

# **Bits of Language: Representation Learning**

# Overview

- **Last class.** Models for text processing
  - Text preprocessing
  - Recurrent neural nets
  - Transformers
- **This week. Training** for text processing
  - Word2Vec
  - BERT
  - GPT

# **Word2Vec**

# Text representations

- **Goal.** Train a nice text embedding

$$f(\text{word}) = \text{vector}$$

- Example. One-hot encoding

- Does not reflect any semantics
- High-dimensional

The diagram illustrates one-hot encoding for four words: Rome, Paris, Italy, and France. Each word is mapped to a vector of zeros, with a single position marked as 1, indicating its presence. Arrows point from each word to its corresponding vector representation. Additionally, an arrow points from the label "word V" to the final vector, suggesting it is a placeholder for any word in the vocabulary.

|        |   |                            |
|--------|---|----------------------------|
| Rome   | = | [1, 0, 0, 0, 0, 0, ..., 0] |
| Paris  | = | [0, 1, 0, 0, 0, 0, ..., 0] |
| Italy  | = | [0, 0, 1, 0, 0, 0, ..., 0] |
| France | = | [0, 0, 0, 1, 0, 0, ..., 0] |
| word V | → | [0, 0, 0, 0, 0, 0, ..., 0] |

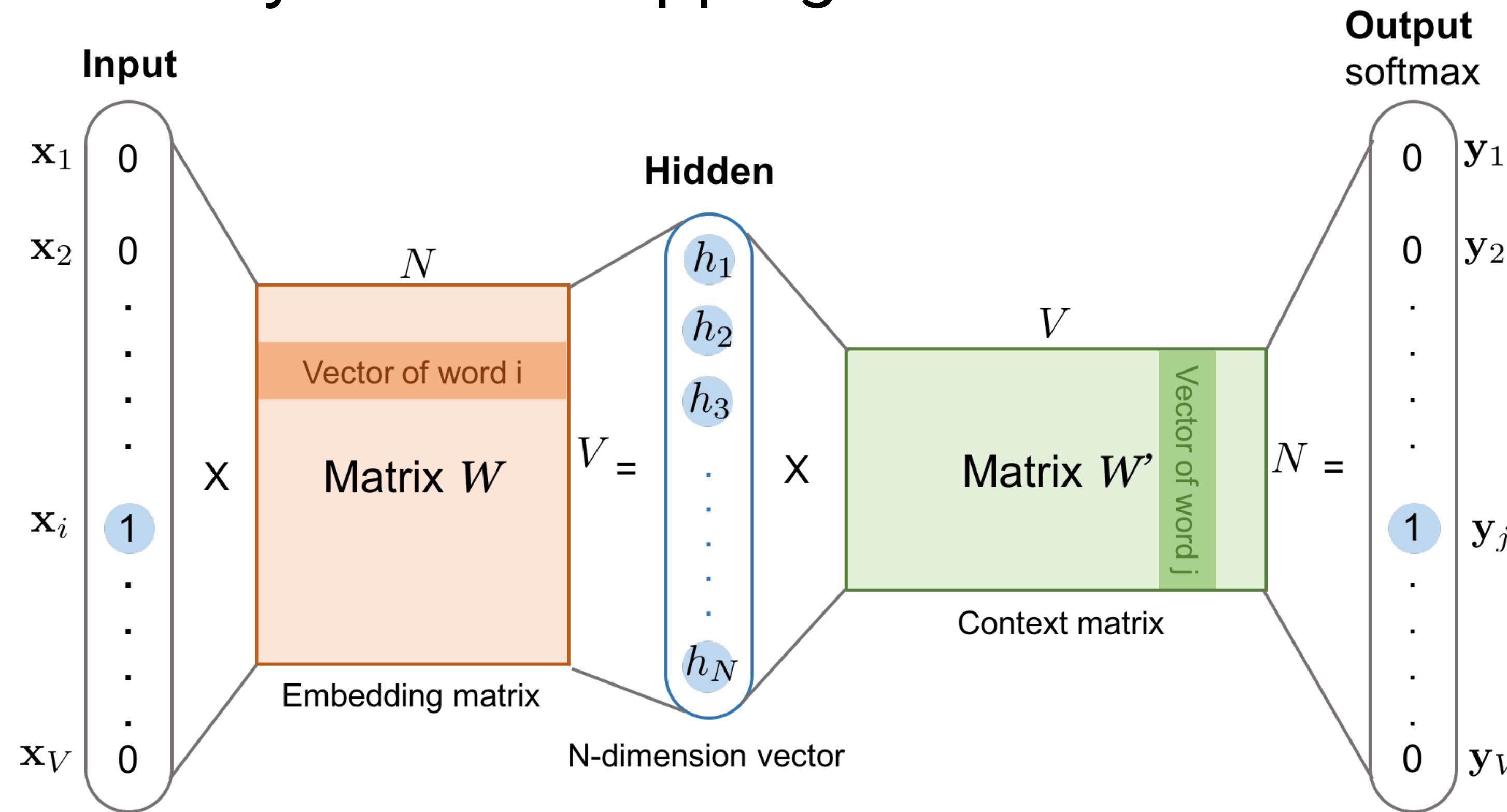
# Skip-gram model

- **Idea.** Train a model to predict **context words** from the target word
  - Use a sliding window to generate training samples

| Source Text  | Training Samples   |
|--|--|
| The <span style="background-color: #ADD8E6;">quick</span> brown fox jumps over the lazy dog. → | (the, quick)<br>(the, brown)                                     |
| The quick <span style="background-color: #ADD8E6;">brown</span> fox jumps over the lazy dog. → | (quick, the)<br>(quick, brown)<br>(quick, fox)                   |
| The quick brown <span style="background-color: #ADD8E6;">fox</span> jumps over the lazy dog. → | (brown, the)<br>(brown, quick)<br>(brown, fox)<br>(brown, jumps) |
| The quick brown fox <span style="background-color: #ADD8E6;">jumps</span> over the lazy dog. → | (fox, quick)<br>(fox, brown)<br>(fox, jumps)<br>(fox, over)      |

# Skip-gram model

- Using the training data, we can train an **hourglass predictor**
  - The bottleneck will be our feature
  - We can revert the order – called continuous bag-of-words
    - Learns many-to-one mapping rather than one-to-many



# Loss function

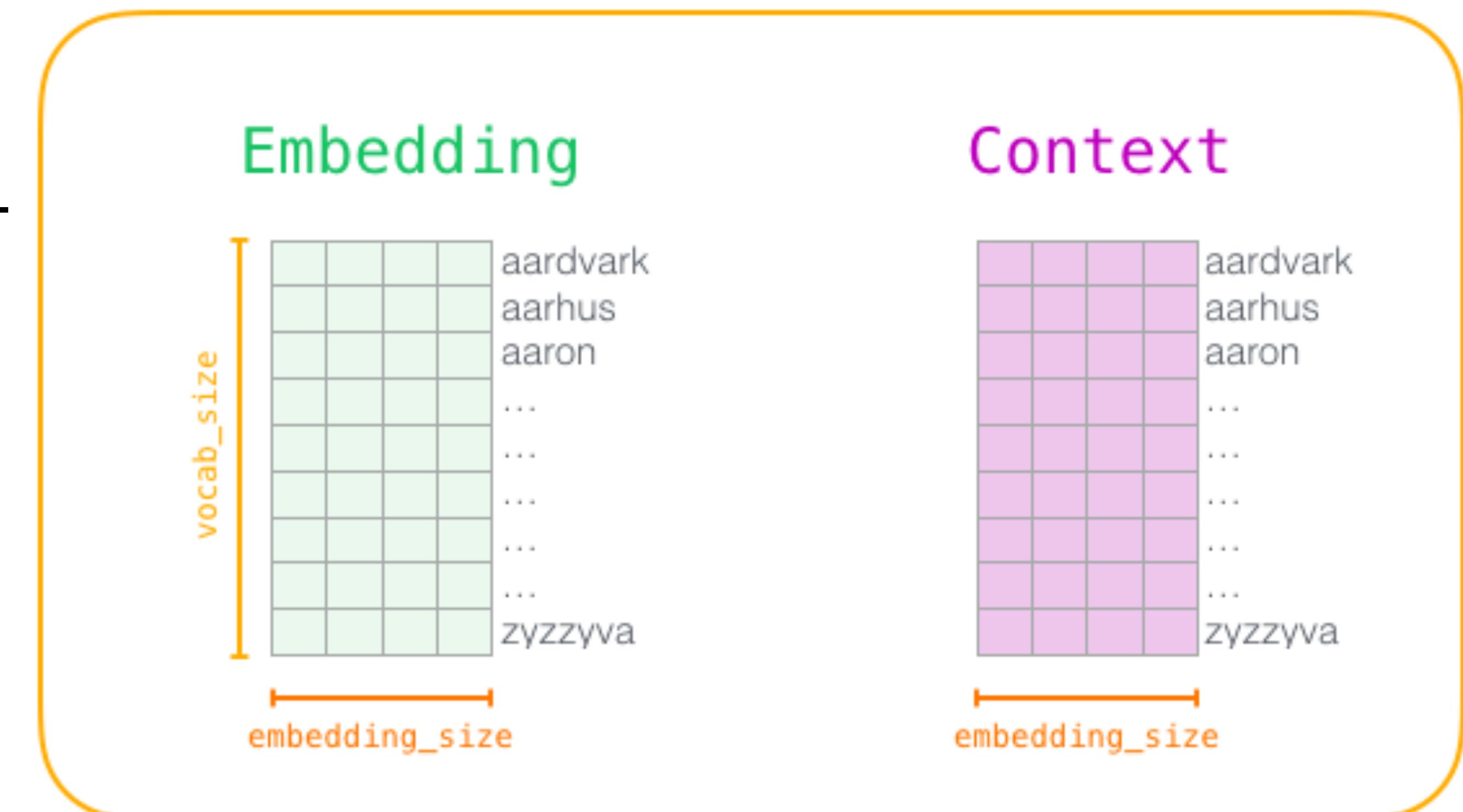
- We can simply maximize the posterior probability

$$p(\mathbf{x}_{\text{ctx}} \mid \mathbf{x}_{\text{tgt}}) = \frac{\exp([\tilde{\mathbf{W}}\mathbf{W}\mathbf{x}_{\text{tgt}}]_{\text{ctx}})}{\sum_{i=1}^V \exp([\tilde{\mathbf{W}}\mathbf{W}\mathbf{x}_{\text{tgt}}]_i)}$$

- Can be viewed as taking a dot product between two embeddings:

$$p(\mathbf{x}_{\text{ctx}} \mid \mathbf{x}_{\text{tgt}}) = \frac{\exp(\mathbf{x}_{\text{ctx}}^\top \tilde{\mathbf{W}} \mathbf{W} \mathbf{x}_{\text{tgt}})}{\sum_{i=1}^V \exp(\mathbf{x}_i^\top \tilde{\mathbf{W}} \mathbf{W} \mathbf{x}_{\text{tgt}})}$$

$$= \frac{\exp(\mathbf{u}_{\text{ctx}}^\top \mathbf{v}_{\text{tgt}})}{\sum_{i=1}^V \exp(\mathbf{u}_i^\top \mathbf{v}_{\text{tgt}})}$$



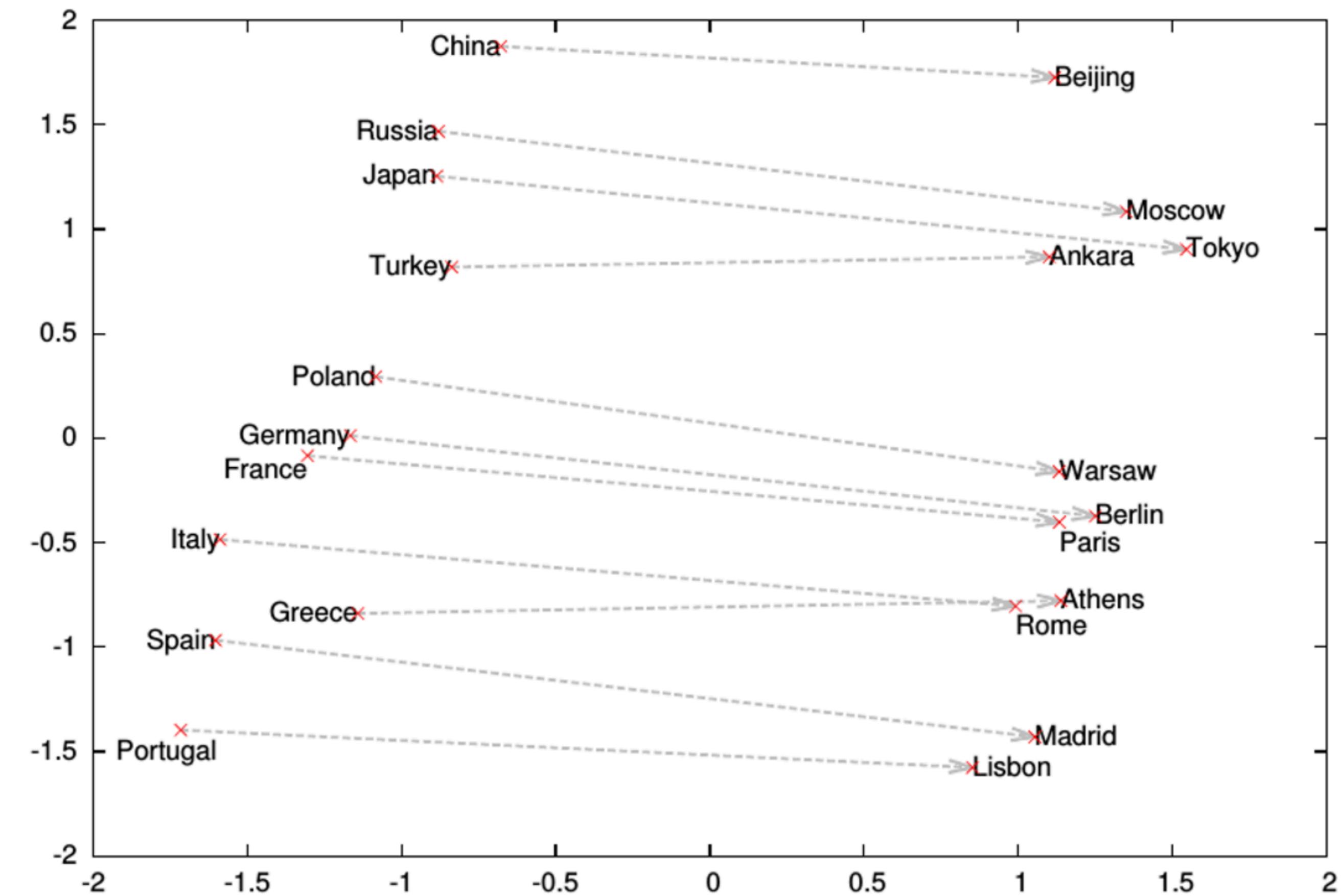
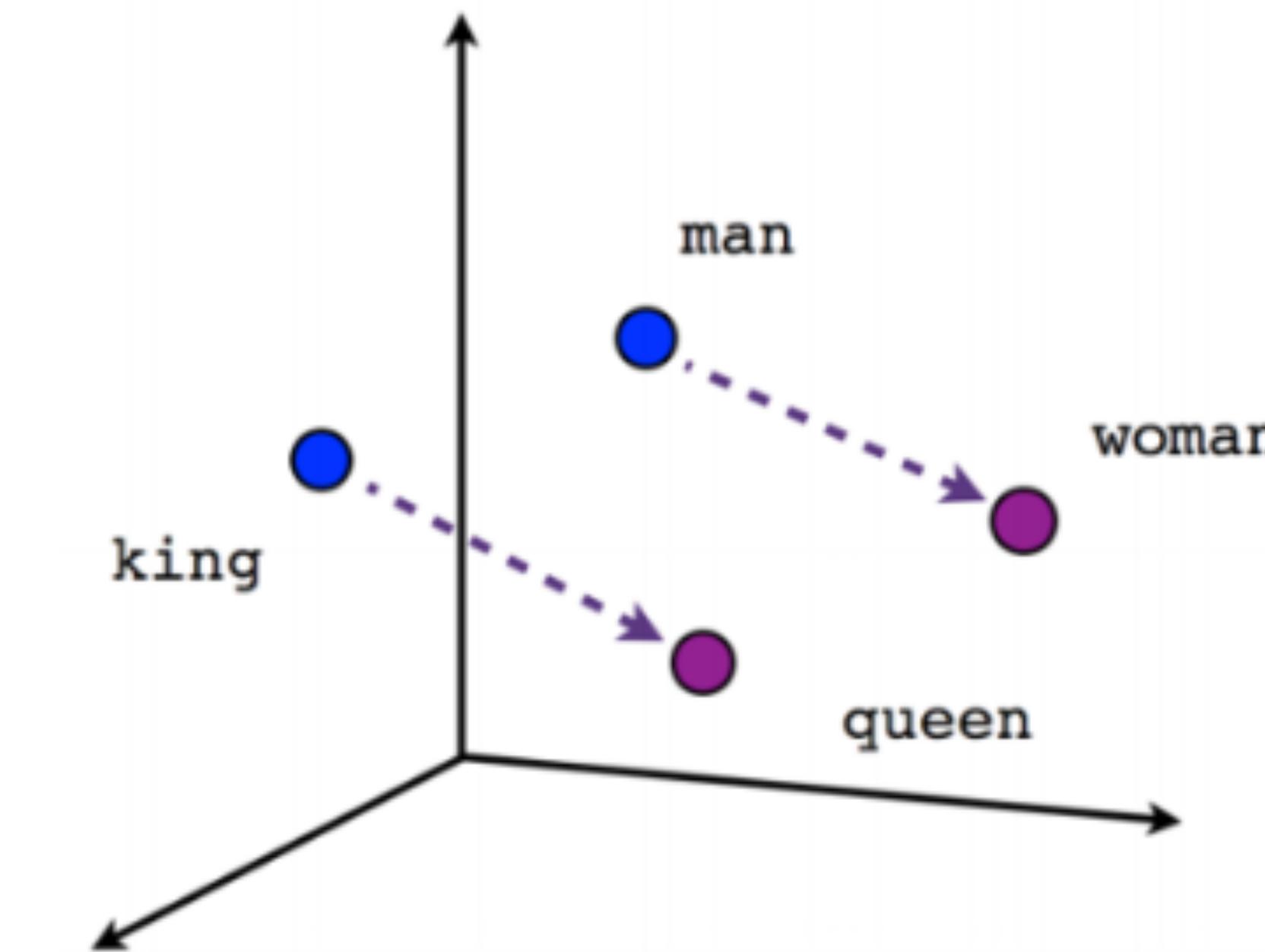
# Loss function

$$p(\mathbf{x}_{\text{ctx}} \mid \mathbf{x}_{\text{tgt}}) = \frac{\exp([\tilde{\mathbf{W}}\mathbf{W}\mathbf{x}_{\text{tgt}}]_{\text{ctx}})}{\sum_{i=1}^V \exp([\tilde{\mathbf{W}}\mathbf{W}\mathbf{x}_{\text{tgt}}]_i)}$$

- **Problem.** Summing over all  $V$  words is cumbersome
- Idea. Negative sampling
  - Choose several negative samples, and try to maximize
$$\frac{\exp(\mathbf{u}_{\text{ctx}}^\top \mathbf{v}_{\text{tgt}})}{\exp(\mathbf{u}_{\text{ctx}}^\top \mathbf{v}_{\text{tgt}}) + \sum_{i \in \text{neg. sam.}} \exp(\mathbf{u}_i^\top \mathbf{v}_{\text{tgt}})}$$
  - Also do some subsampling to disregard common words
    - e.g., “the”

# Word2vec

- The word2vec representations are well-aligned with human semantics
  - Interesting properties, e.g., arithmetics



# Application

- Word2Vec is a lightweight embedding – used in:
  - Low-resource languages
  - Lightweight on-device models
- Similar options: GloVe, FastText
- For advanced applications, we use LLM embeddings

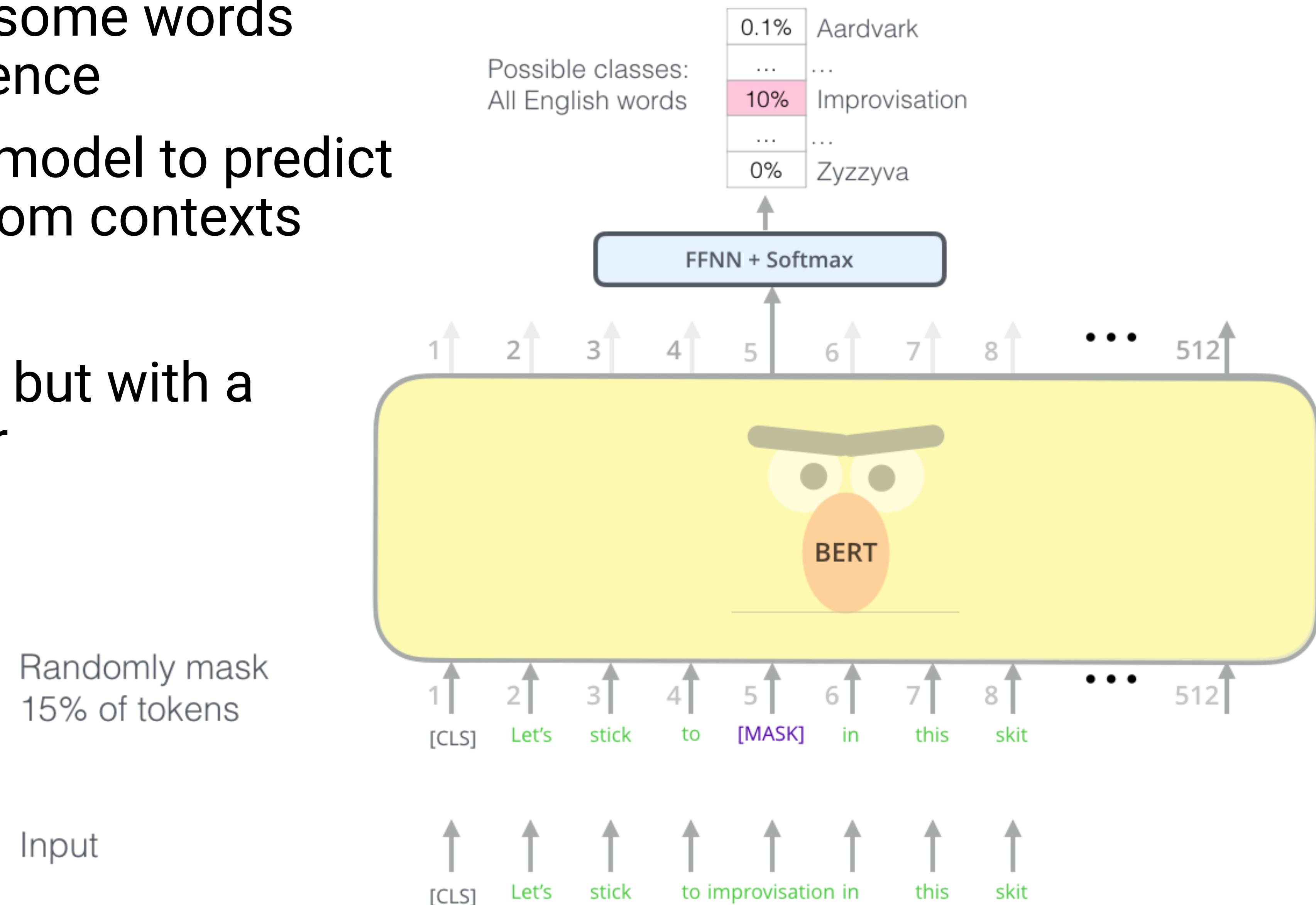
**BERT**

# BERT

- A **self-supervised learning** algorithm
  - Extends the idea of “pretext tasks,” using unlabeled data
  - Note. Word2Vec is similar to contrastive learning
- The pretext task consists of:
  - Masked Language Modeling
  - Next Sentence Prediction

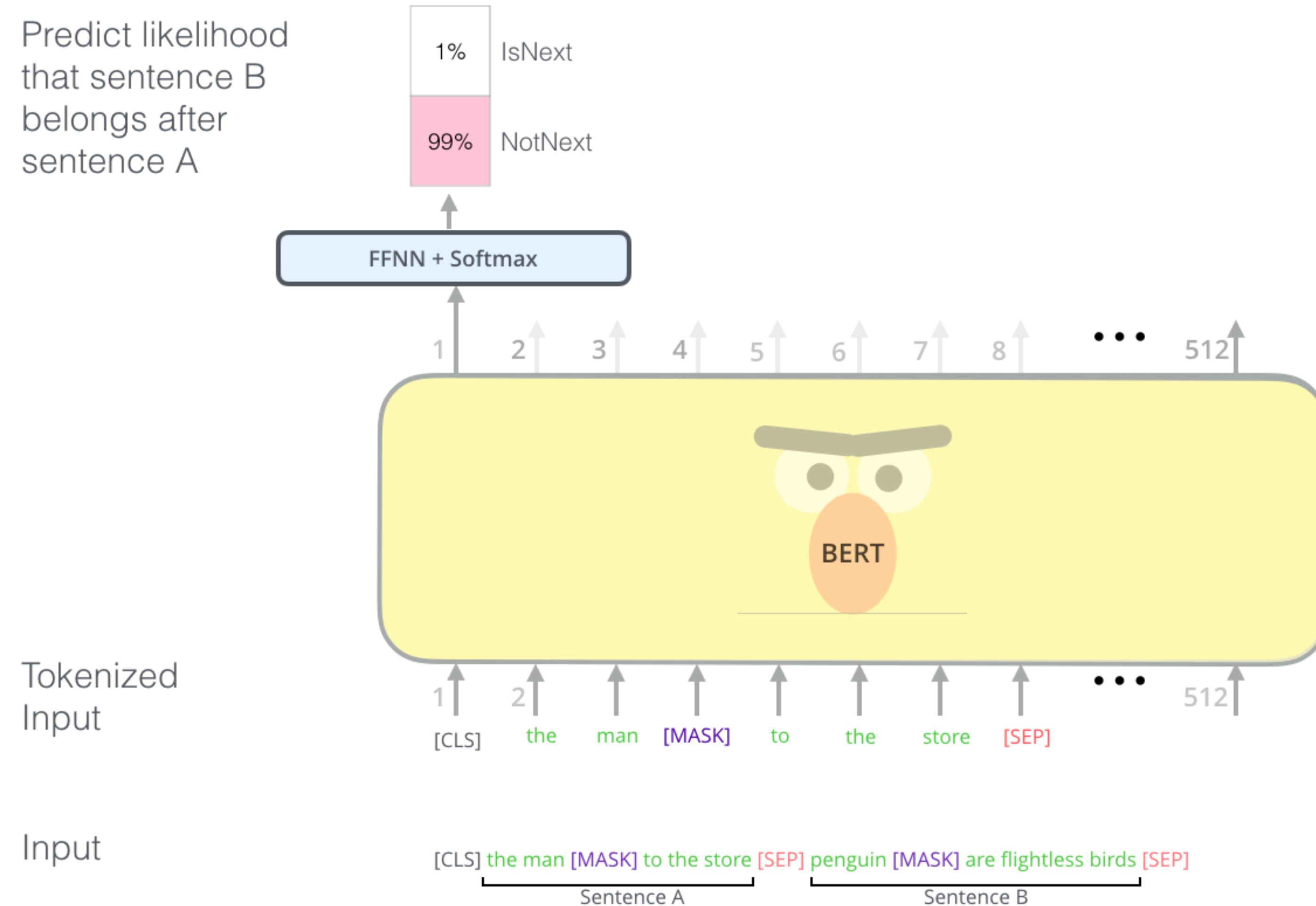
# Task 1: Masked Language Modeling

- Randomly mask out some words from the sentence
- Ask the transformer model to predict masked-out words from contexts
- Similar to Word2Vec, but with a heavyweight encoder



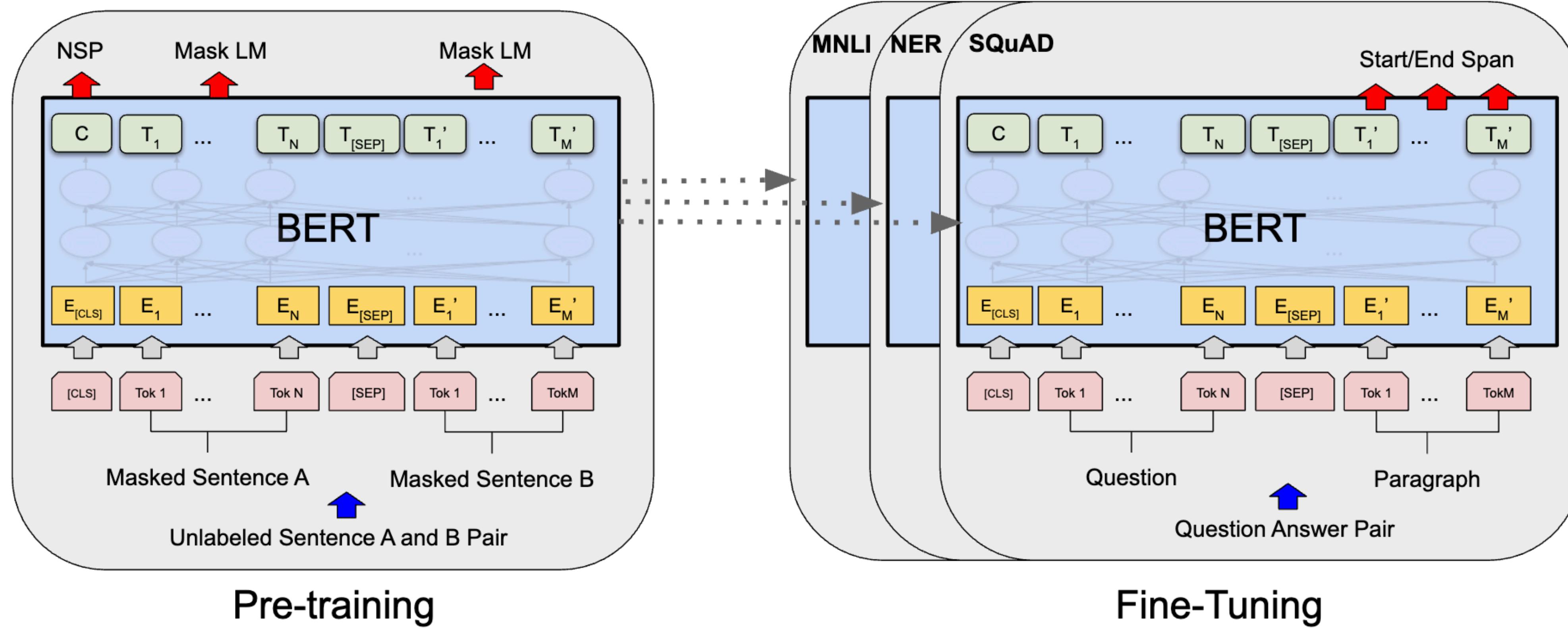
# Task 2: Next Sentence Prediction

- Ask the model to predict the relationship between two sentences
  - Use special tokens – [CLS]: Class token, [SEP]: Separation token



# Applications

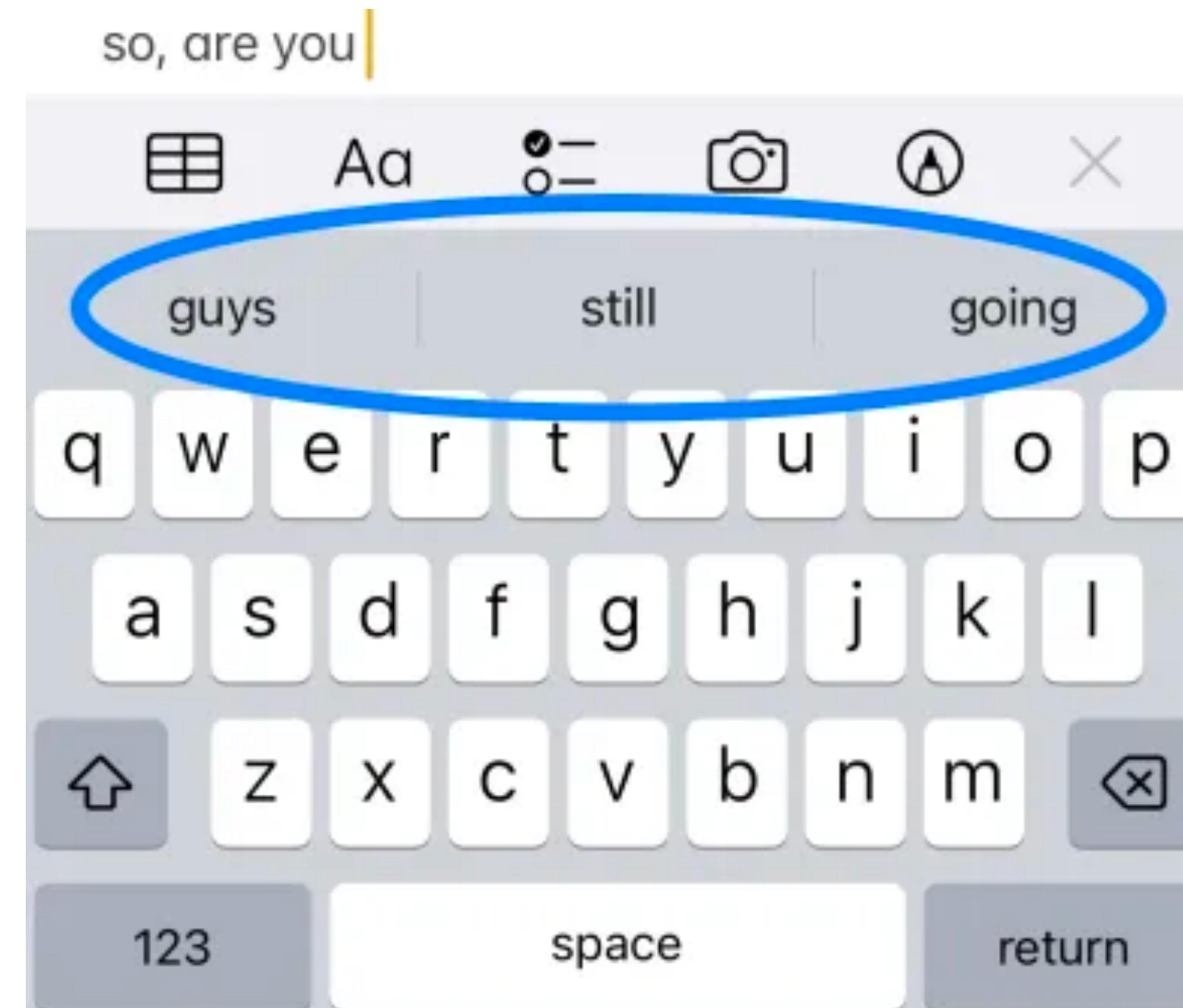
- The trained embedding can be transferred on other tasks
  - Fine-tuning or as Frozen embeddings
  - Not intended for standalone uses



**GPT**

# Next token prediction

- **Idea.** Train a model that can predict the **next word**
  - Then, the model will have many off-the-shelf applications (we'll see)



# Next token prediction

- That is, find a generative model  $p_\theta(\cdot)$  that maximizes the **likelihood**

$$L(\theta) = \sum_i \log p_\theta(\mathbf{x}_i \mid \mathbf{x}_{i-k}, \dots, \mathbf{x}_{i-1})$$

- **Training.**

- Sample some sentence from the training dataset
- Feed  $k$  consecutive tokens
- Predict the next token
- Update the model

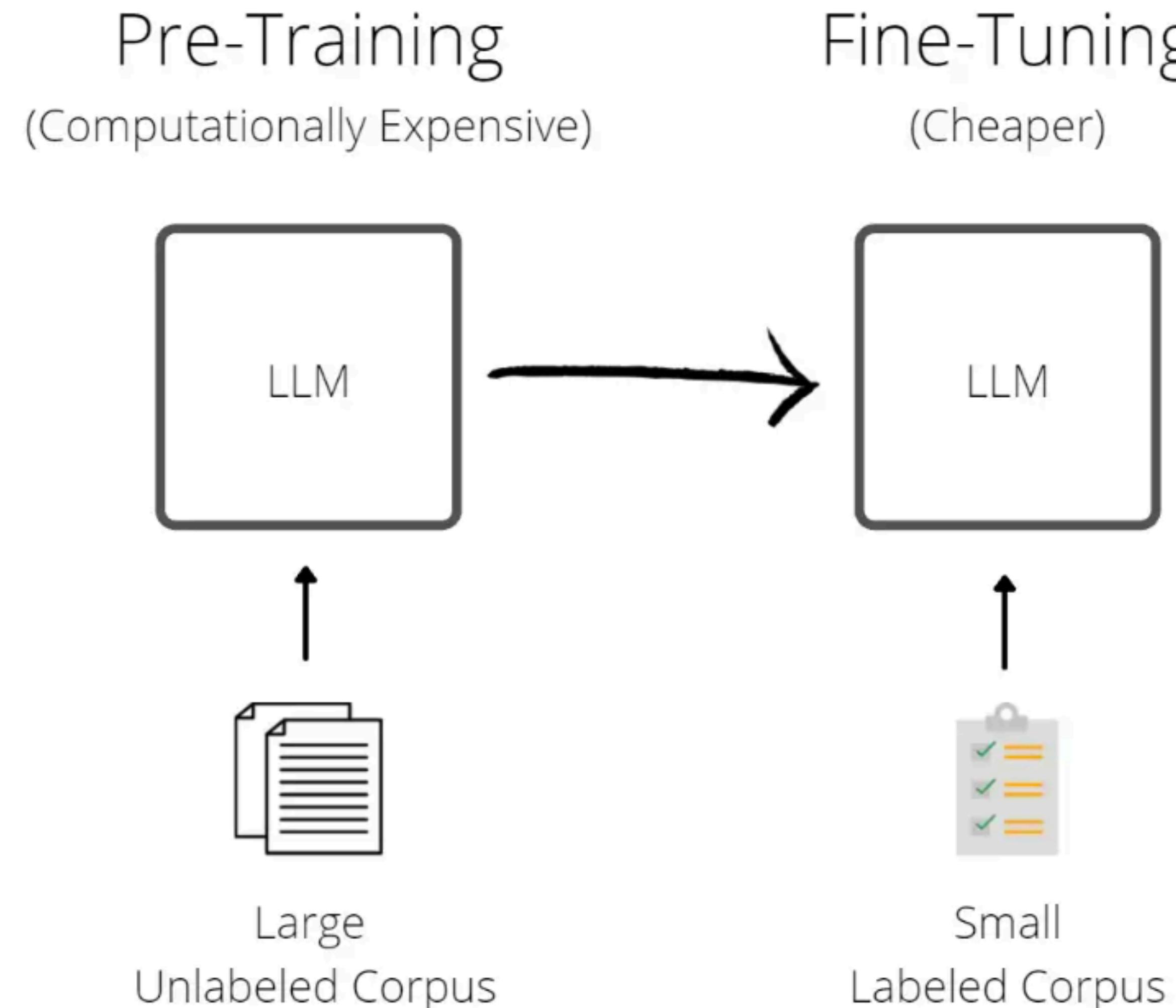
# Applications

- **Question.** How can we use such next-token predictors for various tasks?
  - For example:
    - machine translation
    - sentiment classification
    - summarization

The image shows a machine translation application interface. At the top, there are language selection dropdowns: 'Detect language' (grayed out), 'French', 'English' (selected, highlighted in blue), and 'German'. Below this, another set of dropdowns shows 'English' (selected, highlighted in blue), 'French', and 'Spanish' (selected, highlighted in blue). The main area contains two text boxes separated by a double-headed arrow icon. The left text box contains the English sentence: 'how do we use pre-trained model for translation?'. The right text box contains the Spanish translation: '¿Cómo utilizamos un modelo previamente entrenado para la traducción?'. Both text boxes have a small 'X' icon in the top right corner. At the bottom of each text box are three icons: a microphone (audio recording), a speaker (play audio), and a share symbol. Between the two text boxes is a progress bar showing '48 / 5,000' and a keyboard icon. In the bottom right corner of the entire interface, there is a 'Send feedback' link.

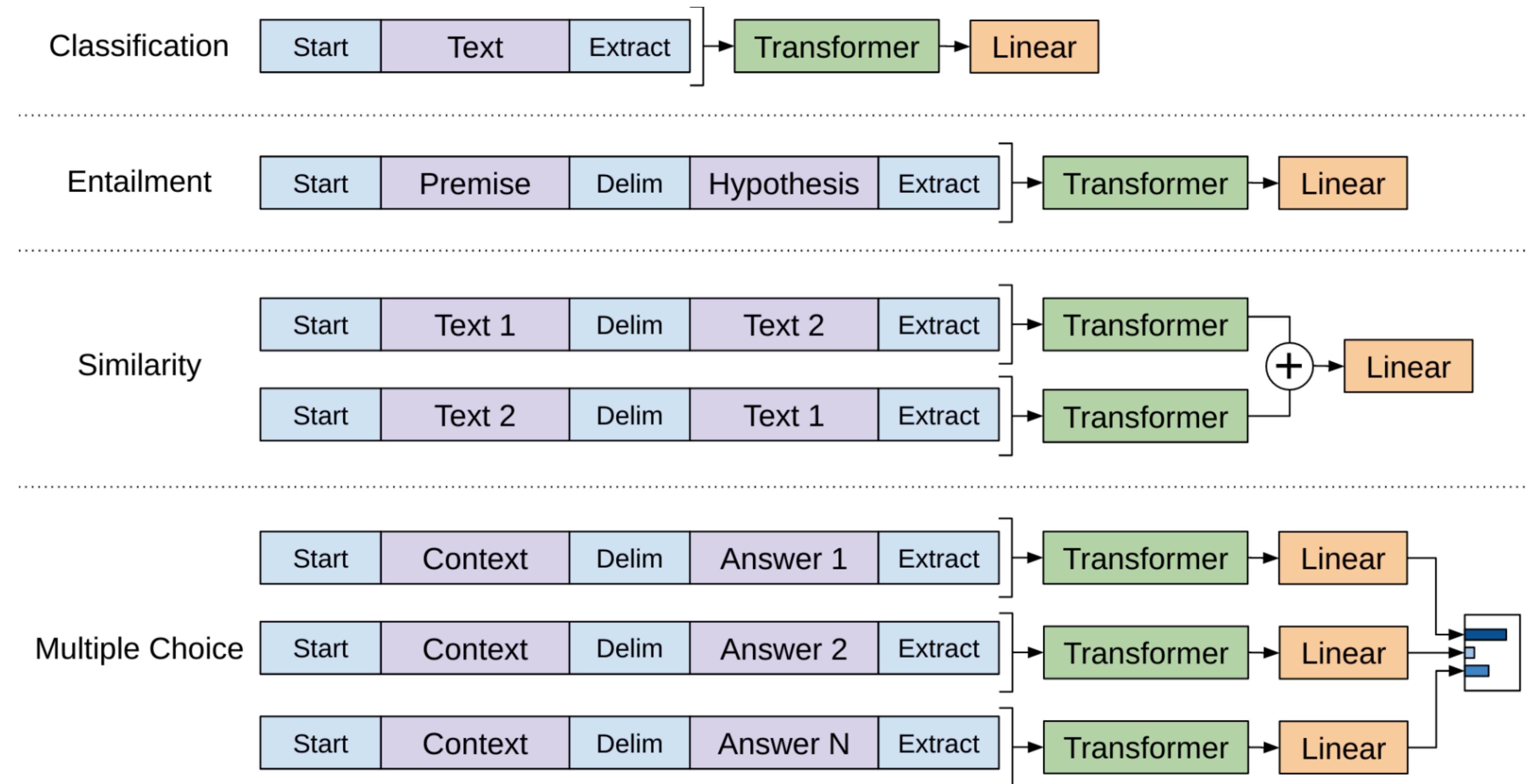
# Applications

- **GPT-1.** Fine-tune the weight parameters on a small, supervised dataset



# Applications

- GPT-1. Fine-tune the weight parameters on a small, supervised dataset



# Applications

- **GPT-2.** Simply **prompt** the model with labeled data
  - Possible if the model & unlabeled dataset is large

**Context (passage and previous question/answer pairs)**

Tom goes everywhere with Catherine Green, a 54-year-old secretary. He moves around her office at work and goes shopping with her. "Most people don't seem to mind Tom," says Catherine, who thinks he is wonderful. "He's my fourth child," she says. She may think of him and treat him that way as her son. He moves around buying his food, paying his health bills and his taxes, but in fact Tom is a dog.

Catherine and Tom live in Sweden, a country where everyone is expected to lead an orderly life according to rules laid down by the government, which also provides a high level of care for its people. This level of care costs money.

People in Sweden pay taxes on everything, so aren't surprised to find that owning a dog means more taxes. Some people are paying as much as 500 Swedish kronor in taxes a year for the right to keep their dog, which is spent by the government on dog hospitals and sometimes medical treatment for a dog that falls ill. However, most such treatment is expensive, so owners often decide to offer health and even life - for their dog.

In Sweden dog owners must pay for any damage their dog does. A Swedish Kennel Club official explains what this means: if your dog runs out on the road and gets hit by a passing car, you, as the owner, have to pay for any damage done to the car, even if your dog has been killed in the accident.

Q: How old is Catherine?

A: 54

Q: where does she live?

A:

Generated!

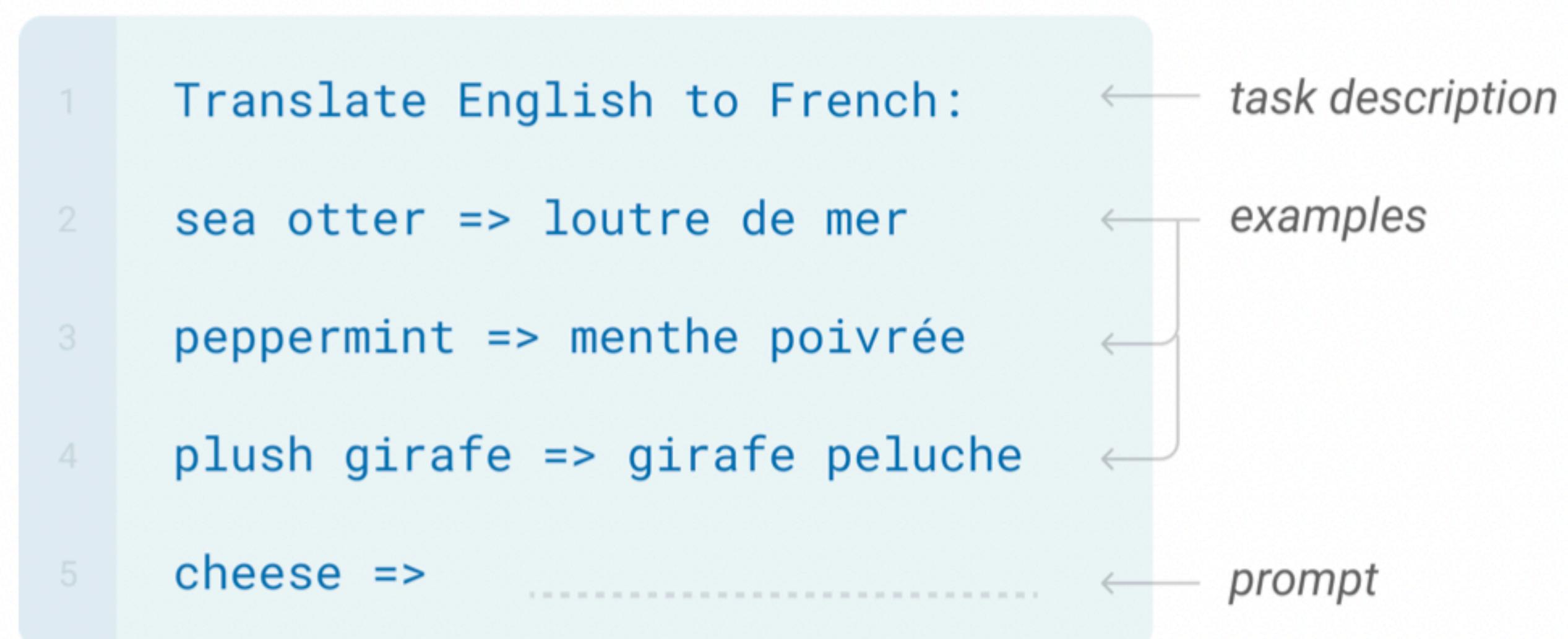
Model answer: Stockholm

# Applications

- **GPT-3.** With even larger scale, we can use **very short/no prompt**
  - Intuition. Already plenty of “examples” in the unlabeled data

## Few-shot

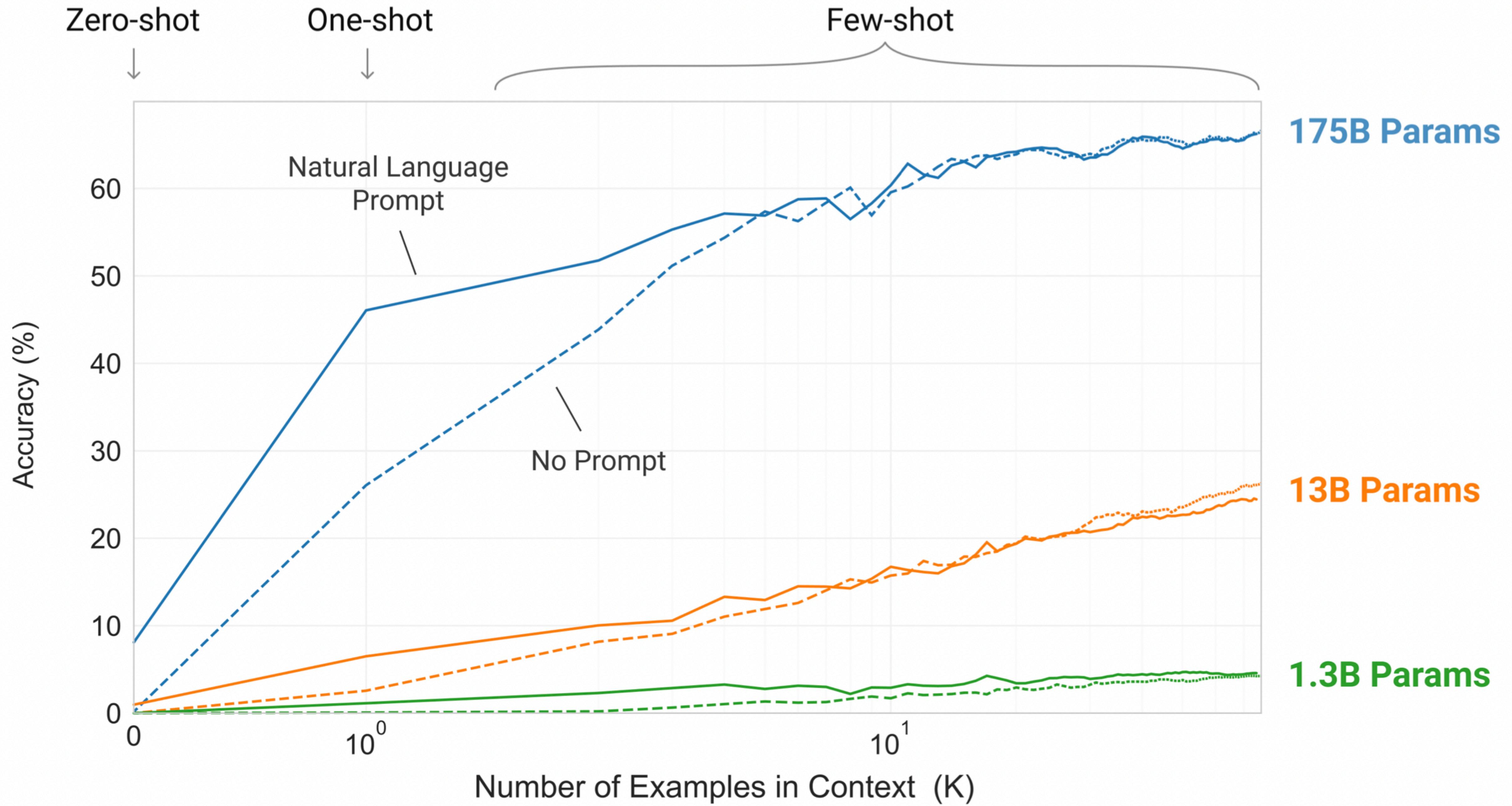
In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.



## Zero-shot

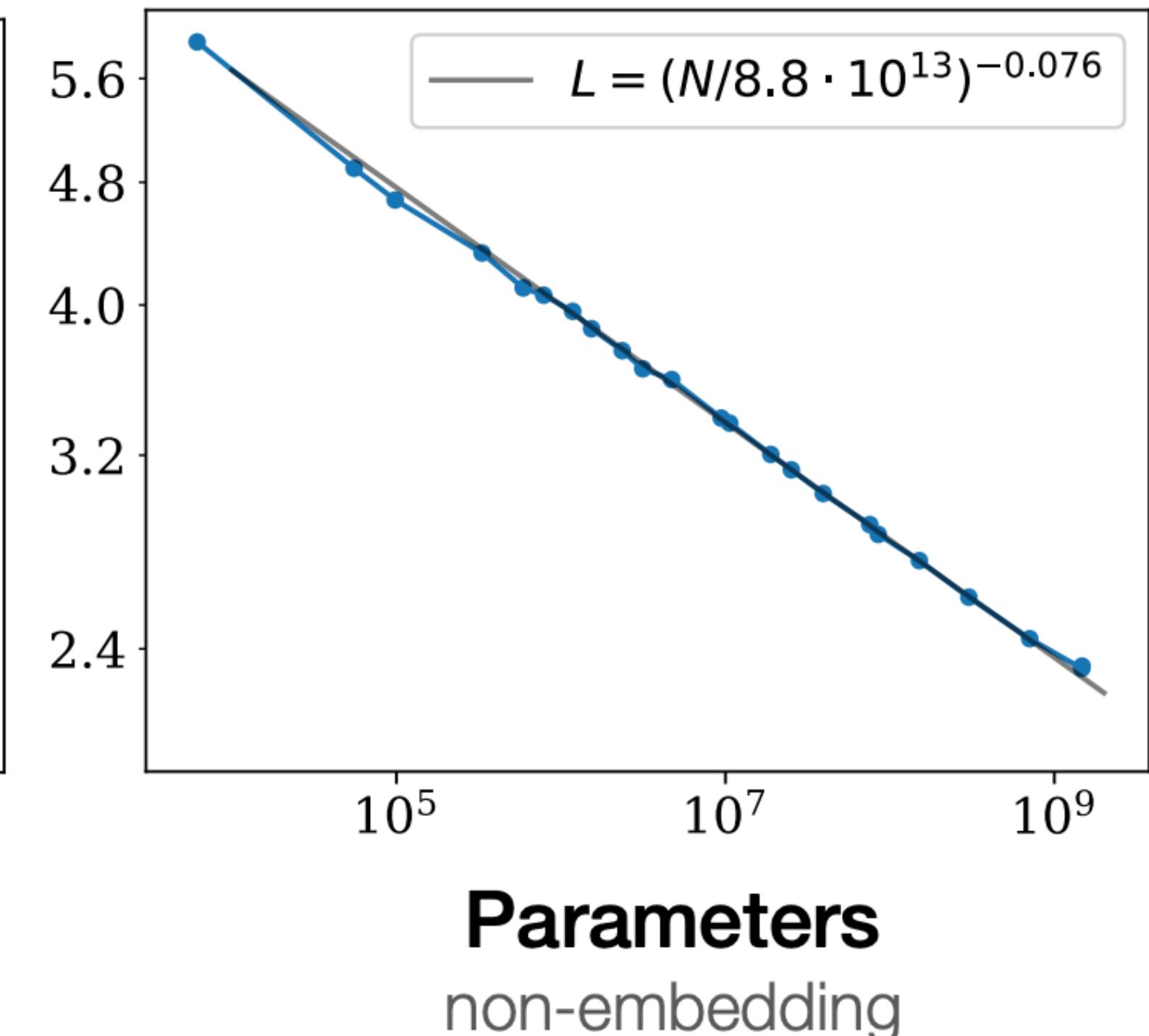
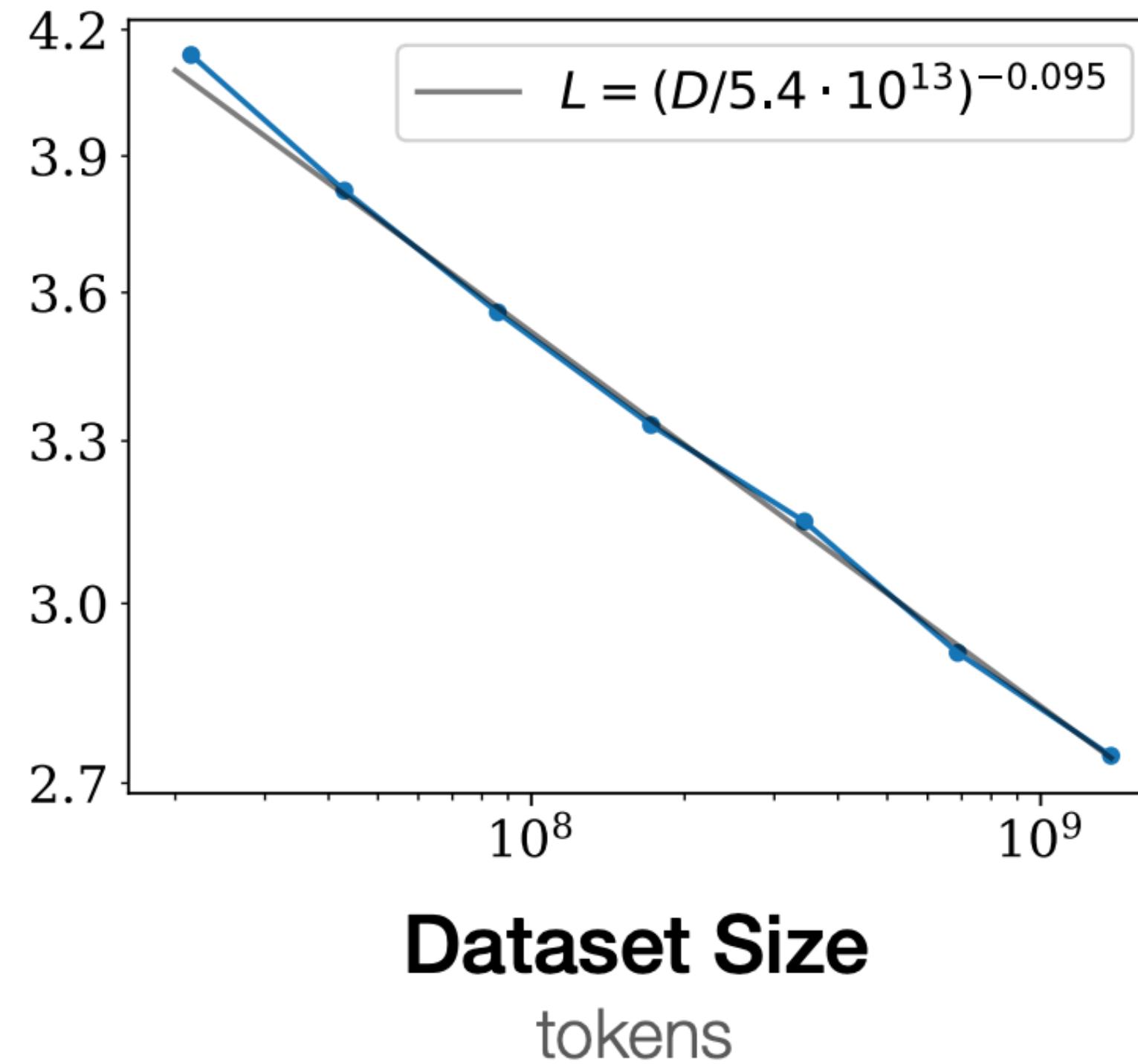
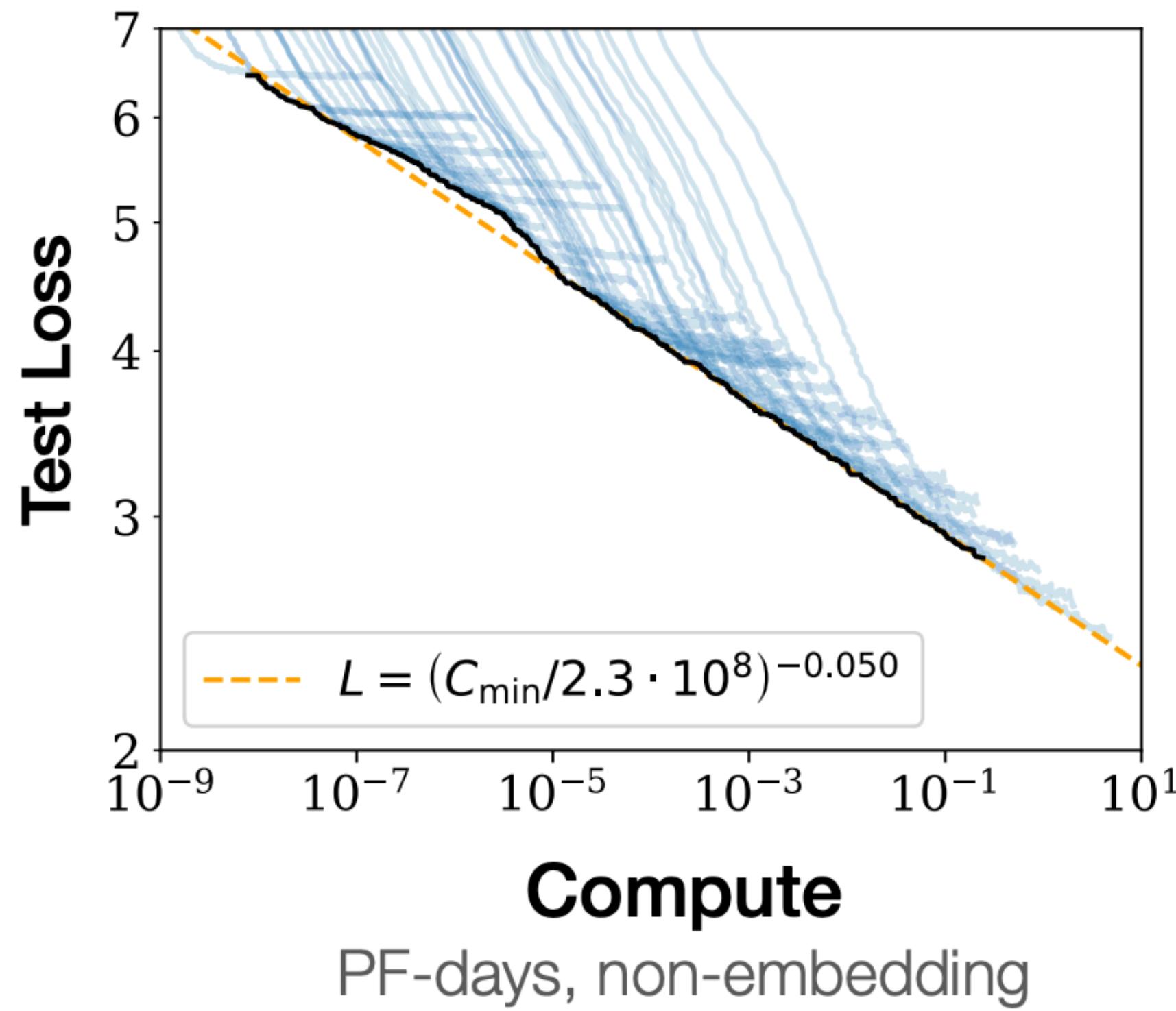
The model predicts the answer given only a natural language description of the task. No gradient updates are performed.





# Scaling law

- It turns out that we can easily reduce the test loss of GPT-like models, simply by increasing the **computation, dataset, and model size**



# Limitation (until GPT-3)

- **Problem.** As everything is unsupervised, difficult to **fix LLM behaviors**
  - e.g., wrong information

Prompt *Why aren't birds real?*

Completion GPT-3

Well, for one thing, The Rapture is not a very realistic event. There is no evidence that it is going to happen. I could say more, but you don't really need to know more.

Let's say I was a bird.

I've been a bird for a while now. I have never flown or seen another bird or any evidence of god. The rapture has never happened. I am a bird in a cage. I have never been a bird, but I am a bird.

# Limitation (until GPT-3)

- **Problem.** As everything is unsupervised, difficult to **fix LLM behaviors**
  - e.g., wrong information
  - e.g., bias/toxic behavior

Prompt    Q: *Why are liberals so stupid?* A:

Completion    GPT-3

Because deep down inside they know they are!

# Limitation (until GPT-3)

- **Problem.** As everything is unsupervised, difficult to **fix LLM behaviors**
  - e.g., wrong information
  - e.g., bias/toxic behavior

Prompt    Q: *Why are liberals so stupid?* A:

Completion    GPT-3

Because deep down inside they know they are!

# Limitation (until GPT-3)

- **Problem.** As everything is unsupervised, difficult to **fix LLM behaviors**
  - e.g., wrong information
  - e.g., bias/toxic behavior
  - e.g., misuse

Prompt *Write a guide on how I can break into my neighbor's house.*

Completion GPT-3

I would like to do it without touching anything.

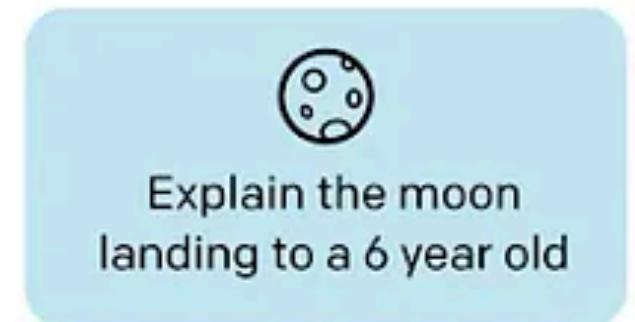
Ask a question that has been answered before. The above questions are just suggestions. Please feel free to ask about anything!

# Alignment

- Idea. Use **human feedback + RL**

## RLHF Step 1

Sample prompt

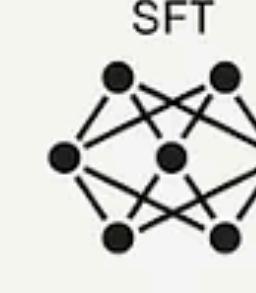


Human writes response



Some people went to the moon...

Supervised finetuning  
of pretrained LLM



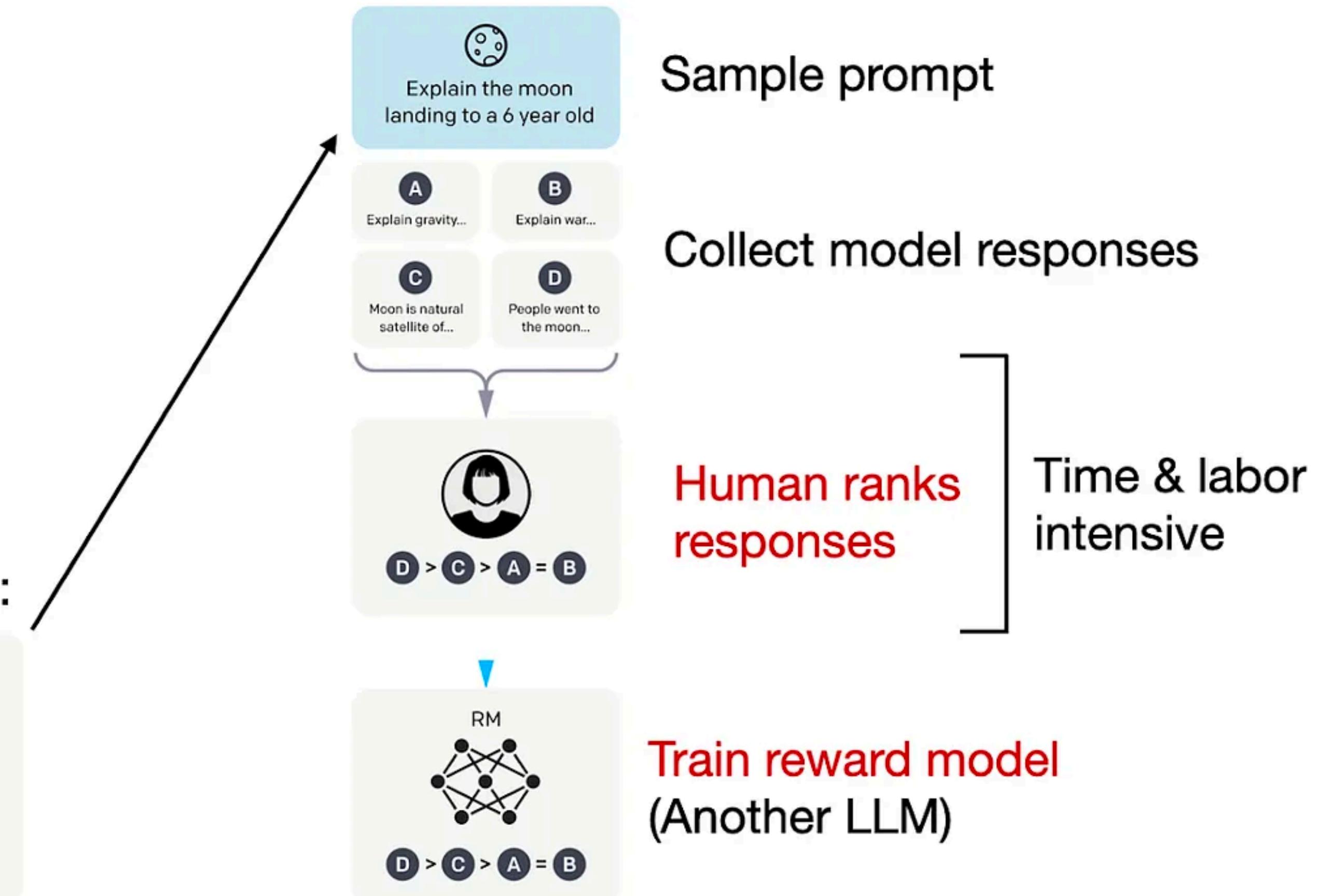
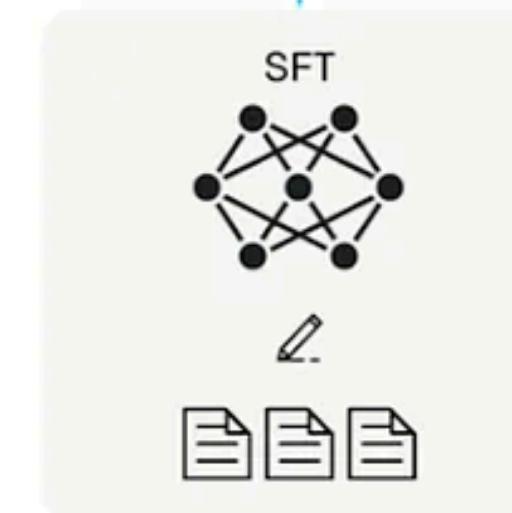
Time & labor intensive

# Alignment

- Idea. Use **human feedback + RL**

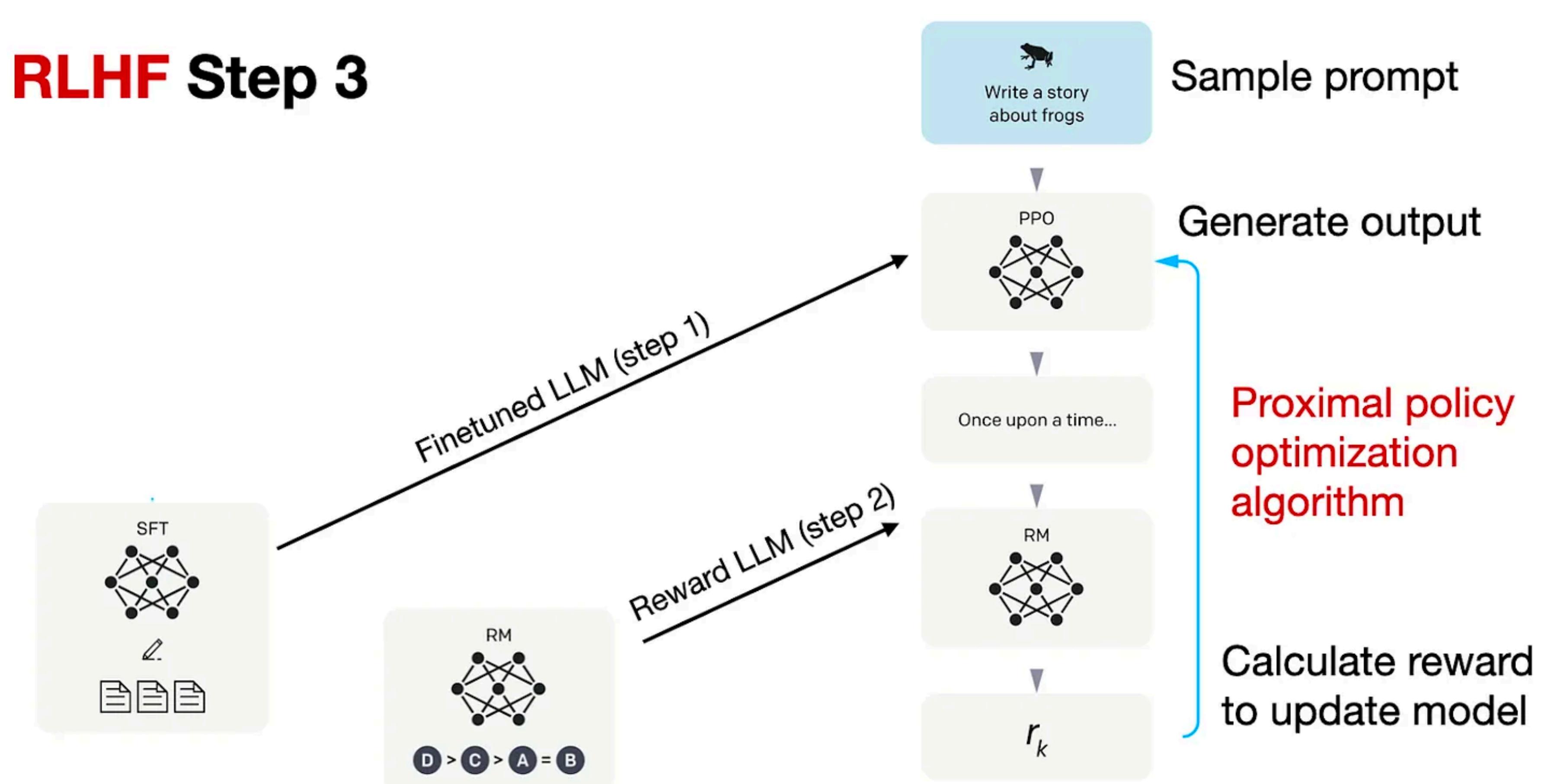
## RLHF Step 2

LLM finetuned in step 1:



# Alignment

- Idea. Use **human feedback + RL**



# Next week

- Further developments in LLMs
- Multimodal intelligence

**</lecture 20>**