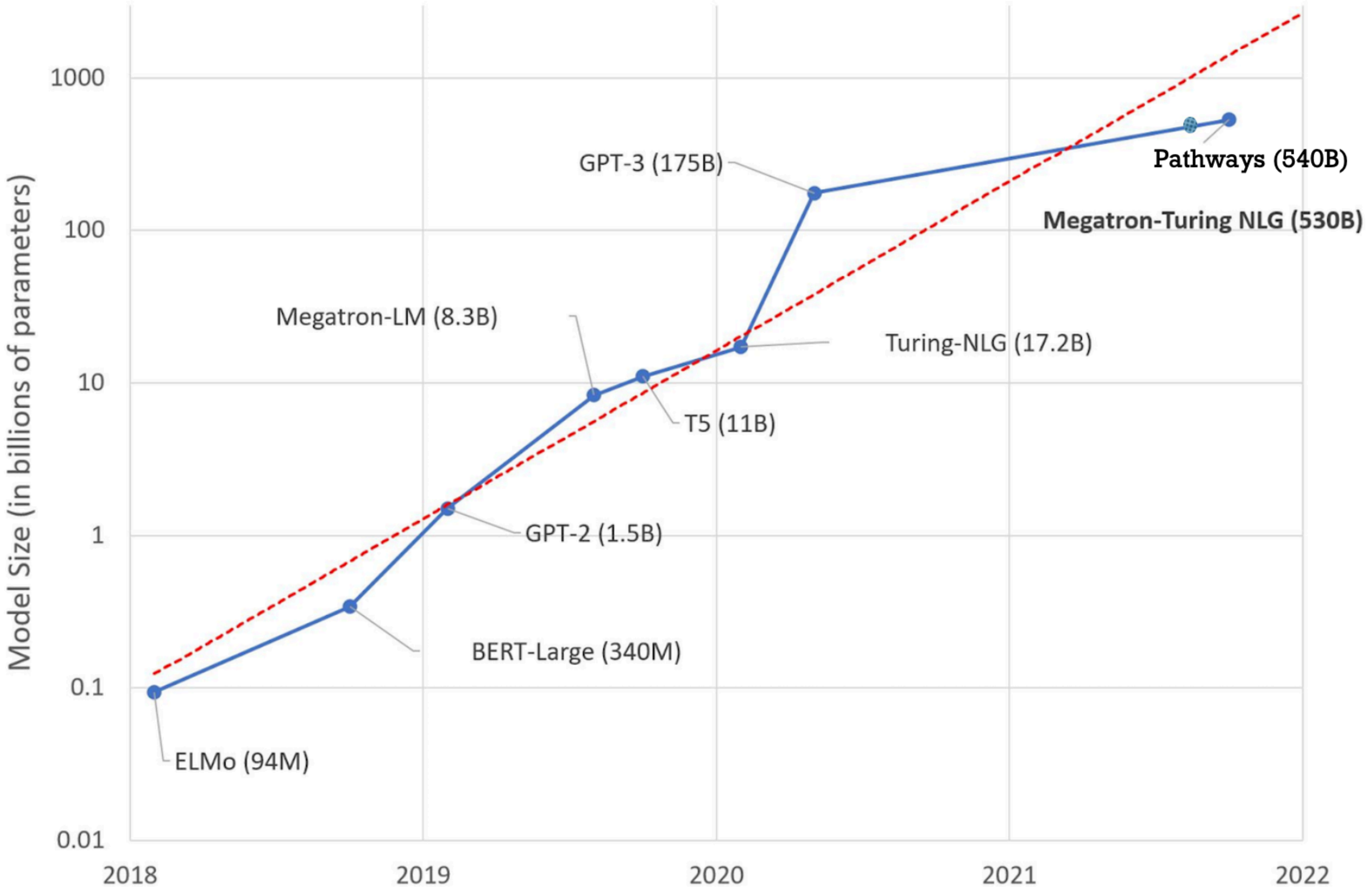
Privacy:

Websites can contain private phone numbers, email addresses, etc.

Skew:

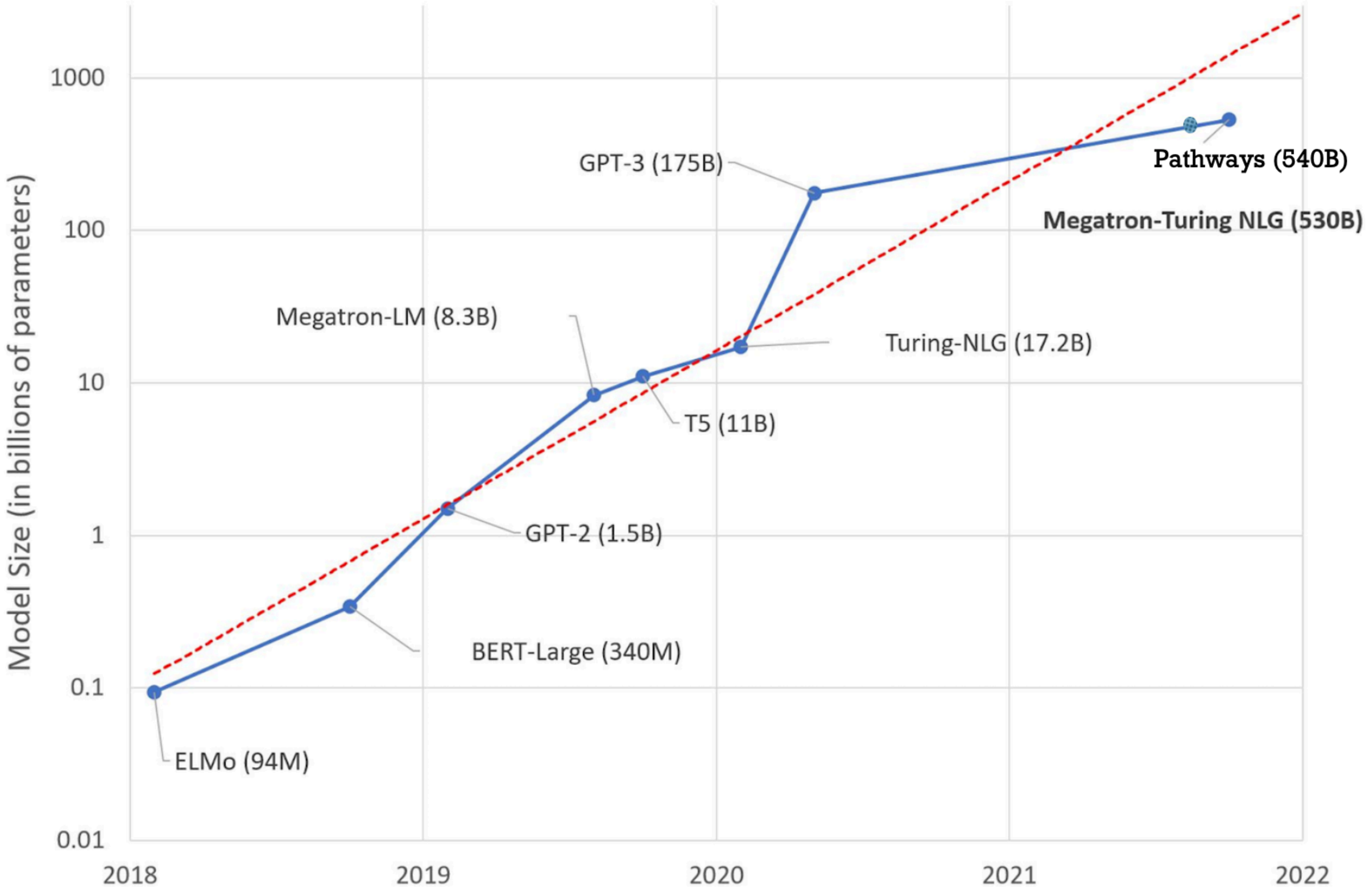
Training data is disproportionately generated by authors from the US, which probably skews resulting topics and opinions

The language model “scaling wars”

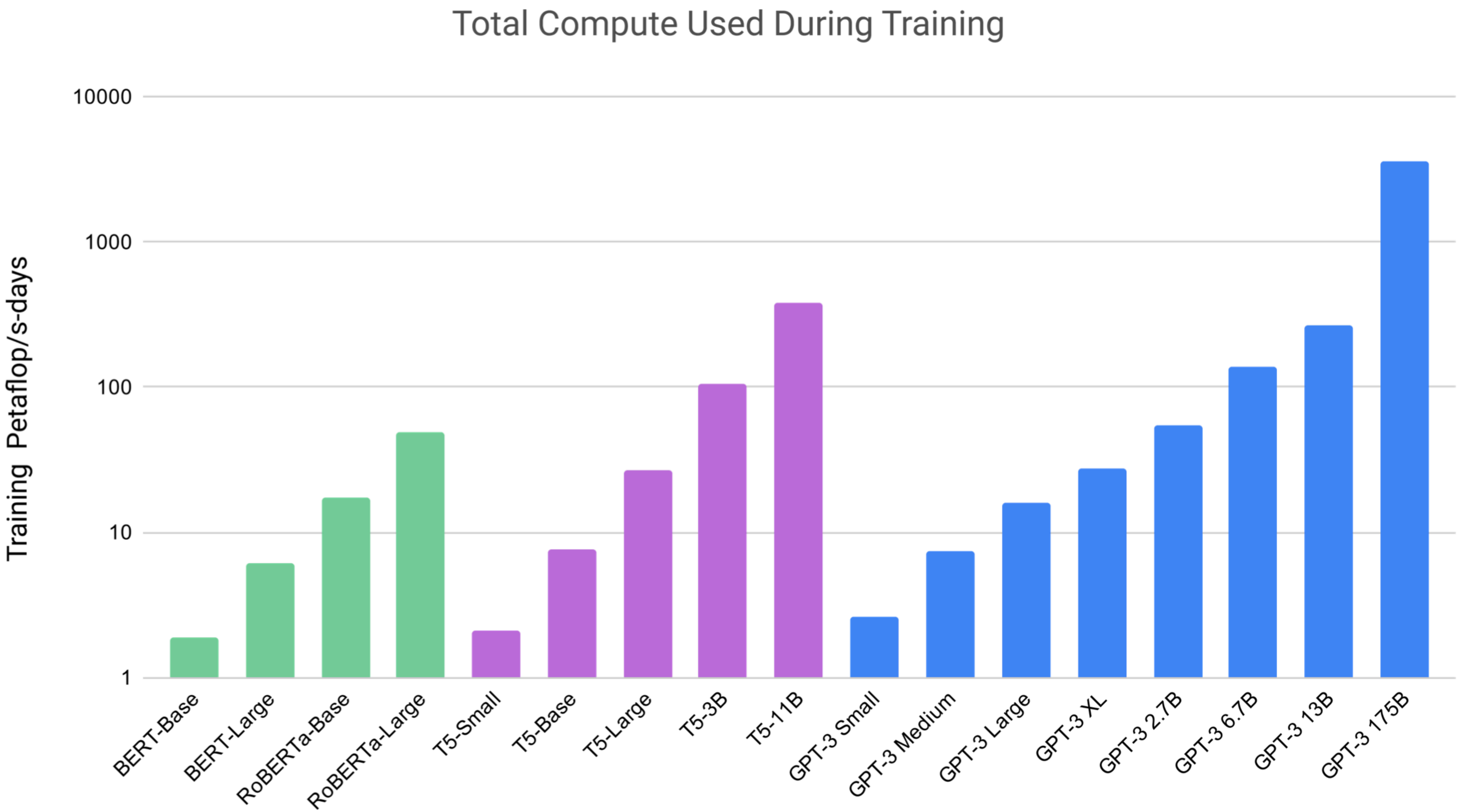


huggingface.co/blog/large-language-models

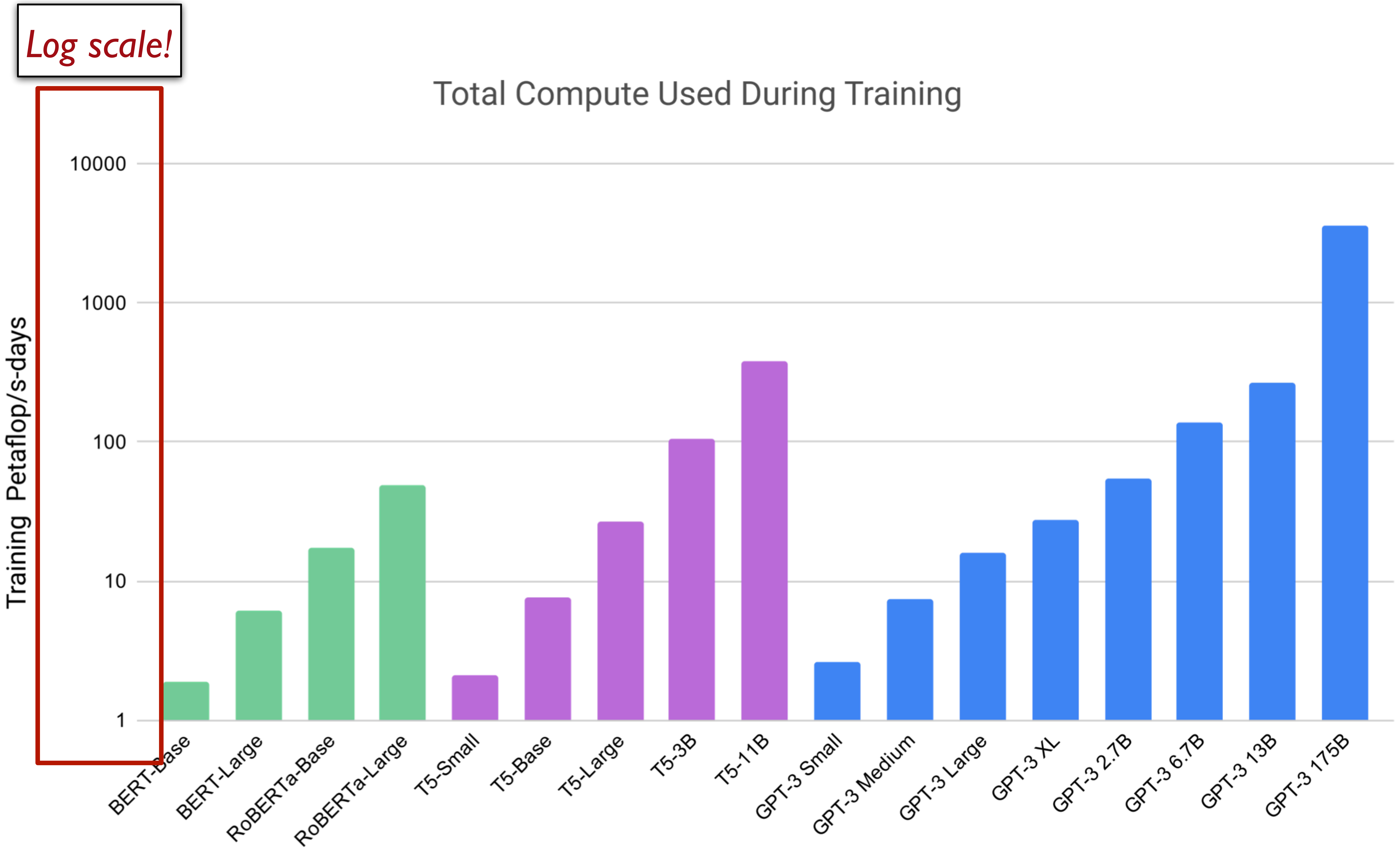
GPT-4: 1.7T params



huggingface.co/blog/large-language-models



Brown et al., 2020



Brown et al., 2020

The training of large language models comes with a significant cost, both in terms of computational resources and environmental impact.

The energy consumption and carbon footprint associated with training these models on massive datasets using powerful hardware has raised concerns about their sustainability and ethical implications.

futurism.com/the-byte/ai-electricity-use-spiking-power-entire-c

10.11.23, 11:57 AM EDT

by VICTOR TANGERMANN

/ ARTIFICIAL INTELLIGENCE

All That Power

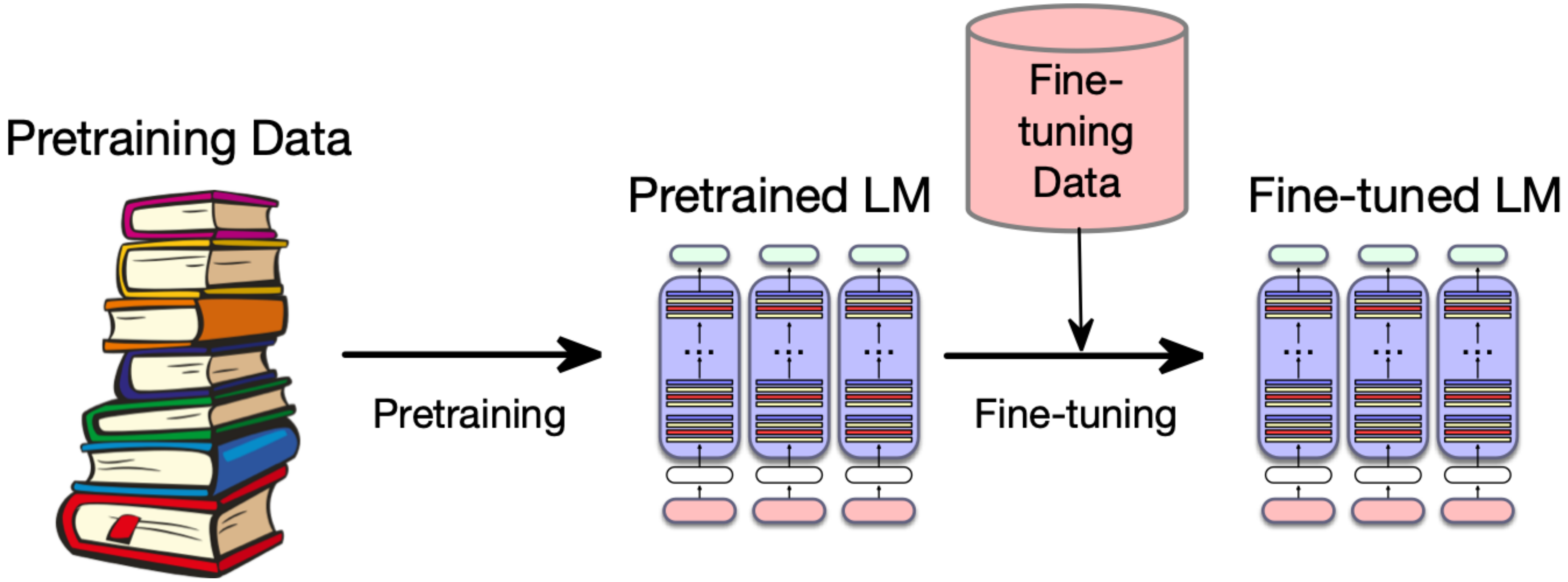
AI chatbots like OpenAI's ChatGPT and Google's Bard consume an astronomical amount of electricity and water — or, more precisely, the massive data centers that power them do.

Finetuning

What happens if we need our LLM to work well on a domain it didn't see in pretraining?

Perhaps some specific medical or legal domain?

Or maybe a multilingual LM needs to see more data on some language that was rare in pretraining?



Finetuning means taking a pretrained model and further adapting some or all of its parameters to some new data.

There are multiple kinds of finetuning. One kind, sometimes called *continued pretraining*, further trains all the parameters of the model on new data

using the same method (word prediction) and loss function (cross-entropy loss) as for pretraining,

as if the new data were just at the end of the pretraining data.

Evaluating large language models

Better LMs are better at predicting text

Recall the chain rule:

$$P(w_{1:n}) = P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_{1:2})\cdots P(w_n \mid w_{1:n-1})$$
$$= \prod_{i=1}^{n} P(w_i \mid w_{<i})$$

So, given a text $w_{1:n}$, we could just compare the log likelihood from two LMs:

$$\text{log likelihood}(w_{1:n}) = \log \prod_{i=1}^{n} P(w_i \mid w_{<i})$$

But raw log-likelihood has a problem:

Probability depends on the size of the test set

- The longer the text, the smaller the probability gets.
- We'd prefer a metric that is per-word, normalized by length.

Perplexity is the inverse probability of the test set, normalized by the number of words.

(The inverse comes from the original definition of perplexity from cross-entropy rate in information theory.)

Probability range is [0, 1]; perplexity range is [1, ∞].

So just as for *n*-gram models, we use perplexity to measure how well the LM predicts unseen text.

The perplexity of a model θ on an unseen test set is *the inverse probability that θ assigns to the test set, normalized by the test set length*.

For a test set of n tokens $w_{1:n}$, the perplexity is:

$$\mathrm{Perplexity}_\theta(w_{1:n}) = P_\theta(w_{1:n})^{-\frac{1}{n}}$$

$$= \sqrt[n]{\frac{1}{P_\theta(w_{1:n})}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P_\theta(w_i \mid w_{<i})}}$$

The higher the probability of the word sequence, the lower the perplexity.

Thus the lower the perplexity of a model on the data, the better the model.

Minimizing perplexity is the same as maximizing probability.

Many other factors that we evaluate, like:

Fairness

Benchmarks measure gendered and racial stereotypes, or decreased performance for language from or about some groups.

Size

Big models take lots of GPUs and time to train, memory to store

Energy usage

Can measure kWh or kilograms of CO_2 emitted

Ethical and safety issues in LLMs

Hallucination

Chatbots May 'Hallucinate' More Often Than Many Realize

What Can You Do When A.I. Lies About You?

People have little protection or recourse when the technology creates and spreads falsehoods about them.

Air Canada loses court case after its chatbot hallucinated fake policies to a customer

The airline argued that the chatbot itself was liable. The court disagreed.

Privacy

Abuse and toxicity

Cleaning Up ChatGPT Takes Heavy Toll on Human Workers

Contractors in Kenya say they were traumatized by effort to screen out descriptions of violence and sexual abuse during run-up to OpenAI's hit chatbot

The New AI-Powered Bing Is Threatening Users.

Lots more

Harm (suggesting dangerous actions)

Fraud

Emotional dependence

Bias

Mary Shelley's Frankenstein

Centered on the problem of creating artificial agents without considering ethical and humanistic concerns.

This is a nice LLM. This is a good LLM. This is a mother's angel.

Acknowledgments

The lecture incorporates material from:

Jurafsky & Martin, *Speech and Language Processing*, 3rd ed. draft