Quiz instructions

Get a blue or black pen out now, before we hand out the quiz.

Do not talk or use electronic devices once we start handing out the quiz.

Leave the quiz facing down until we say it is time to start.

There are two versions of the quiz. The people either side of you must have a different coloured quiz to you.

When we say it is time to stop, hand your quiz to the end of your row.

Make sure you write your name and SID.

This is a closed-book, closed-note quiz. No electronic devices may be used in any way.

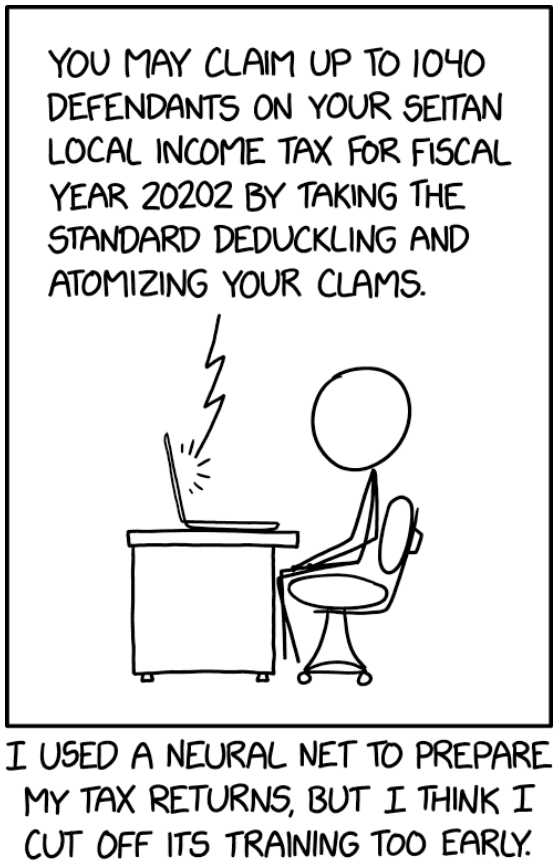
Note your responses by completely filling in the relevant circle(s) an square(s): ●

If you make a mistake, put an X over the filled in circle / square: ⊗

COMP 4446 / 5046
Lecture 9: Models – Large Language Models

Jonathan K. Kummerfeld

Semester 1, 2024



THE UNIVERSITY OF SYDNEY

menti.com 1561 6341

[I ended up getting my tax return prepared at a local place by a really friendly pretrained neural net named Greg.]

Source: https://xkcd.com/2265

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Feedback survey frequent comments

Lectures
- Move slower, with more detail on each concept and examples
- More text, e.g., recap slides

Quiz
- Time is a challenge
- Concerns about cheating
- Adjust coding answer format
- MCQs can have multiple answers

Workshops
- I like my tutor
- Hard to complete in the time
- Introduce more discussion

Assignments
- Using PyTorch in them
- More detail in instructions
- Hints on test cases

Reminders:
- No-builds versions of slides
- Extra materials on Canvas
- Solutions to workshops and assignments are on Ed

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

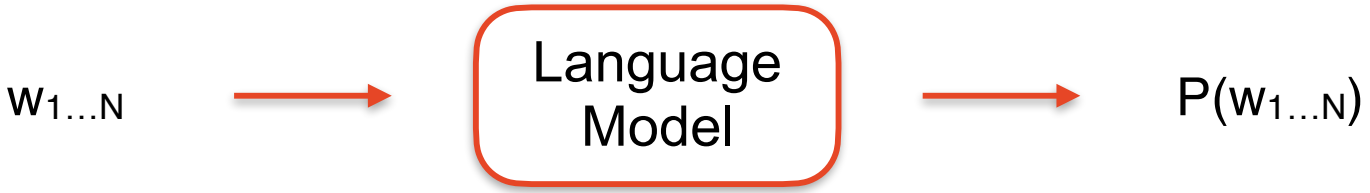
Language Models

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Language models assign a probability to a string



For example:

P("We are at The University of Sydney")

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can use this to predict the next word

Input = $w_{1...N}$

Next word = argmax$_{t \text{ in words}}$ $p(w_{1...N}, t)$

Language Model

For example:

Input = “We are at The University of”

P(“We are at The University of Sydney”) >
P(“We are at The University of chocolate”)
P(“We are at The University of Joe”)
P(“We are at The University of running”)
…

“Sydney” is the argmax

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

N-Gram language models estimate this probability based on counts of word sequences

Count(of Sydney) = How often it occurs in some data

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

How do we calculate the score / probability?

Idea 1 - Use a big lookup table

$$P(w_{1\ldots7})$$

7 words P

Too many options! |V| * |V| * |V| * |V| * |V| * |V| * |V| = $|V|^7$

Idea 2 - Use the chain rule to break up the calculation

$$P(w_{1\ldots7}) = \frac{P(w_{1\ldots7})}{P(w_{1\ldots6})} P(w_{1\ldots6})$$

$$= P(w_7 \mid w_{1\ldots6})\ P(w_{1\ldots6})$$

$$= P(w_7 \mid w_{1\ldots6})\ P(w_6 \mid w_{1\ldots5})\ P(w_5 \mid w_{1\ldots4})\ P(w_4 \mid w_{1\ldots3}) \ldots$$

Still too many options!

Idea 3 - Use an approximation

$$P(w_{1\ldots7}) = P(w_7 \mid w_{1\ldots6})\ P(w_6 \mid w_{1\ldots5})\ P(w_5 \mid w_{1\ldots4})\ P(w_4 \mid w_{1\ldots3}) \ldots$$

$$= P(w_7 \mid w_6)\ P(w_6 \mid w_5)\ P(w_5 \mid w_4)\ P(w_4 \mid w_3) \ldots$$

Markov? ✓

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

This approximation (a Markov assumption) is very strong

P("We are at The University of Sydney")
~ P(Sydney | of) * P(of | University) * P(University | The) …

Markov Assumption with n = 2

Also called a *bigram model*

Markov assumption:
The future is independent of the past given the present

From Ancient Greek suffix -γραμμα (-gramma),
from γράμμα (grámma, "written character, letter, that which is drawn"),
from γράφω (gráphō, "to scratch, to scrape, to graze").

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Why is it a strong assumption?

Surprising matching scores: by using markov

1 P("I thought of a story, wrote the story, and published the story")

=

P("I thought of a story, and published the story, wrote the story")

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Why is it a strong assumption?

Surprising matching scores:

P("I thought of a story, *wrote the story*, and published the story")

=

P("I thought of a story, and published the story, *wrote the story*")

Cannot capture long-distance dependencies

People love to eat with **their** hands 

I love to eat with **my** hands 

People love to eat with **my** hands 

I love to eat with **their** hands 

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can make a less strong assumption

Markov Assumption with n = 2 (*bigram model*)

P("We are at The University of Sydney")
~ P(Sydney | *of*) * P(of | University) …

Markov Assumption with n = 3 (*trigram model*)

P("We are at The University of Sydney")
~ P(Sydney | *University of*) * P(of | The University) …

Markov Assumption with n = 4 (*4-gram model*)

P("We are at The University of Sydney")
~ P(Sydney | *The University of*) * P(of | at The University) …

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Why not set n very high?

Sparsity - the longer the sequence, the harder it is for us to accurately estimate these probabilities

Storage complexity - many many options to store

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

How do we actually calculate one of these probabilities?

$$P(\text{Sydney} \mid \text{of}) = \frac{\text{Count(of Sydney)}}{\text{Count(of)}} = \frac{\text{Count(of Sydney)}}{\sum_{w \in \text{vocab}} \text{Count(of } w\text{)}}$$

Count(of Sydney) = How often it occurs in some data

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What happens for rare word combinations?

P(The bird opened the cage)

"bird opened"? 0

problem

$$P(\text{opened} \mid \text{bird}) = \frac{\text{Count(bird opened)}}{\text{Count(bird)}}$$

$$= \frac{0}{\text{Count(bird)}}$$

$$= 0$$

There are a range of ways to deal with this, which we won't get into

no detail in this Unit

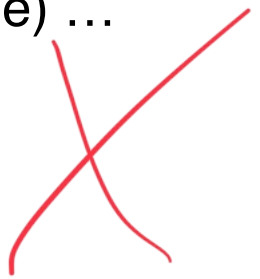
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What do we do for the start and end of a sequence?

P(We are at The University of Sydney)
~ P(Sydney | of) * P(of | University) * P(University | The) …



COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What do we do for the start and end of a sequence?

P(We are at The University of Sydney)
~ P(Sydney | of) *
P(of | University) *
P(University | The) *
P(The | at) *
P(at | are) *
P(are | We) *
P(We | ??????)

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What do we do for the start and end of a sequence?

P(We are at The University of Sydney)
~ P(Sydney | of) *
P(of | University) *
P(University | The) *
P(The | at) *
P(at | are) *
P(are | We) *
P(We | <start>)

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What do we do for the start and end of a sequence?

P(We are at The University of Sydney)
~ P(Sydney | University of) *
P(of | The University) *
P(University | at The) *
P(The | are at) *
P(at | We are) *
P(are | <start> We) *
P(We | <start> <start>)

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What do we do for the start and end of a sequence?

P(We are at The University of Sydney)
~ P(Sydney | The University of) *
P(of | at The University) *
P(University | are at The) *
P(The | We are at) *
P(at | <start> We are) *
P(are | <start> <start> We) *
P(We | <start> <start> <start>)

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What do we do for the start and end of a sequence?

P(We are at The University of Sydney)
~ P(<end> | University of Sydney) *
P(Sydney | The University of) *
P(of | at The University) *
P(University | are at The) *
P(The | We are at) *
P(at | <start> We are) *
P(are | <start> <start> We) *
P(We | <start> <start> <start>)

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We use log probabilities to avoid numerical issues

Vcabulary size = 100,000

Mean probability of next word = 10^{-6}

Paragraph of text = 100 words

Probability = $(10^{-6})^{100} = 10^{-600}$

Smallest positive floating point number: ~10^{-307}
(Ignoring subnormal numbers)

Underflow!

Using log(Probability) resolves this

small number

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Recap: N-Gram Language Modelling

Language models take a string as input and produce a score for it as output.

N-gram language models do this by making a Markov assumption and estimating probabilities based on counts of word sequences seen in data. Care must be taken to handle rare words, the start and end of sequences, and small probabilities.

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

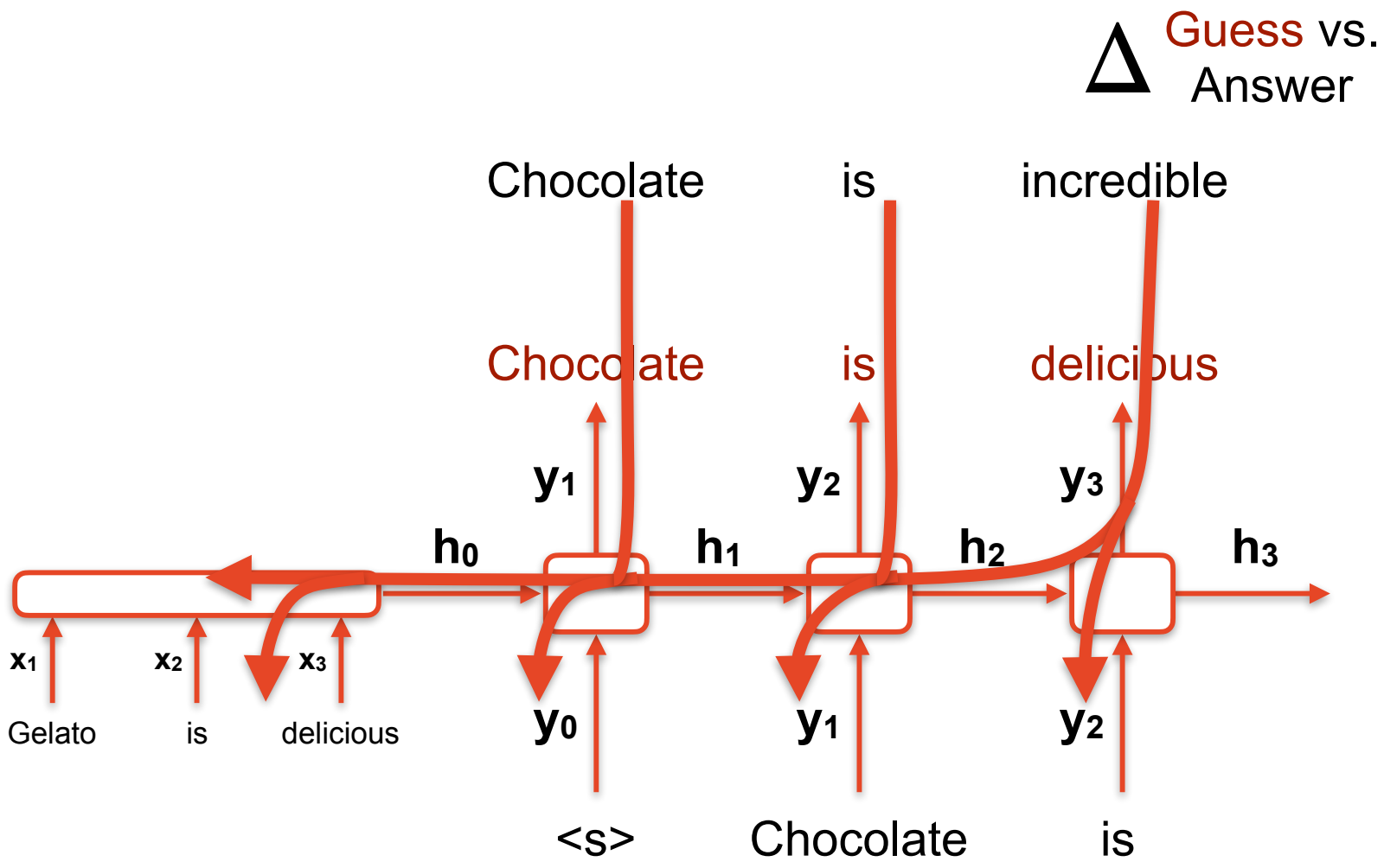
Training LLMs

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

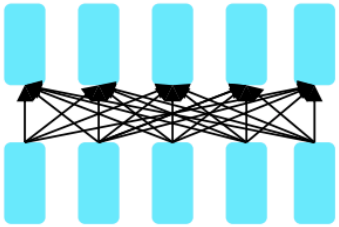
We can train these models with **next-token prediction**



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

What distributions are these modelling?



Encoders

$$P(x_i \mid x_{1\ldots i-1}, x_{i+1\ldots N})$$

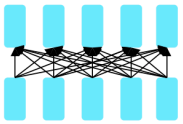
COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Encoder

We can train with **masked-token prediction**

Model: BERT



Full name: "Bidirectional Encoder Representations from Transformers"

uess vs. Answer

Sydney

Cambridge

Note: The input does not contain the answer, so we can do attention over the whole sequence

Ignore outputs for unmasked words

Model

The idea is similar to word2vec training

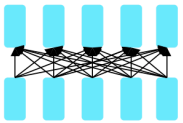
<s> I like [MASK] University's NLP course .

COMP 4446 / 5046
Lecture 8, 2025

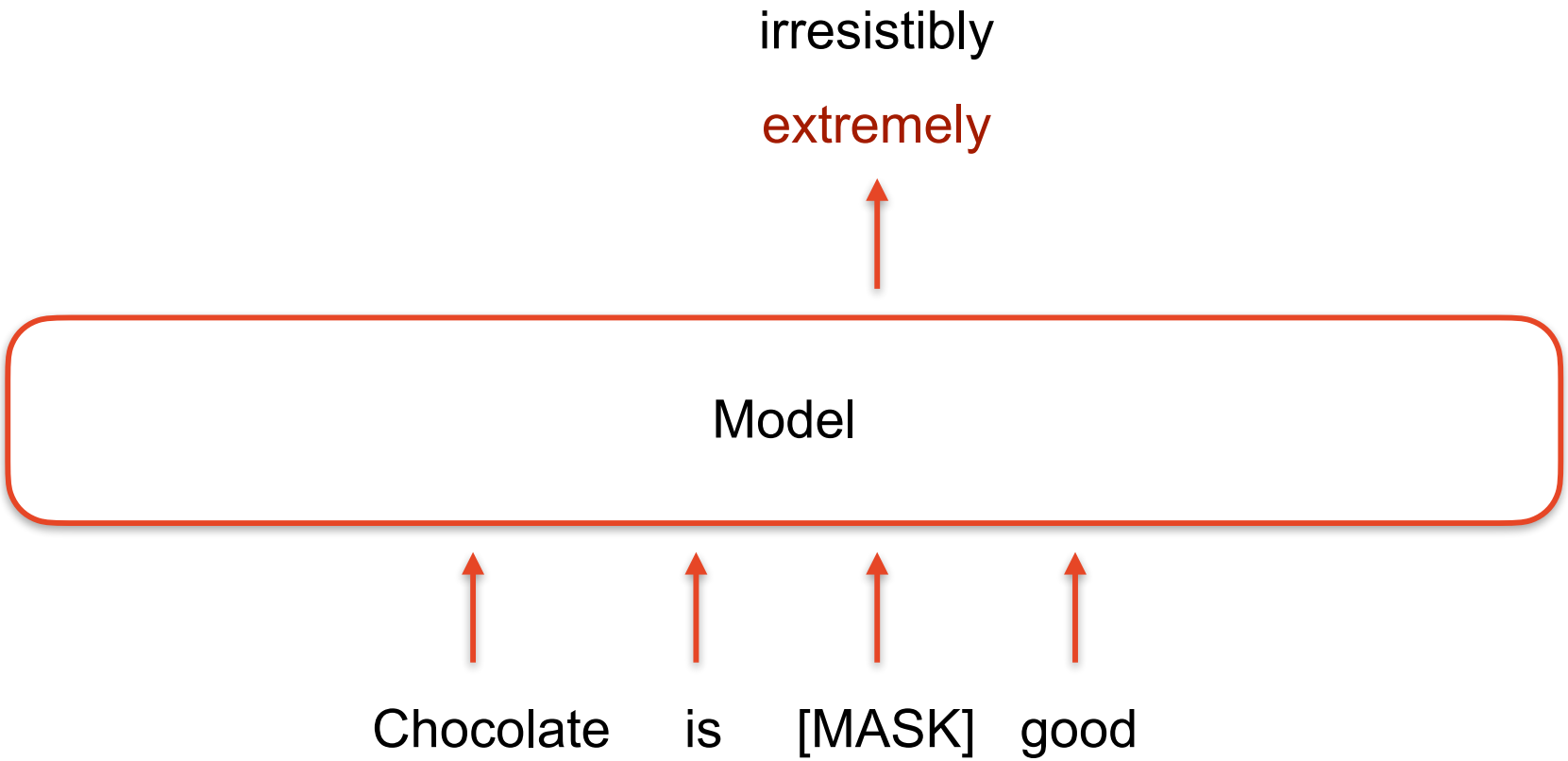
menti.com 4182 4438

We can train with **masked-token prediction**

Model: BERT



Δ Guess vs. Answer

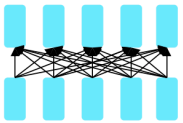


COMP 4446 / 5046
Lecture 8, 2025

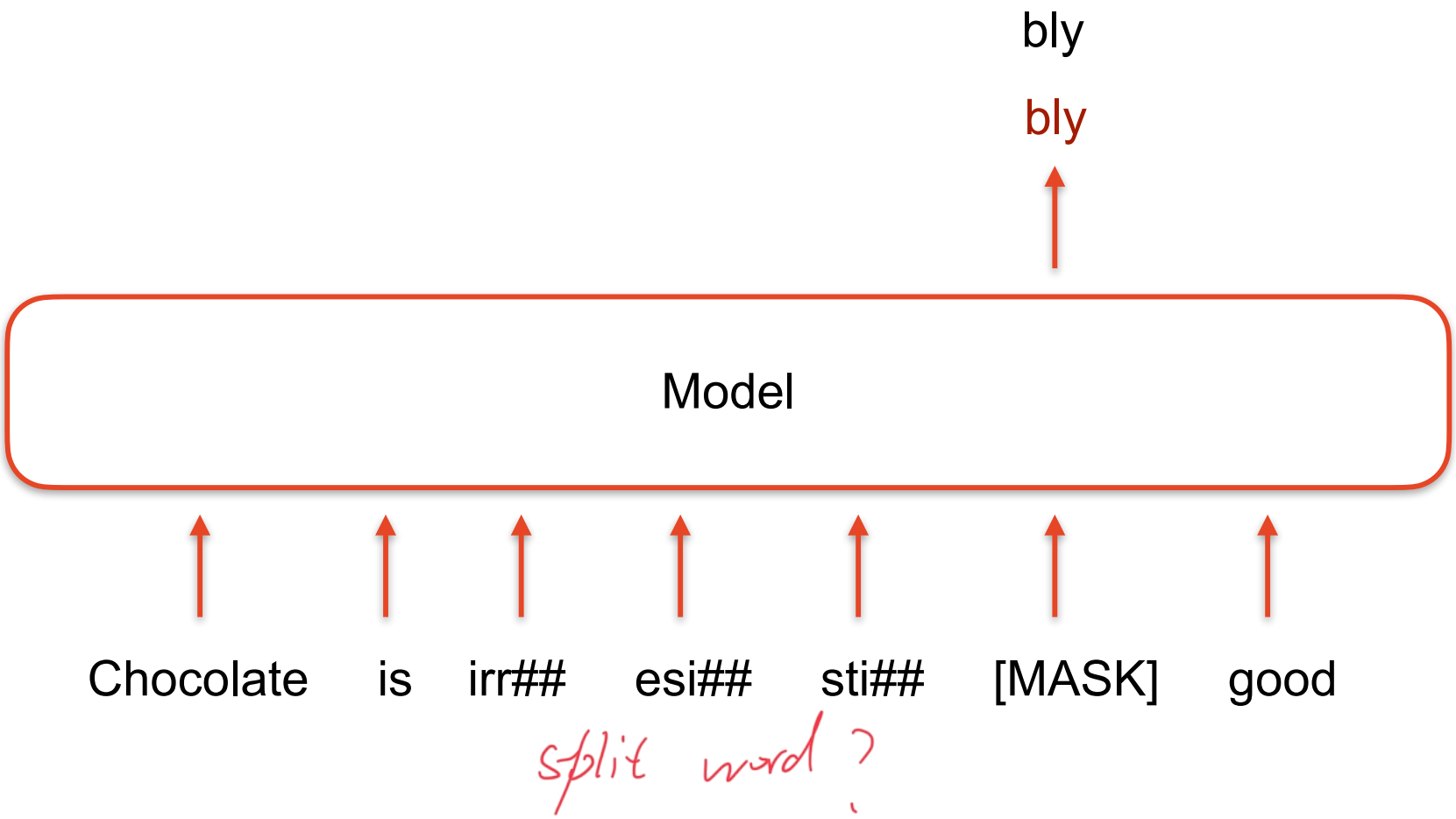
menti.com 4182 4438

We can train with **masked-token prediction**

Model: BERT



Δ Guess vs. Answer

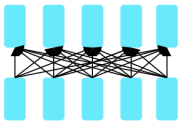


COMP 4446 / 5046
Lecture 8, 2025

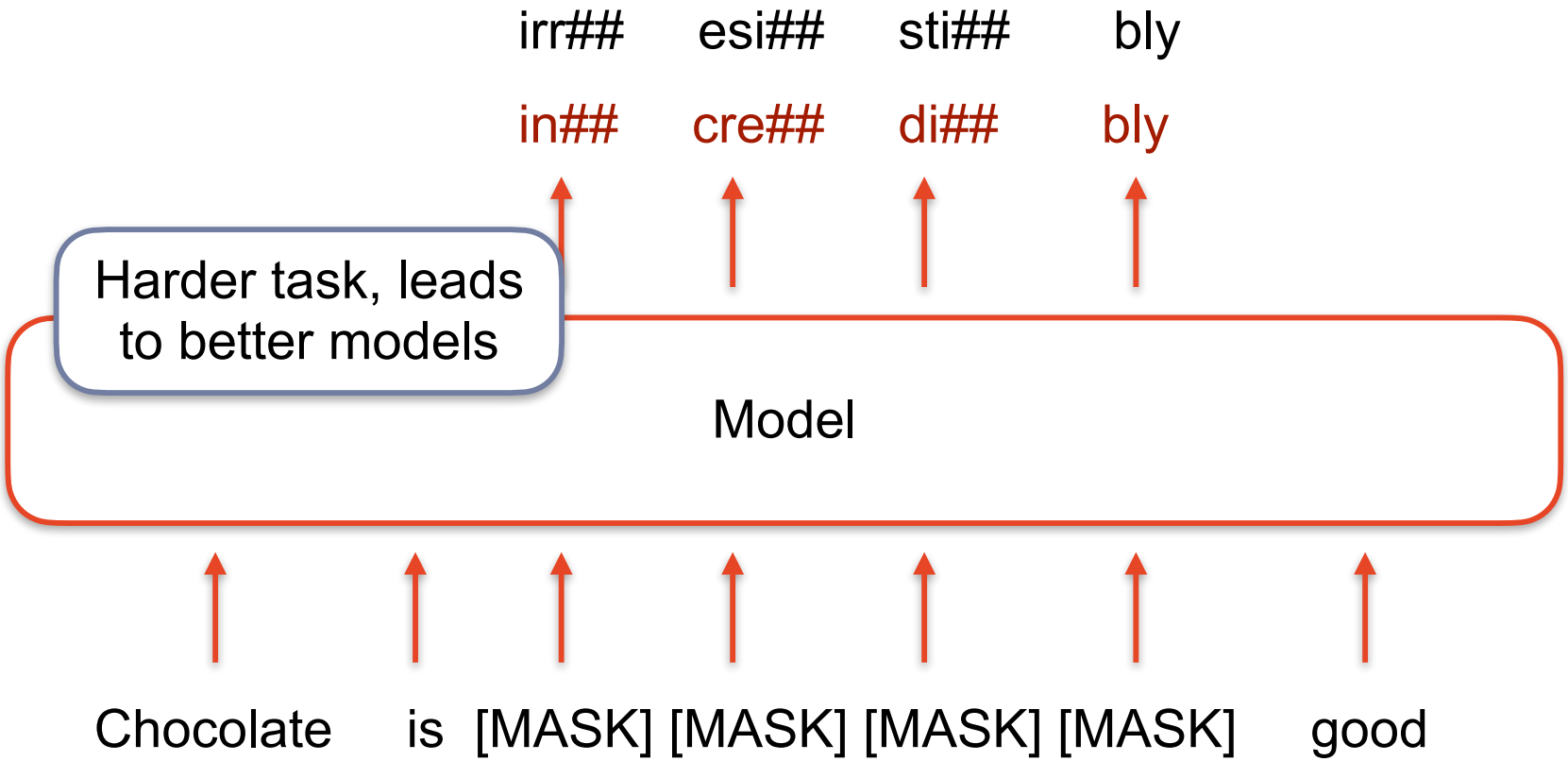
menti.com 4182 4438

We can train with **span prediction**

Model: SpanBERT



Δ Guess vs. Answer

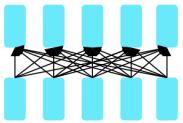


COMP 4446 / 5046
Lecture 8, 2025

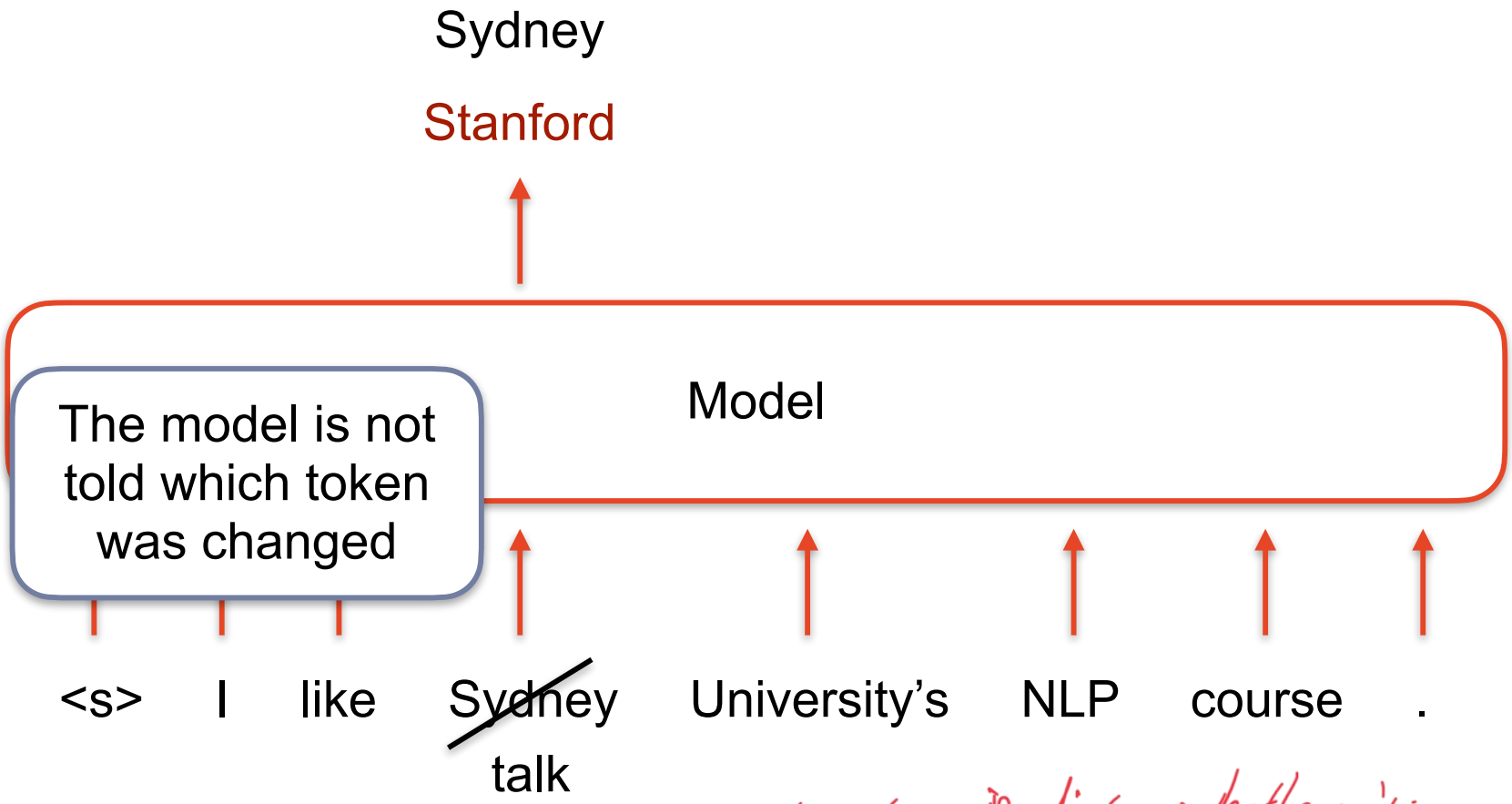
menti.com 4182 4438

We can train with **random-token correction**

Model: BERT



Δ Guess vs. Answer



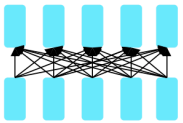
modify input, predict whether it's wrong or not

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with a combination

Model: BERT



Δ Guess vs. Answer

Sydney Stanford NLP AI course class

BERT combines these and also checks for some unmasked words

Model

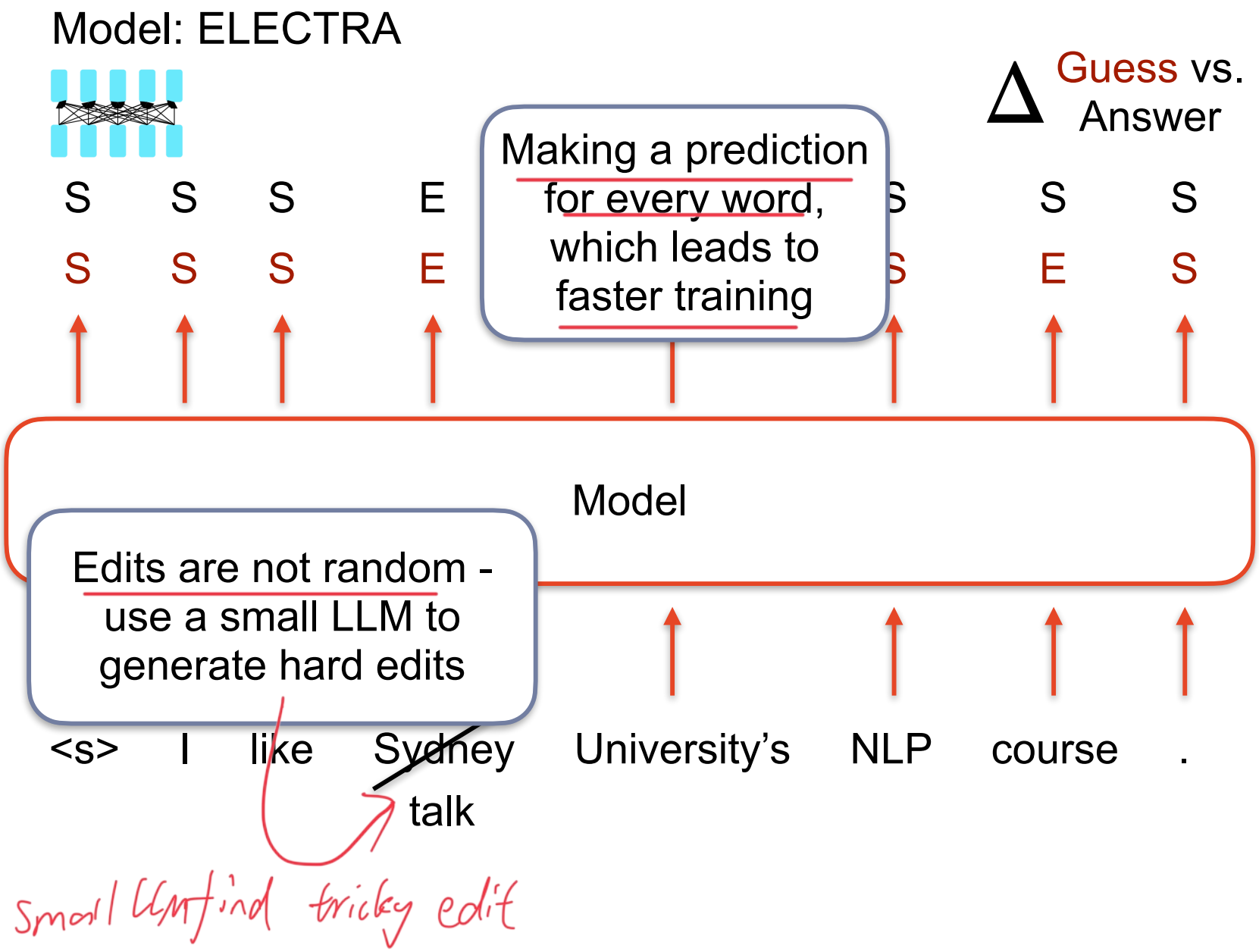
<s> I like Sydney talk University's [MASK] course .

两种都对

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with **token edit detection**

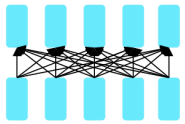


COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with **next sentence prediction**

Model: BERT



Yes Next

Not Next

Subsequent research suggests this does not actually help much in BERT

e.g., RoBERTa does better than BERT by training for longer and not using this task

Δ Guess vs. Answer

Model

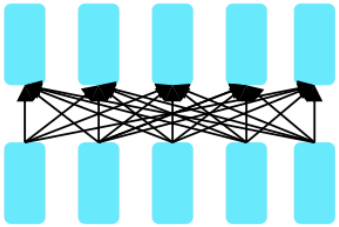
[CLS] I like NLP [SEP] Transformers are cool [SEP]

2 sentences

COMP 4446 / 5046
Lecture 8, 2025

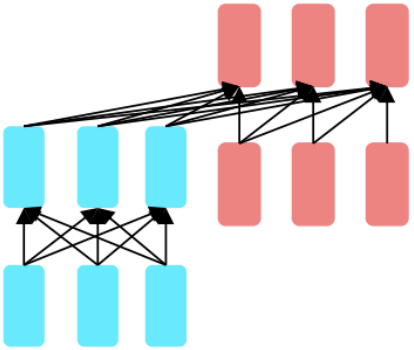
menti.com 4182 4438

What distributions are these modelling?



Encoders

$$P(x_i \mid x_{1\ldots i-1}, x_{i+1\ldots N})$$



Encoder-Decoders

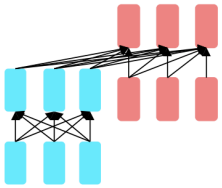
$$P(y_i \mid x_{1\ldots N}, y_{1\ldots i-1})$$

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with **masked sequence prediction**

Model: BART



Δ Guess vs. Answer

I study NLP at Sydney Uni

I study CS at my school

Encoder-Decoder

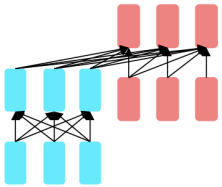
I study [MASK] at [MASK]

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with **masked span prediction**

Model: T5



Δ Guess vs. Answer

<X> NLP <Y> Sydney Uni

<X> Comp. Sci. <Y> Uni

Encoder-Decoder

I study <X> at <Y>

不知道具体长度

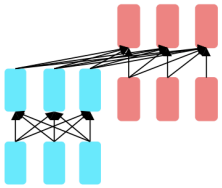
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can train with **deleted sequence prediction**

Model: BART



Δ Guess vs. Answer

I study NLP at Sydney Uni

I study CS at my school

Encoder-Decoder

I study at

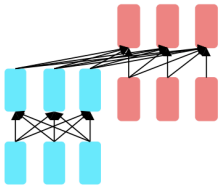
The model needs to work out where deletion happened, as well as what was deleted

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with **deleted span prediction**

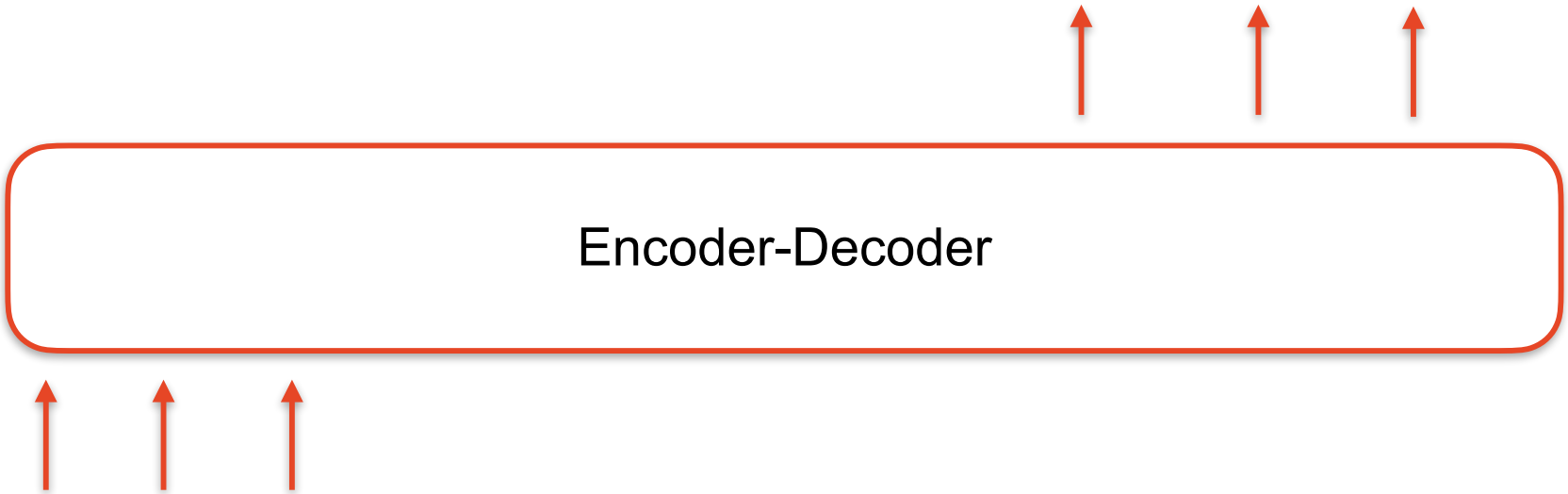
Model: T5



Δ Guess vs. Answer

NLP Sydney Uni

Comp. Sci. Uni



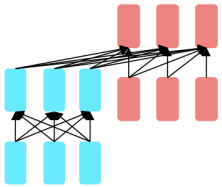
I study at

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with **permuted sequence prediction**

Model: BART



Δ Guess vs. Answer

I study NLP at Sydney Uni

At Sydney Uni I study NLP

Encoder-Decoder

NLP Uni Sydney study I at

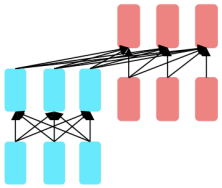
Rearrange sentence

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with **rotated sequence prediction**

Model: BART



Δ Guess vs. Answer

I study NLP at Sydney Uni

At Sydney Uni I study NLP

Encoder-Decoder

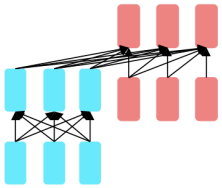
Sydney Uni I study NLP at

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train with **infilling sequence prediction**

Model: BART



Δ Guess vs. Answer

I study NLP at Sydney Uni

I study CS at my school

Encoder-Decoder

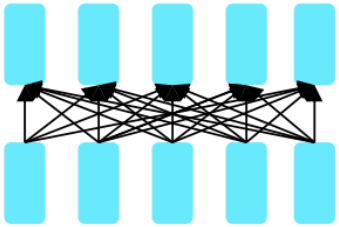
I [MASK] study [MASK] at [MASK]

Like span prediction, but masks can also be replaced by nothing

COMP 4446 / 5046
Lecture 8, 2025

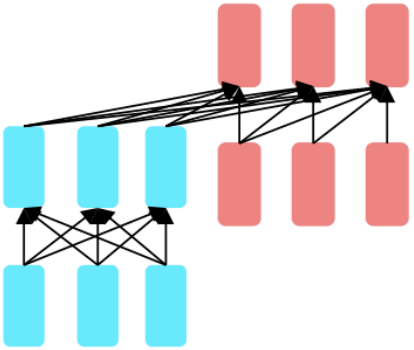
menti.com 4182 4438

What distributions are these modelling?



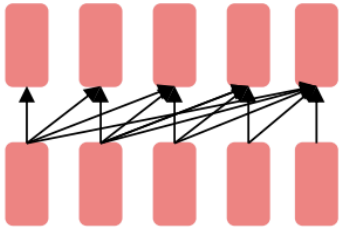
whole

Encoders $P(x_i \mid x_{1\ldots i-1}, x_{i+1\ldots N})$



whole encoder + decoder before current

Encoder-Decoders $P(y_i \mid x_{1\ldots N}, y_{1\ldots i-1})$



Decoders $P(y_i \mid y_{1\ldots i-1})$

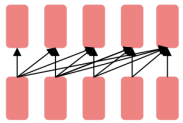
before current

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train these models with **next-token prediction**

Model: GPT



Δ Guess vs. Answer

I	like	Sydney	University's	NLP	course	.	</s>
I	think	a	University's	AI	course	a	</s>

Model

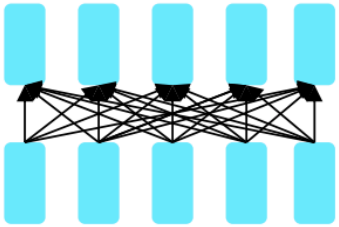
<s>	I	like	Sydney	University's	NLP	course	.

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

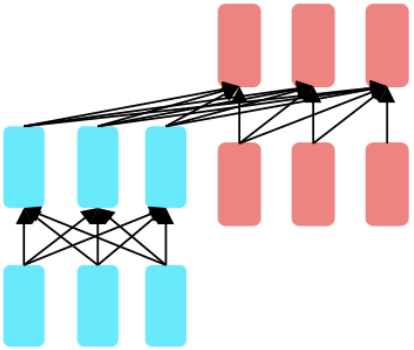
Each model type needs different training - but similar ideas



Encoders

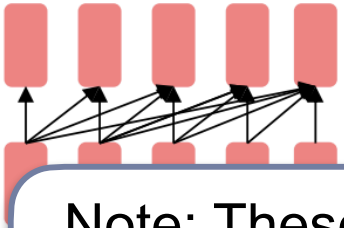
Example models:

BERT, ELECTRA



Encoder-Decoders

T5, BART



Decoders

ELMo, GPT

Note: These are ‘classic’ examples of these approaches.
There are better variants today, e.g., ModernBERT

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

What are we teaching them?

Sydney Uni is located in _____, Australia.
[facts]

I put ___ fork down on the table.
[syntax]

The woman walked across the street, checking for traffic over ___ shoulder.
[coreference]

I went to the ocean to see the fish, turtles, seals, and _____.
[lexical semantics/topic]

Overall, the value I got from the two hours watching it was the sum total of the popcorn and the drink. The movie was ___.
[sentiment]

Iroh went into the kitchen to make some tea. Standing next to Iroh, Zuko pondered his destiny. Zuko left the _____.
[some reasoning – this is harder]

I was thinking about the sequence that goes 1, 1, 2, 3, 5, 8, 13, 21, ____
[some basic arithmetic; they don't learn the Fibonnaci sequence]

Stanford CS224 lectures

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Recap: Training LLMs

Three General Model Types: Models can be classified as (a) encoders, (b) encoder-decoders, or (c) decoders. They differ in their architecture and most natural uses. Most models today are decoders.

Training Tasks: All of these models train by taking plain text, somehow modifying it, and then getting the model to identify / fix the modification. Different model types are trained with different tasks. The most widely used and well known are:

Next token prediction - given the start of text, predict the next token
Masked token prediction - given text with some tokens masked out, predict the masked tokens

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Using LLMs

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

The difference between pre-training and training

Both implemented the same way - backpropagation based on a loss function

Differences:

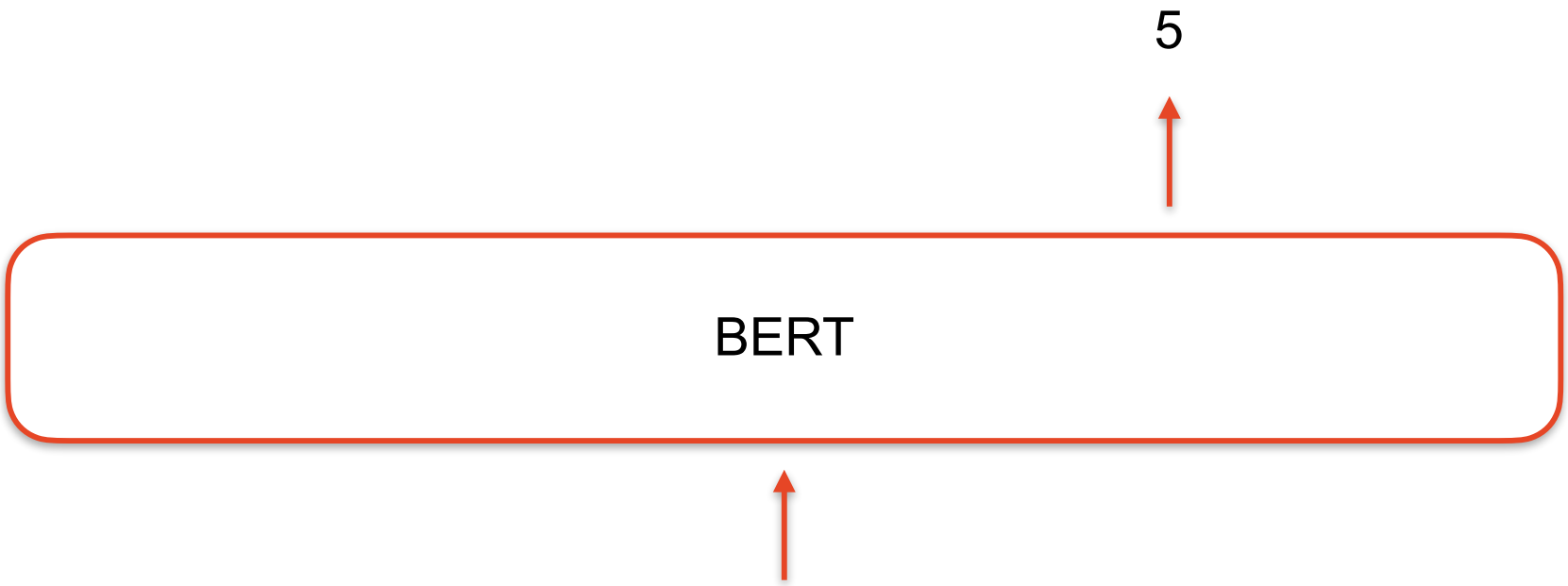
Pre-training	• Done once • Long training • Trained on lots of GPUs • Input + Output are based on raw text
Training	• Done many times • Can be short or long training • Can be on limited compute or a lot • Task specific data

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can describe our task like their training tasks



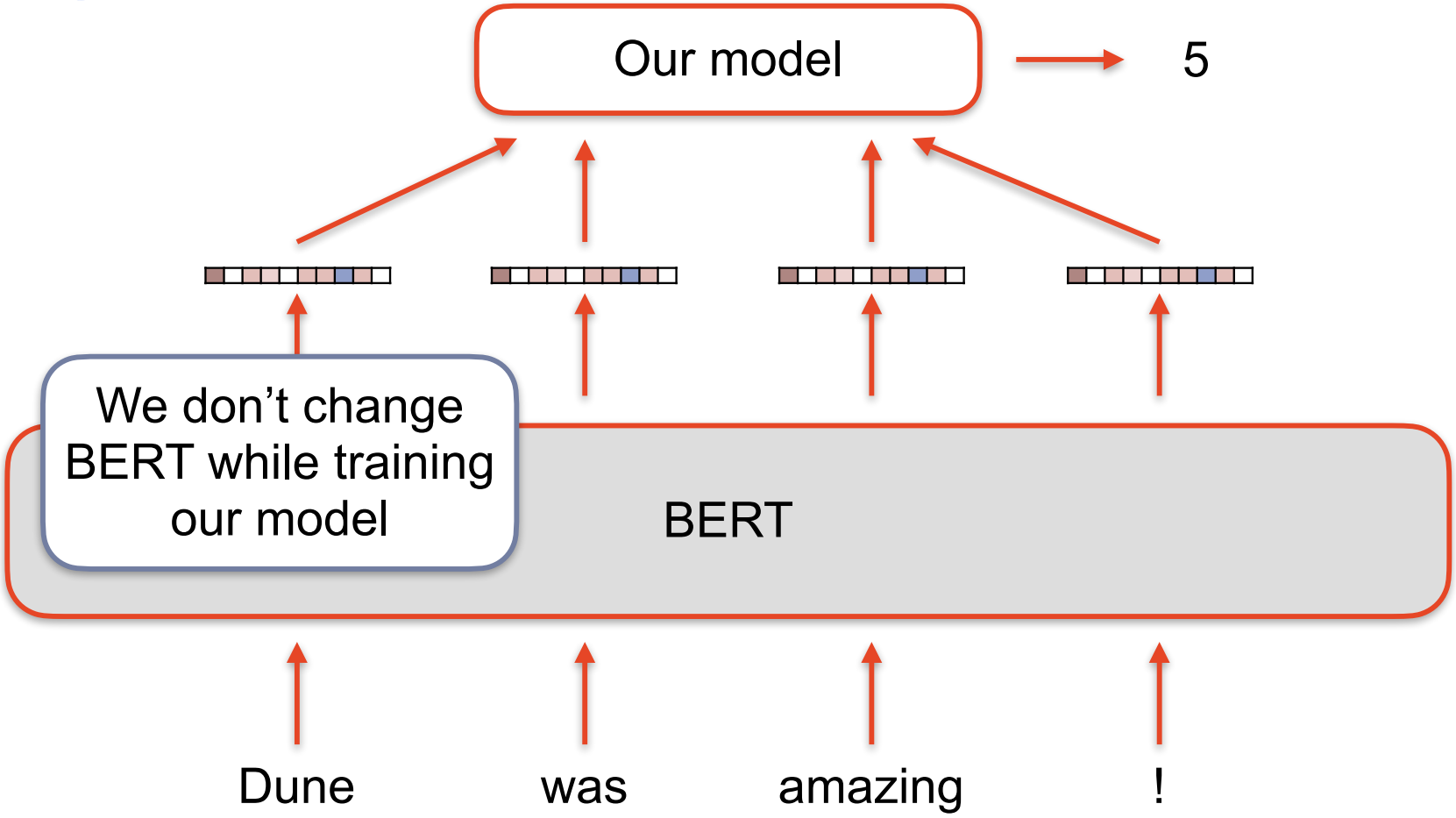
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can use the output representations as input to a model

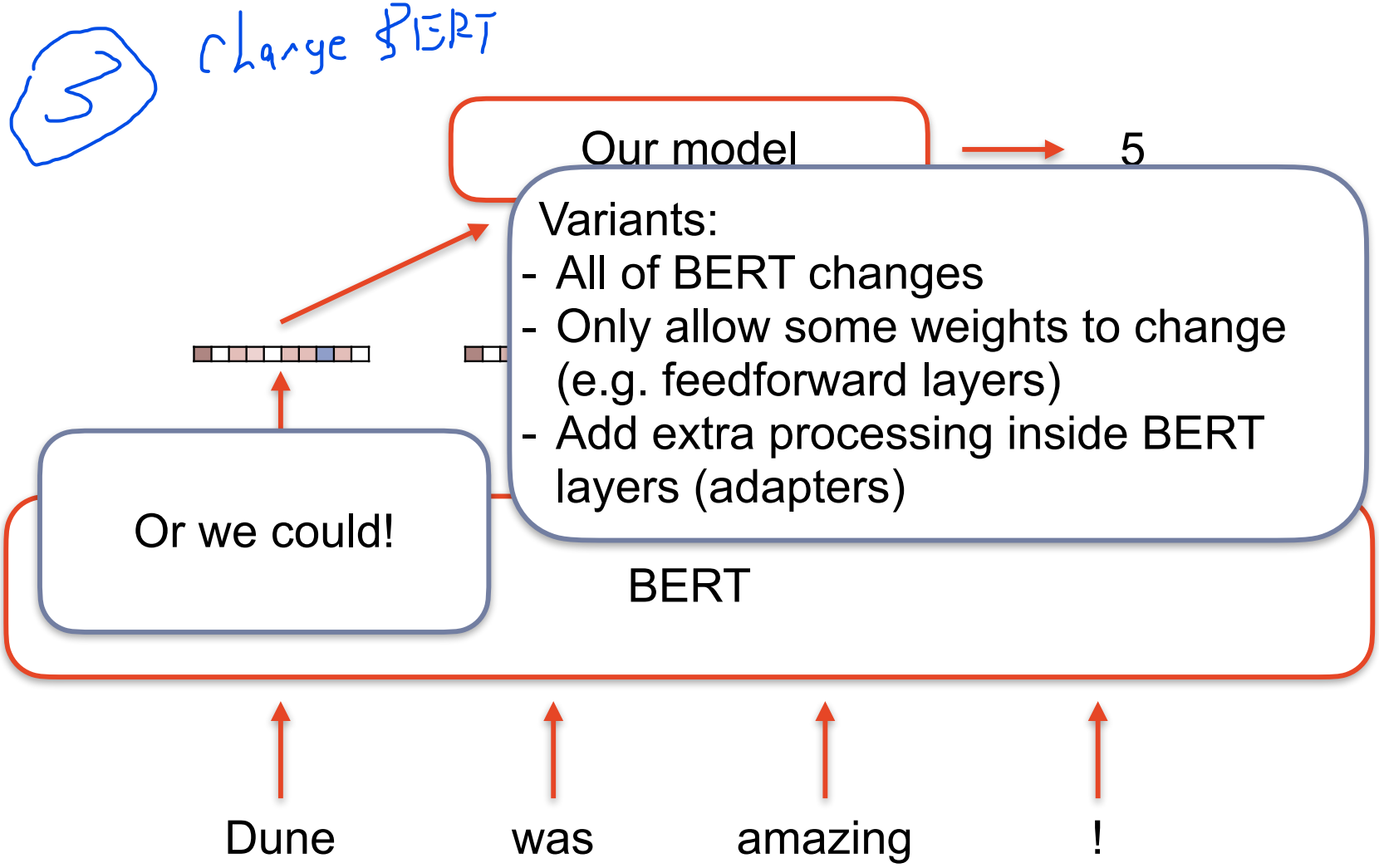
(2) as input of our model + do not change BERT



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can train them on our own data + task (fine-tuning)



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Which tasks can be done by each model directly?
(i.e., without passing the vectors into another model on top)

Let's consider:

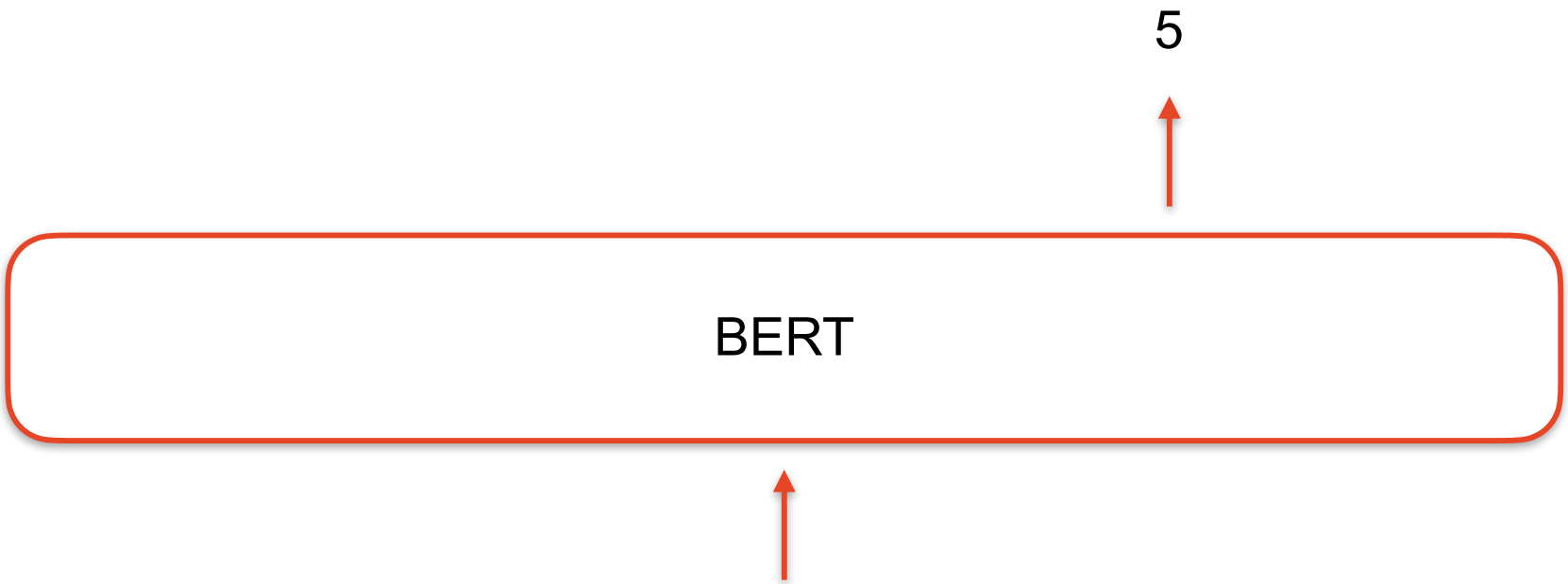
Sentiment (Classification)
NER (Token predictions)
Coreference (Structured Output)
Summarisation (Generation)
Translation (Generation, in another language)

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

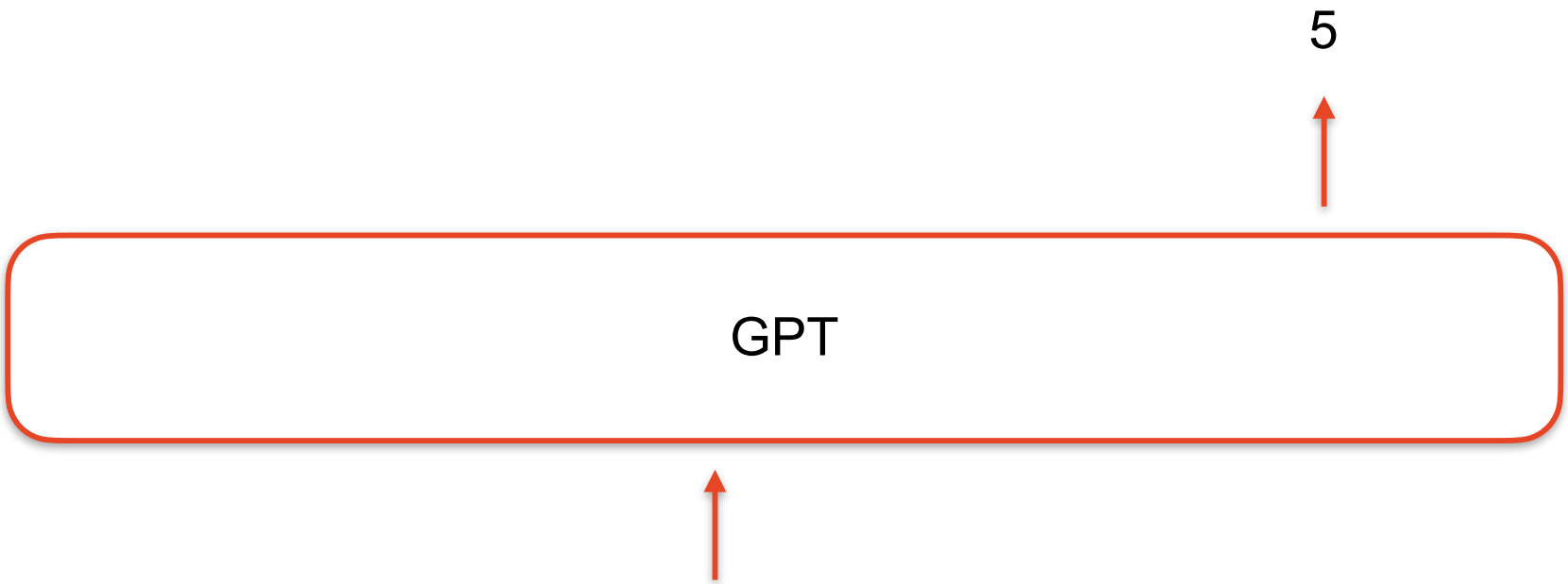
Sentiment with an encoder model



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

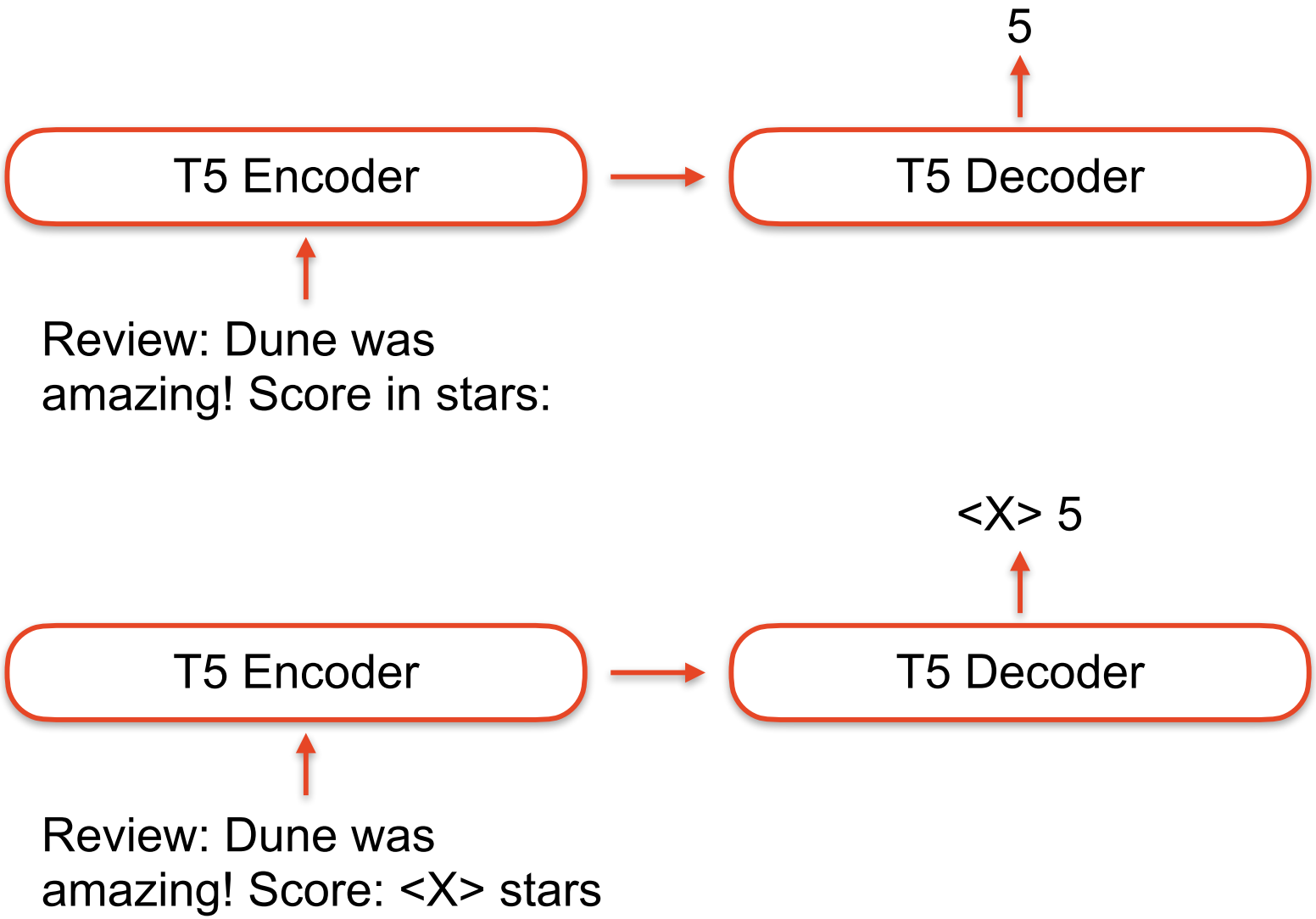
Sentiment with a decoder model



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Sentiment with an encoder-decoder model

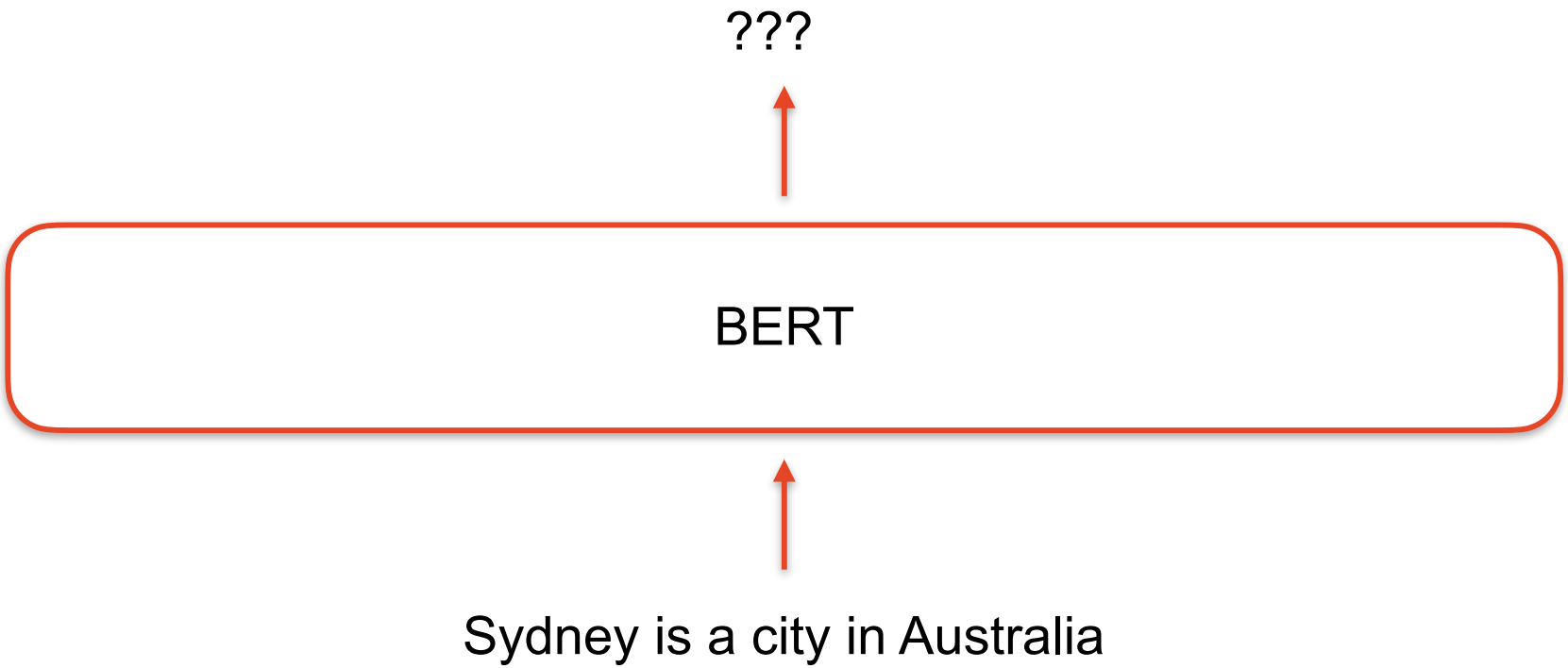


COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

NER with an encoder model



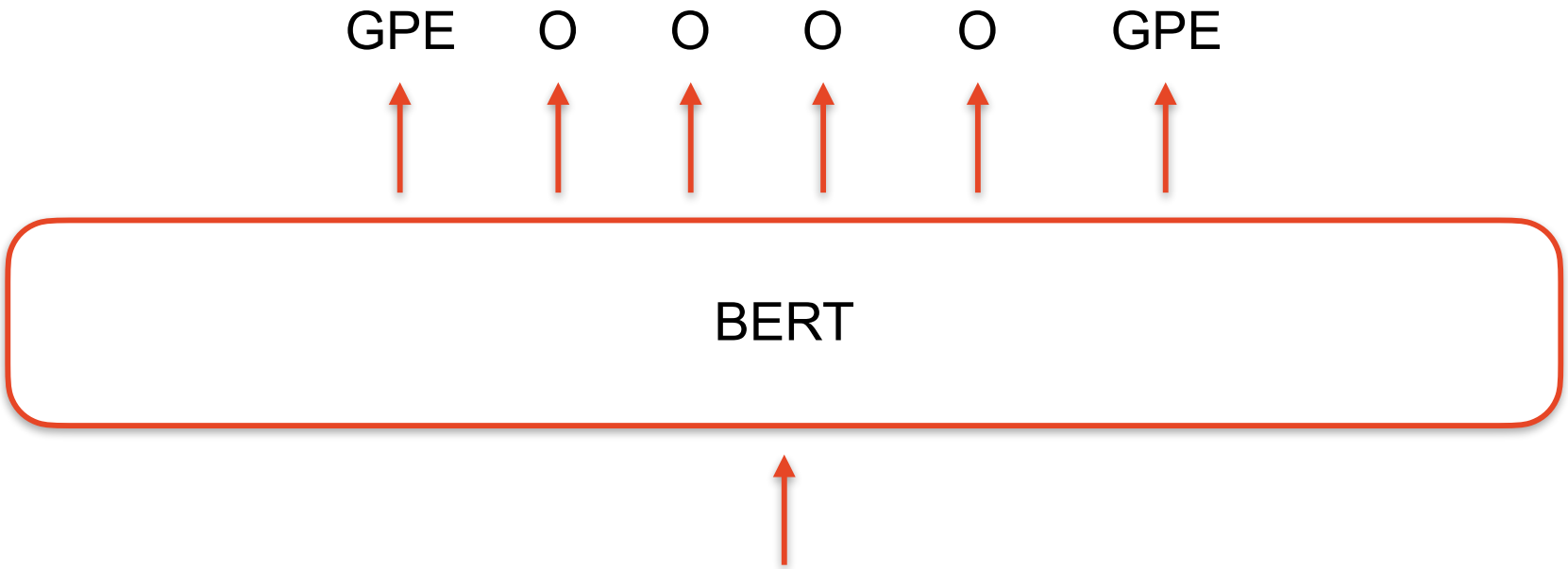
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

NER with an encoder model

Train with NER data to 'teach' BERT to produce "LOC", "GPE", "O", etc



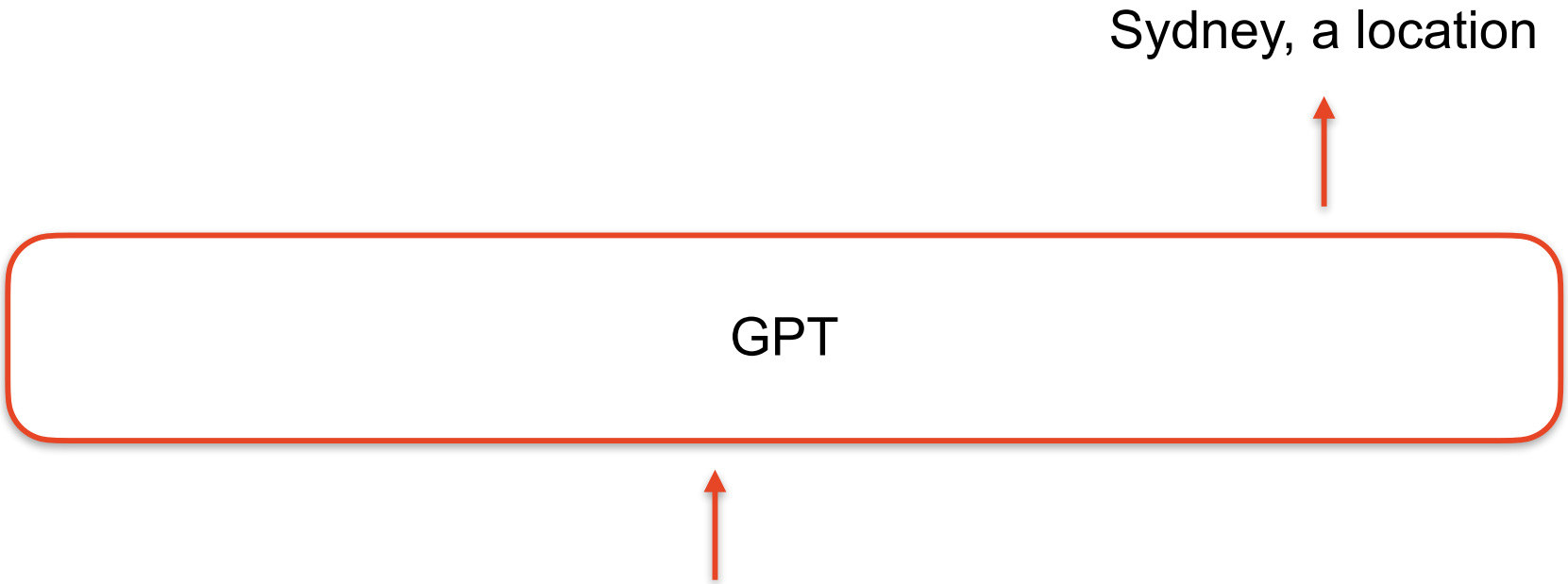
每个词之间加上 mask

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

NER with a decoder model



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

NER with an encoder model

Sydney is a city in Australia.
The entities in this text are:
→ BERT → Sydney

Sydney is a city in Australia.
The entities in this text are:
Sydney
→ BERT → ,

Sydney is a city in Australia.
The entities in this text are:
Sydney,
→ BERT → a

Syd…lia.
The…e:
Sy…
→ BERT → location

Slow / expensive, so not commonly done

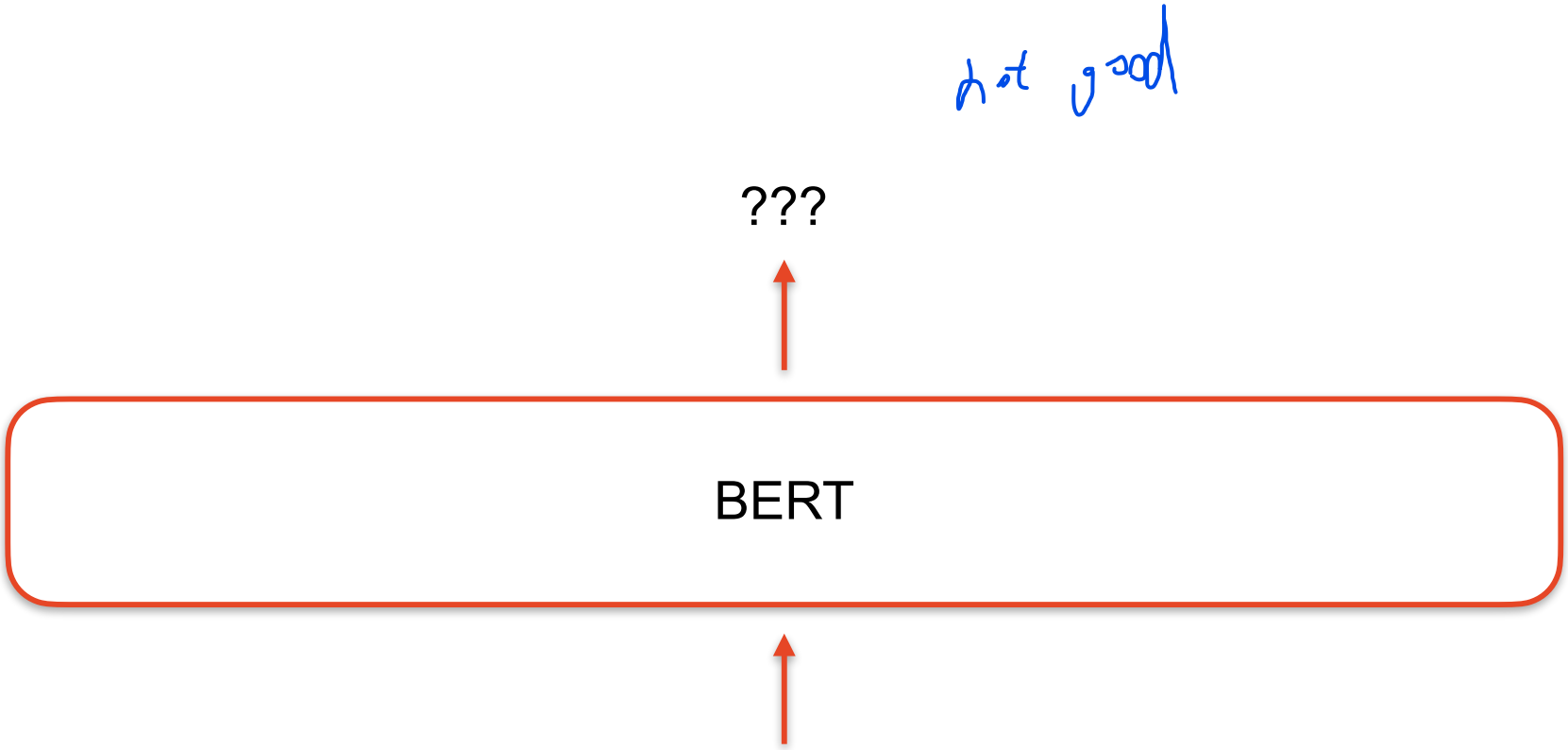
attention every where

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Coreference with an encoder model

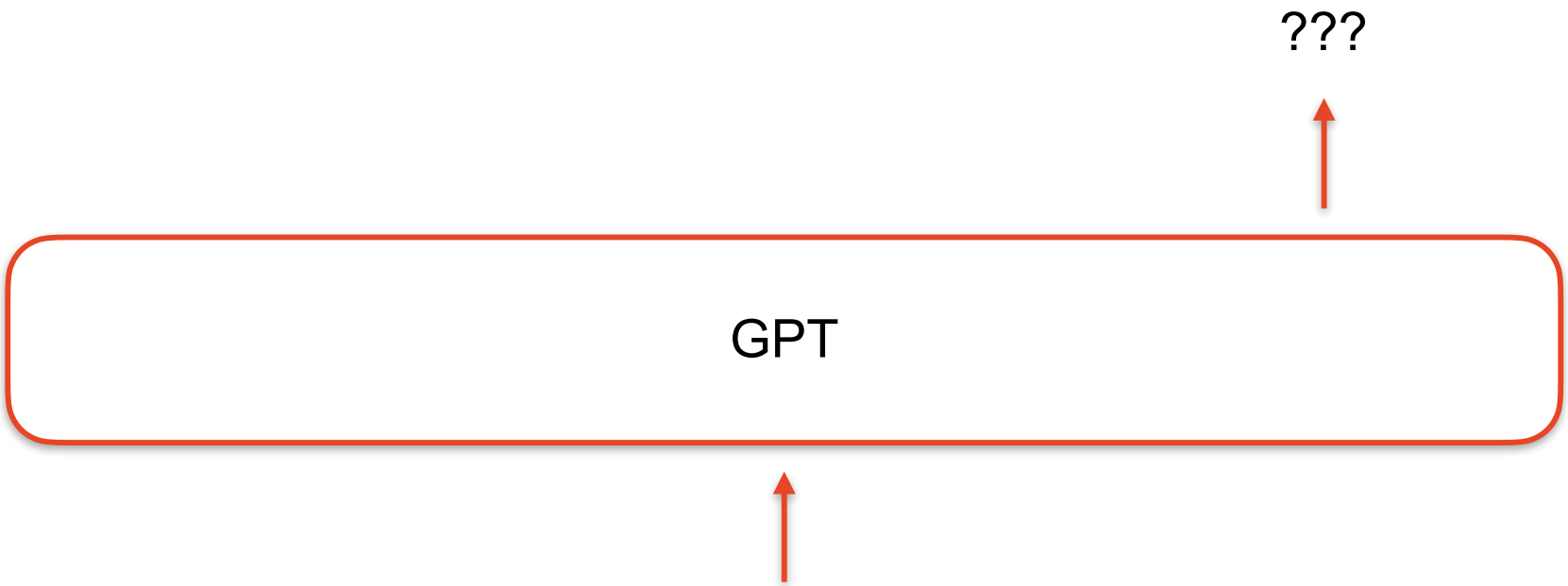


COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Coreference with a decoder model

not good



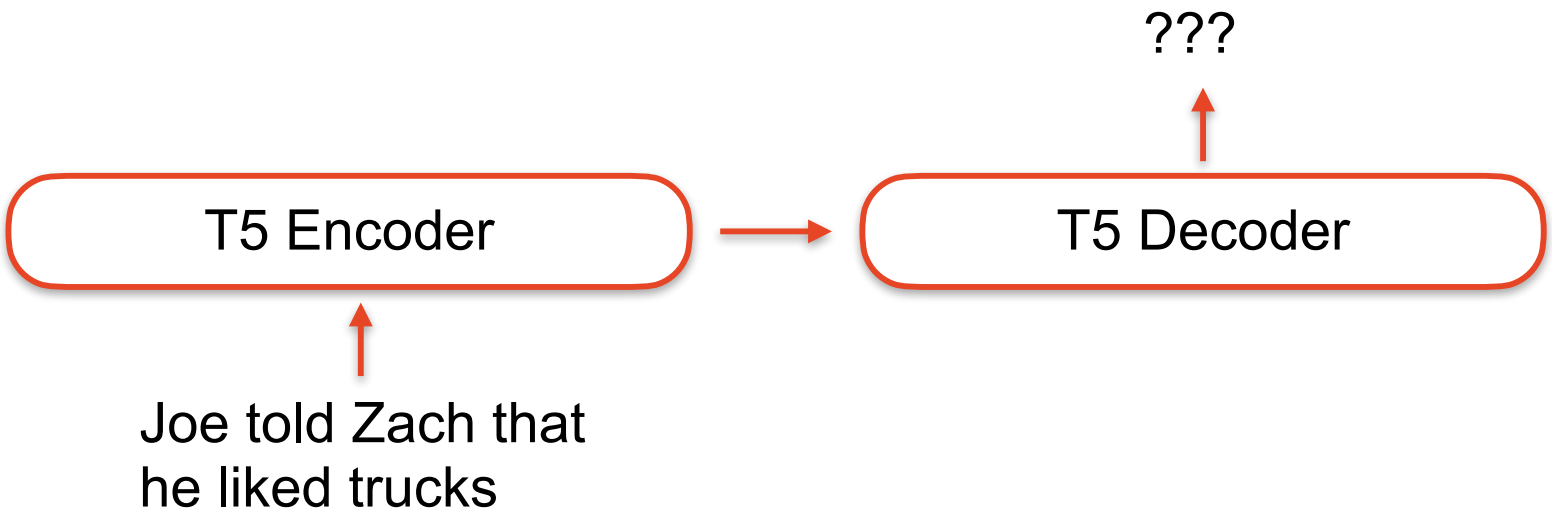
COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Coreference with an encoder-decoder model

Coreference Resolution through a seq2seq Transition-Based System

Bernd Bohnet[1], **Chris Alberti**[2], **Michael Collins**[2]
[1]Google Research, The Netherlands [2]Google Research, USA
{bohnetbd,chrisalberti,mjcollins}@google.com



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Coreference with an encoder-decoder model

Input: *Speaker-A* I still have n't gone to that fresh French restaurant by your house
Prediction: SHIFT: next sentence

Input: *Speaker-A* I_2 still have n't gone to that fresh French restaurant by your house *Speaker-A* I_{17} 'm like dying to go there
Prediction:
A $I_{17} \rightarrow I_2$
B SHIFT: next sentence

Input: *Speaker-A* [1 I] still have n't gone to that fresh French restaurant by your house *Speaker-A* [1 I] 'm like dying to go there *Speaker-B* You mean the one right next to the apartment
Prediction:
A You → [1
B the apartment → your house
C the one right next to the apartment → that fresh French restaurant by your house
D SHIFT: next sentence

Input: *Speaker-A* [1 I] still have n't gone to [3 that fresh French restaurant by [2 your house]] *Speaker-A* [1 I] 'm like dying to go there *Speaker-B* [1 You] mean [3 the one right next to [2 the apartment]] *Speaker-B* yeah yeah yeah
Prediction: SHIFT: next sentence

One advantage of an encoder-decoder is that the input and output can have very different properties

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

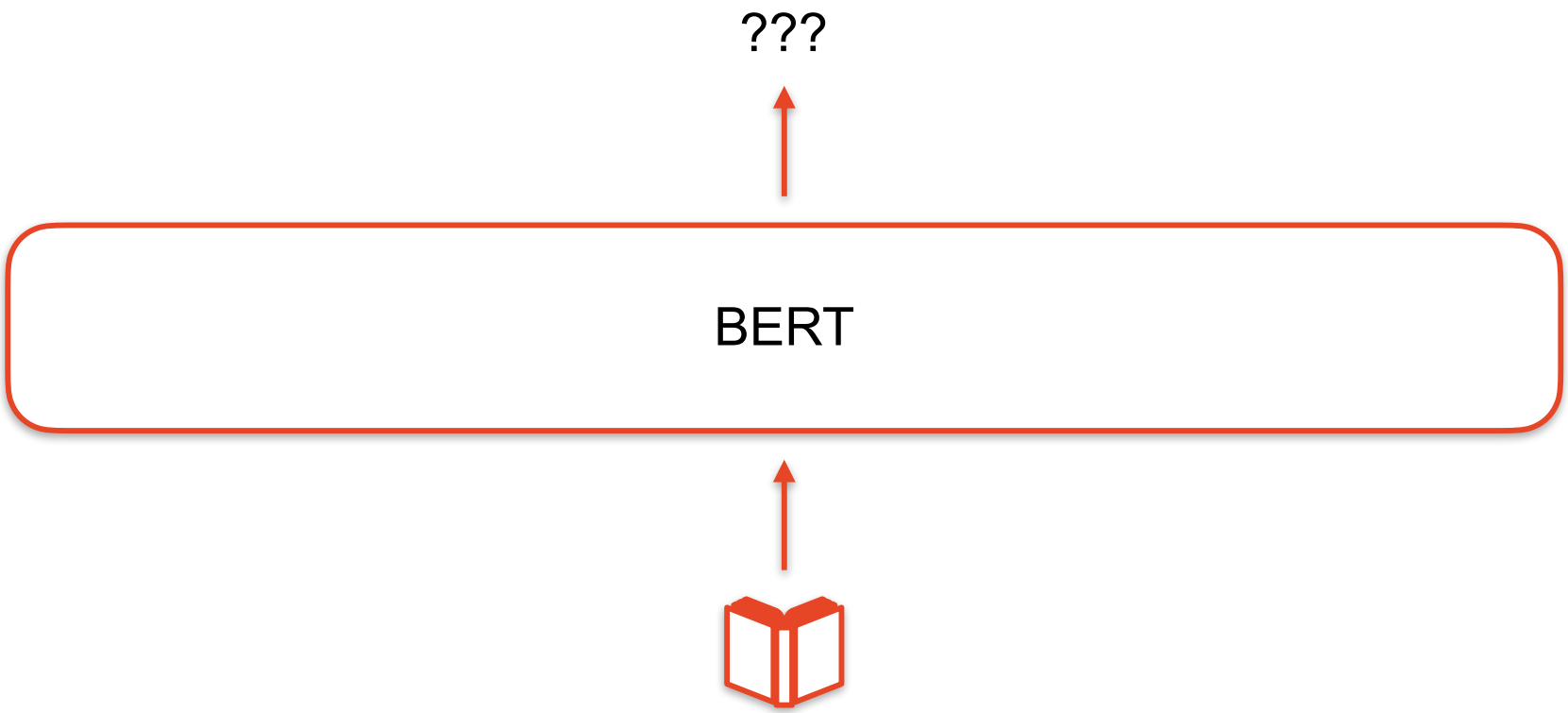
Coreference with an encoder-decoder model

	LM	Decoder	MUC			B^3			$CEAF_{\Phi_4}$			Avg.
			P	R	F1	P	R	F1	P	R	F1	F1
English												
Lee et al. (2017)	–	neural e2e	78.4	73.4	75.8	68.6	61.8	65.0	62.7	59.0	60.8	67.2
Lee et al. (2018)	Elmo	c2f	81.4	79.5	80.4	72.2	69.5	70.8	68.2	67.1	67.6	73.0
Joshi et al. (2019)	BERT	c2f	84.7	82.4	83.5	76.5	74.0	75.3	74.1	69.8	71.9	76.9
Yu et al. (2020)	BERT	Ranking	82.7	83.3	83.0	73.8	75.6	74.7	72.2	71.0	71.6	76.4
Joshi et al. (2020)	SpanBERT	c2f	85.8	84.8	85.3	78.3	77.9	78.1	76.4	74.2	75.3	79.6
Xia et al. (2020)	SpanBERT	transitions	85.7	84.8	85.3	78.1	77.5	77.8	76.3	74.1	75.2	79.4
Wu et al. (2020)	SpanBERT	QA	88.6	87.4	88.0	82.4	82.0	82.2	79.9	78.3	79.1	*83.1**
Xu and Choi (2020)	SpanBERT	hoi	85.9	85.5	85.7	79.0	78.9	79.0	76.7	75.2	75.9	80.2
Kirstain et al. (2021)	LongFormer	bilinear	86.5	85.1	85.8	80.3	77.9	79.1	76.8	75.4	76.1	80.3
Dobrovolskii (2021)	RoBERTa	c2f	84.9	87.9	86.3	77.4	82.6	79.9	76.1	77.1	76.6	81.0
Link-Append	mT5	transition	87.4	88.3	87.8	81.8	83.4	82.6	79.1	79.9	79.5	**83.3**
Arabic												
Aloraini et al. (2020)	AraBERT	c2f	63.2	70.9	66.8	57.1	66.3	61.3	61.6	65.5	63.5	63.9
Min (2021)	GigaBERT	c2f	73.6	61.8	67.2	70.7	55.9	62.5	66.1	62.0	64.0	64.6
Link-Append	mT5	transition	71.0	70.9	70.9	66.5	66.7	66.6	68.3	68.6	68.4	**68.7**
Chinese												
Xia and Durme (2021)	XLM-R	transition	–	–	–	–	–	–	–	–	–	69.0
Link-Append	mT5	transition	81.5	76.8	79.1	76.1	69.9	72.9	74.1	67.9	70.9	**74.3**

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Summarisation with an encoder model

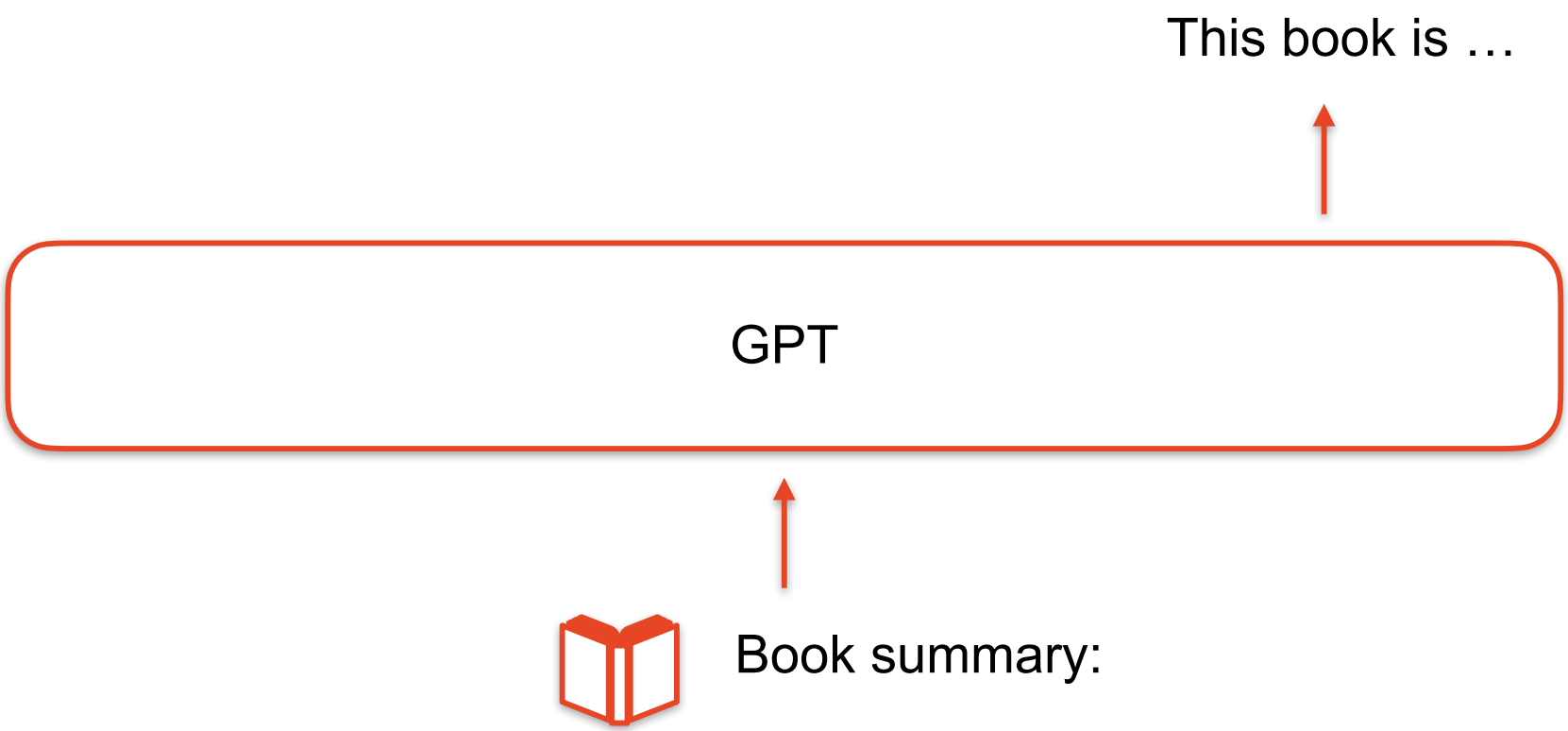


COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

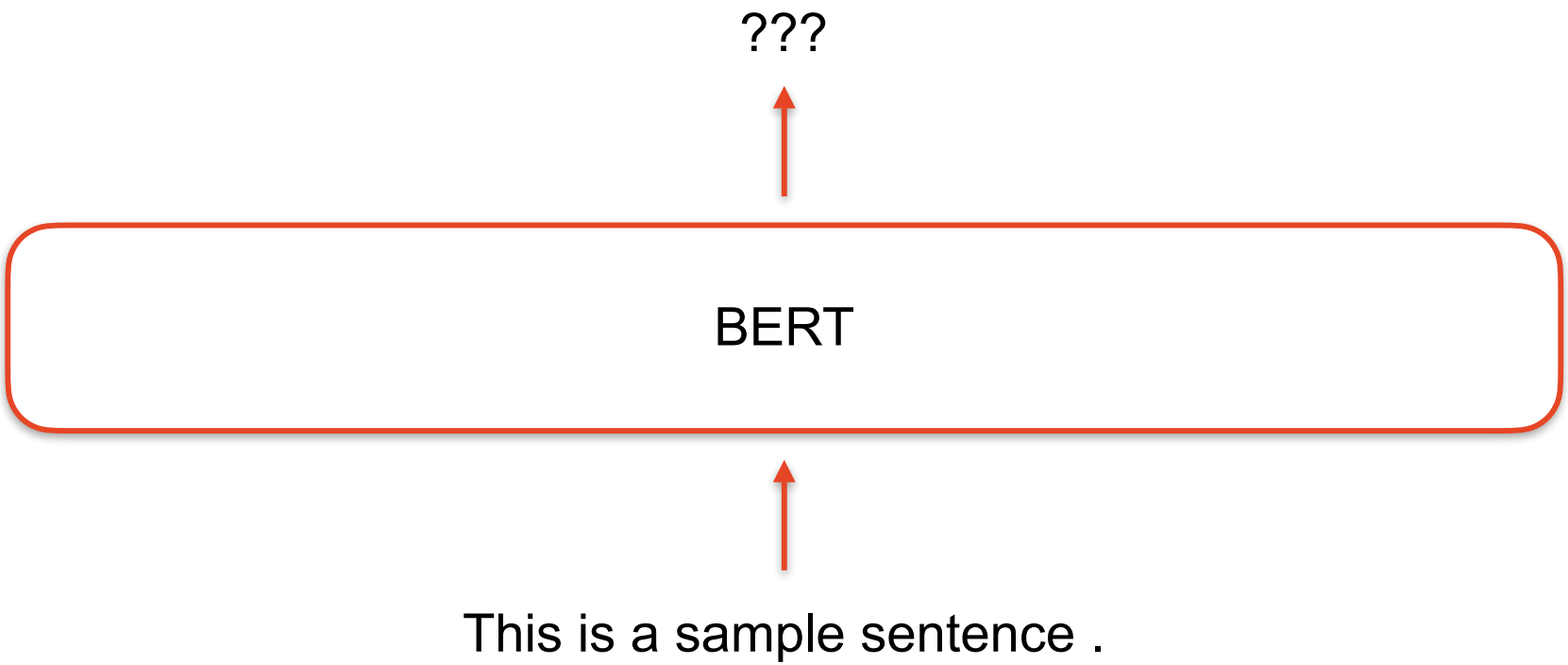
Summarisation with a decoder model



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

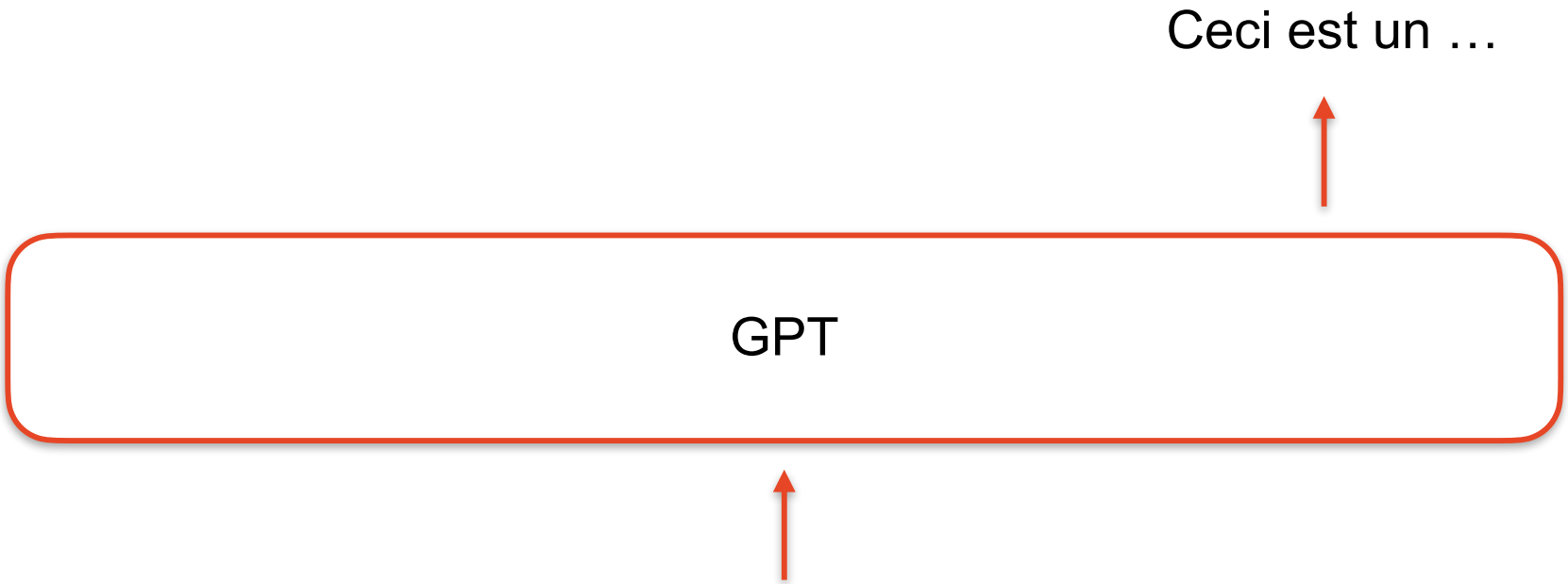
Translation with an encoder model



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Translation with a decoder model



COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

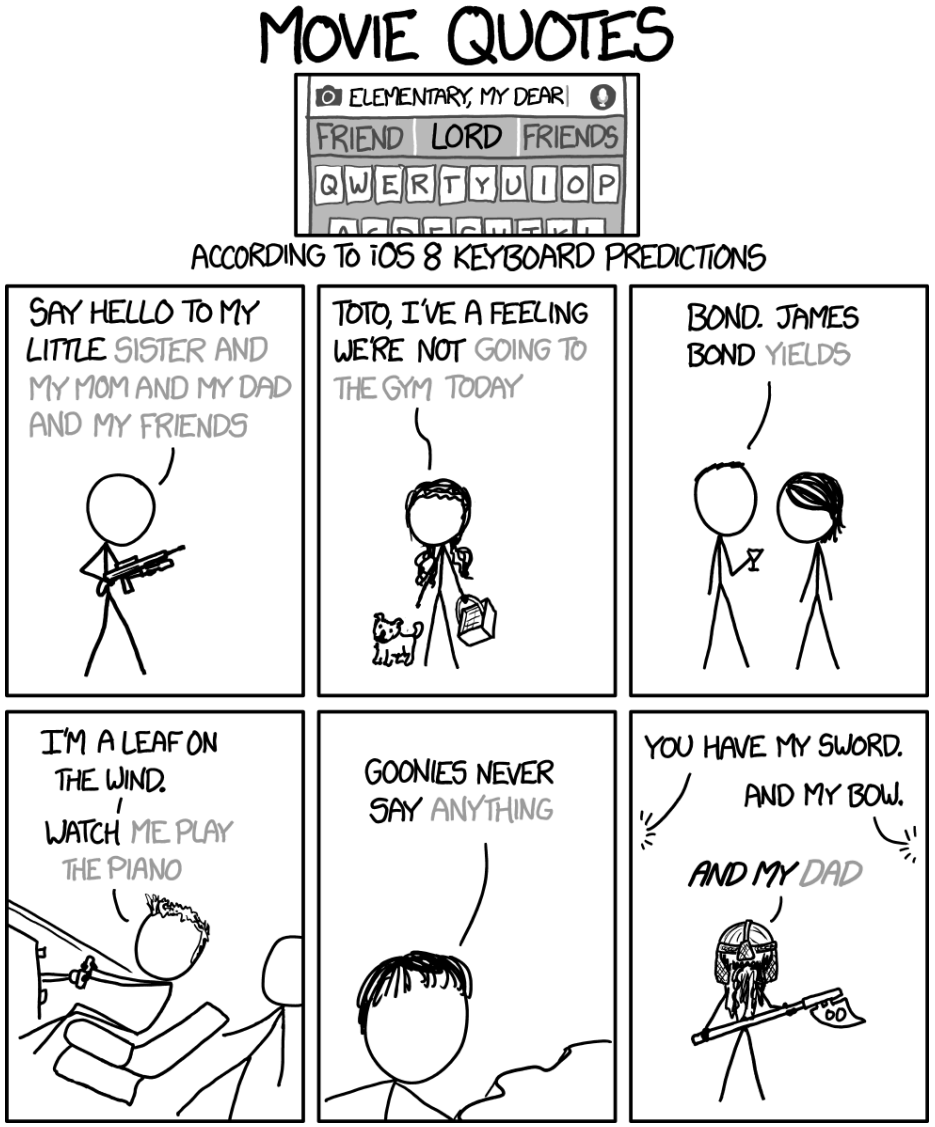
Recap: Using LLMs

Tasks using Language Models: We can use a language model as part of a system to do a task, where the LMs outputs are the input to a task specific model. When we do this, we can keep the LM fixed or fine-tune it. This is similar to how we used word embeddings in earlier lectures.

Tasks as Language Modelling: Many different tasks can be expressed in a way that uses a language model to do the task. Encoder models are somewhat more limited in what they can easily do.

3 minute Break - stretch and visit Menti

menti.com
4182 4438



iOS Keyboard

[More actual results:
"Hello. My name is Inigo Montoya. You [are the best. The best thing ever]",
"Revenge is a dish best served [by a group of people in my room]", and
"They may take our lives, but they'll never take our [money]."]

Source: https://xkcd.com/1427/

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Evaluating LLMs

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can compare language models quantitatively with Mean Reciprocal Rank (MRR)

"This is an example sentence"

Input: <start>
Output: [The, This, NLP, …]

rank = 2

Input: This
Output: [is, book, lecture, …]

rank = 1

…

Input: This is an example
Output: [of, …, sentence, …]

rank = 1 ? 2

$$\text{MRR} = \frac{1}{|\text{samples}|} \sum_{\text{samples}} \frac{1}{\text{rank}}$$

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can compare language models quantitatively with Perplexity

$$Perplexity(text) = P(w_1 w_2 \ldots w_N)^{-\frac{1}{N}}$$

$$= \sqrt[N]{\frac{1}{P(w_1 w_2 \ldots w_N)}}$$

Lower perplexity is better

Lowest possible value is 1 (when the probability is 1)

No upper limit

"N" depends on tokenisation!

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can compare language models quantitatively with Perplexity

0-9

LM$_A$ assigns equal probability to all digits

$$Perplexity(1, 2) = (\frac{1}{10}\ \frac{1}{10})^{-\frac{1}{2}}$$

$$= (\frac{1}{10}^{2})^{-\frac{1}{2}}$$

$$= 10$$

LM$_B$ assigns 50% to the digit 1, equal probability to the rest

$$Perplexity(1, 2) = (\frac{1}{2}\ \frac{1}{18})^{-\frac{1}{2}}$$

$$= (\frac{1}{36})^{-\frac{1}{2}}$$

$$= 6$$

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

MRR and Perplexity are *intrinsic* metrics.
There are also *extrinsic* metrics.

Text

Evaluate (WER)

This is an example

這是一個例子

Evaluate (BLEU)

Change Language Model component

Measure change in metric

Benefits: (1) Measures what we care about
(2) Fewer assumptions about data

Drawbacks: (1) Harder to do
(2) Entanglement with model

More on benchmarks in future!

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Recap: Evaluating LLMs

Mean Reciprocal Rank: A simple way to evaluate predictions, but rarely used.

Perplexity: For a long time, the standard way to evaluate LMs. It is based on the probability the LM assigns to the test text. The equation rescales based on the length of the text, so it depends on the tokenisation used.

Intrinsic vs. Extrinsic Metrics: The metrics above are intrinsic because they use only the LM itself. Similarly, the word analogy task is an intrinsic measure of word embedding quality. In contrast, extrinsic metrics use the LM as part of a system to do a task, which can be more informative. Modern benchmarks blur the lines here because they only use an LM, but the choice of prompt or other details may matter.

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Efficiency

COMP 4446 / 5046
Lecture 8, 2025

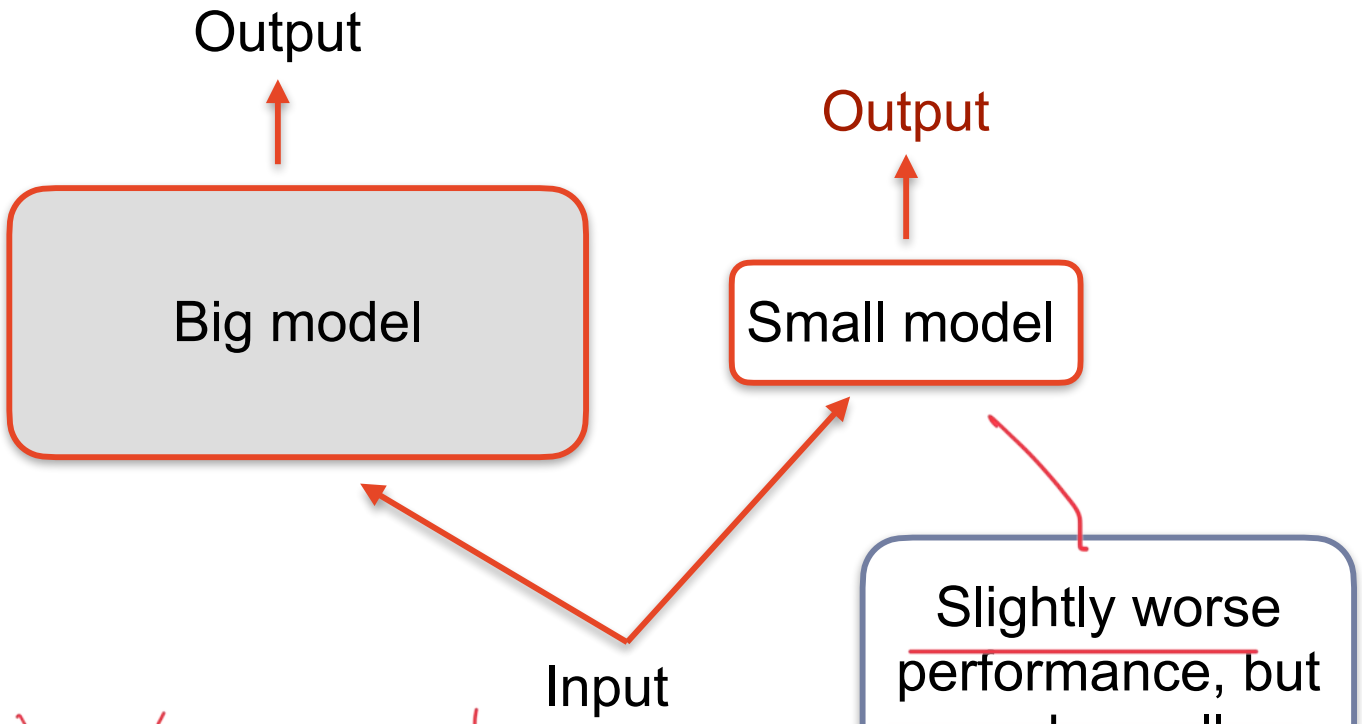
Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can use a big model to train a smaller model to do almost as well

DistilBERT

Δ Guess vs. Answer



用 big train small
让 small 达到和 big 一样的效果

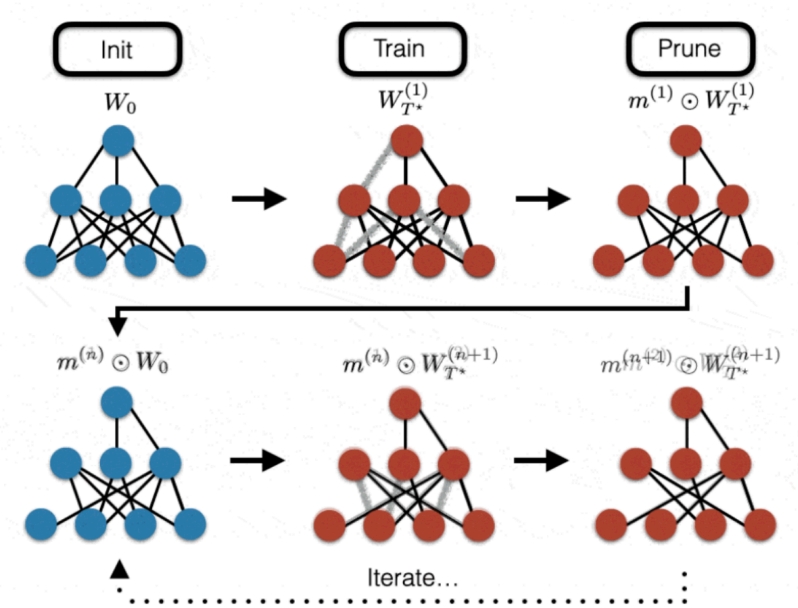
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can prune a big model and still get the same accuracy

Adding sparsity



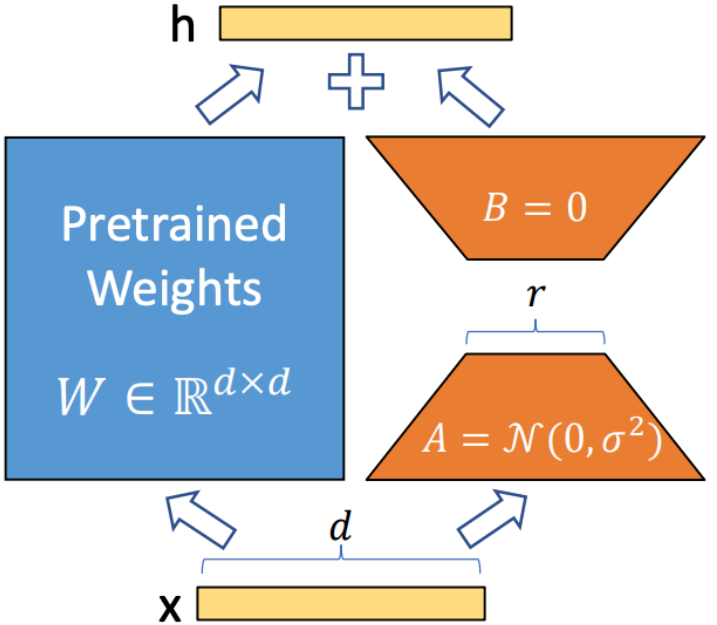
Berkeley NLP Course

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can fine-tune a big model more efficiently

LoRA



low Rank approximation

Hu et al. (2021)

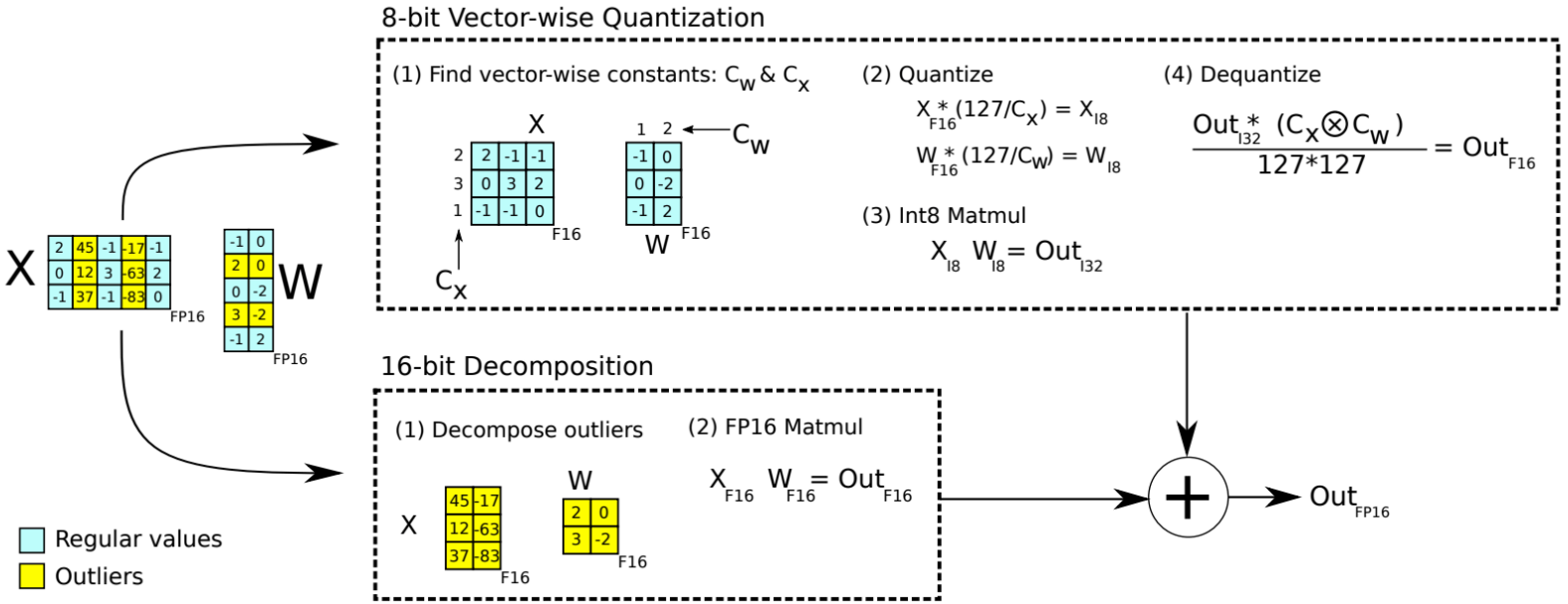
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can use a reduced numerical precision version of the model

LLM.int8



Dettmers et al. (2022)

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can use a reduced numerical precision version of the model

LLM.int8

			Largest Model that can be run	
Class	Hardware	GPU Memory	8-bit	16-bit
Enterprise	8x A100	80 GB	**OPT-175B / BLOOM**	**OPT-175B / BLOOM**
Enterprise	8x A100	40 GB	**OPT-175B / BLOOM**	OPT-66B
Academic server	8x RTX 3090	24 GB	**OPT-175B / BLOOM**	OPT-66B
Academic desktop	4x RTX 3090	24 GB	**OPT-66B**	OPT-30B
Paid Cloud	Colab Pro	15 GB	**OPT-13B**	GPT-J-6B
Free Cloud	Colab	12 GB	**T0/T5-11B**	GPT-2 1.3B

Dettmers et al. (2022)

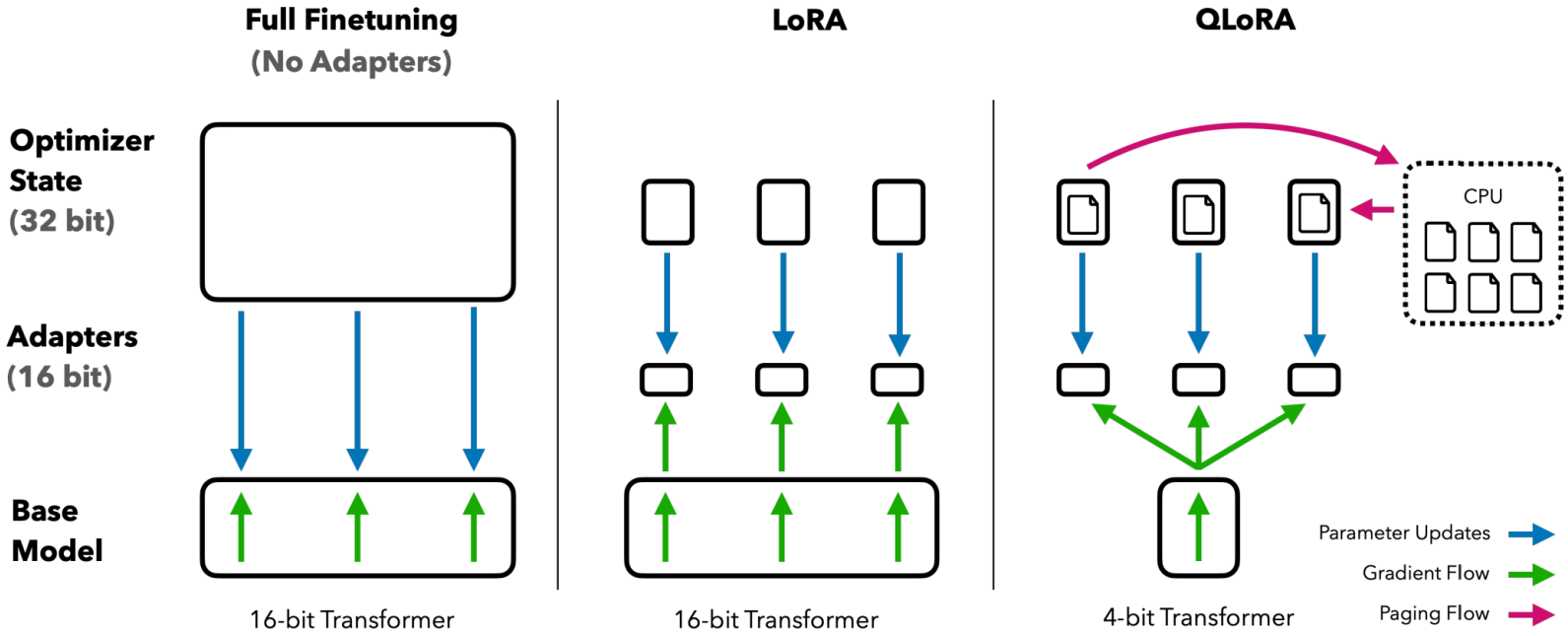
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can combine these ideas

QLoRA



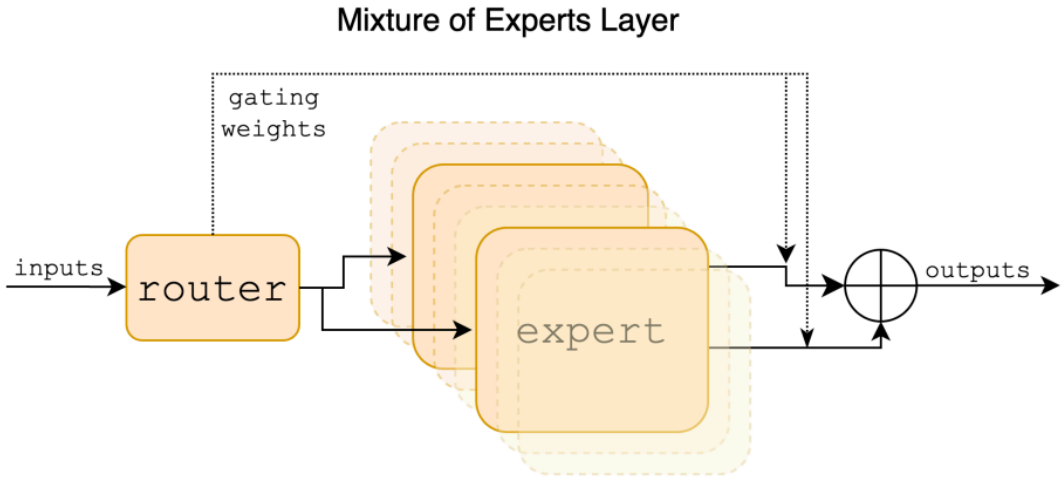
Dettmers et al. (2023)

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

We can run a set of smaller models together

SMoE



Jiang et al. (2024)

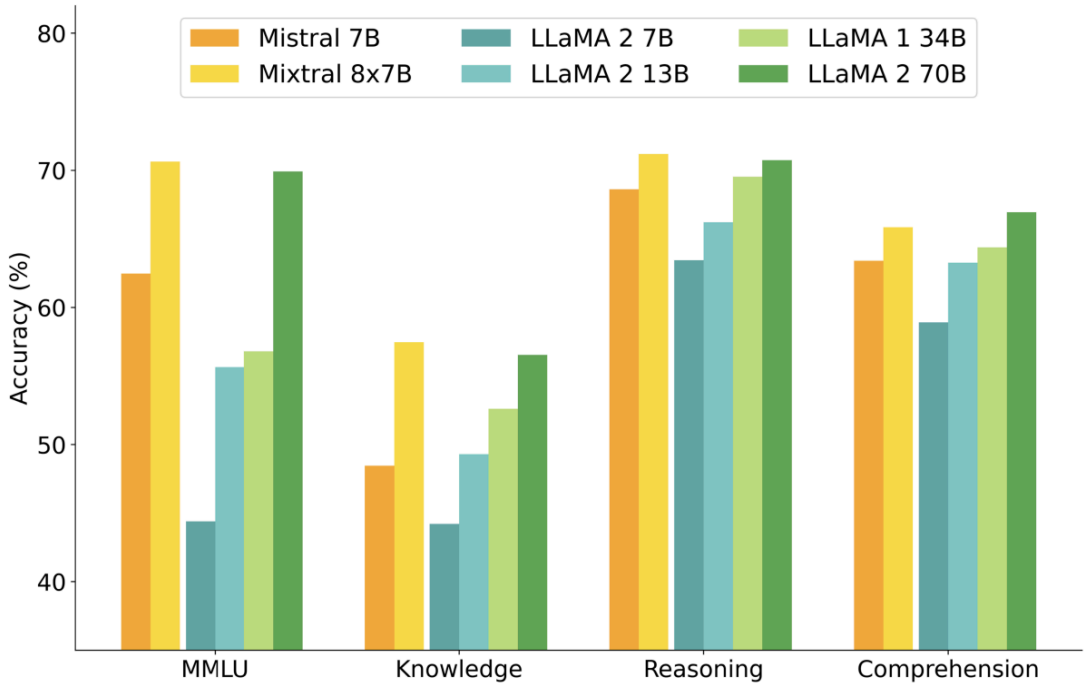
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

We can run a set of smaller models together

SMoE



Jiang et al. (2024)

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Recap: Efficiency

Reducing model size: To make models smaller we can use a large model to train a small model (distillation) or we can prune parts of a large model. Pruning is hard to do in a way that is computationally useful, so distillation is much more common

Increasing training memory efficiency: During training, fitting all the updates to the weights in memory can be expensive. Methods like LoRA allow us to approximate the changes, enabling training on lower memory GPUs.

Numerical approximation: We do not need full precision for our models. Reducing numerical precision can save GPU memory.

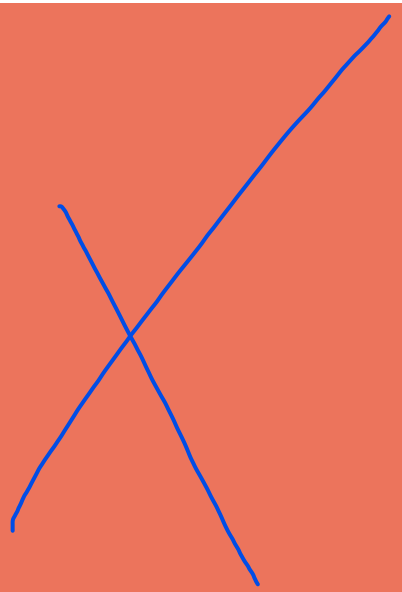
Mixtures of models: To improve performance we can make models that are composed of a set of smaller models. Only one (or a few) models are active at a time.

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Other Models



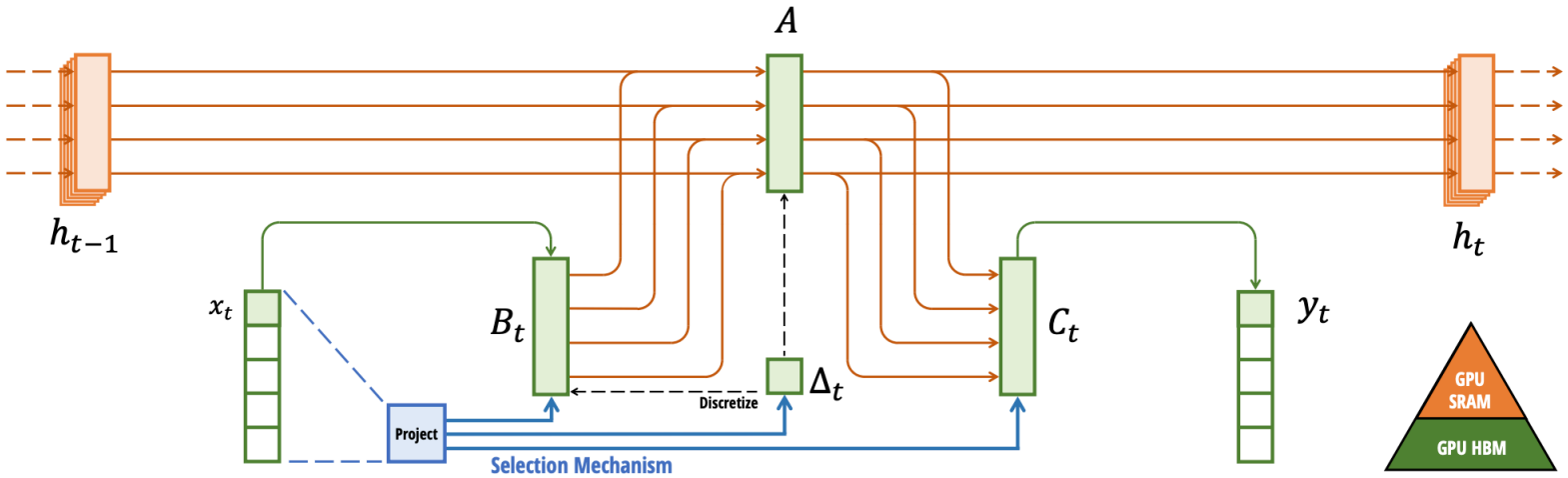
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What about models that don't use the transformer?

Mamba



Gu and Dao (2023)

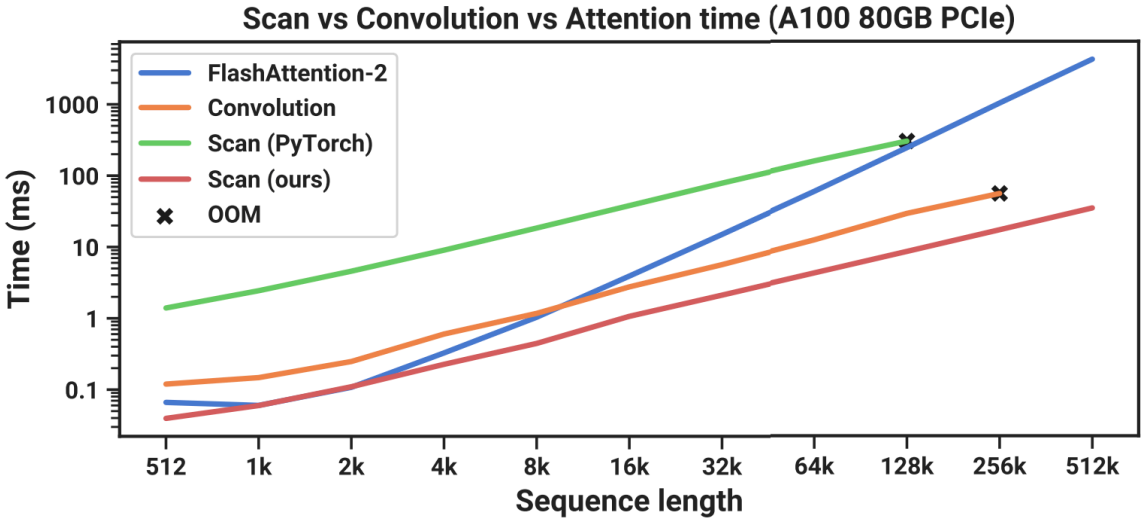
COMP 4446 / 5046
Lecture 8, 2025

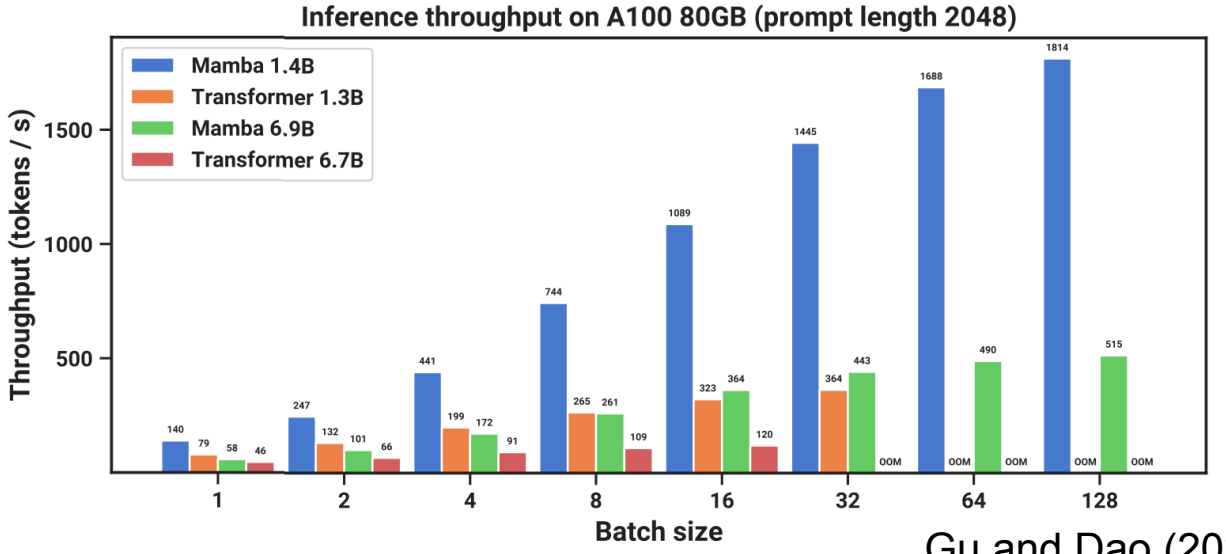
Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What about models that don't use the transformer?

Mamba





Gu and Dao (2023)

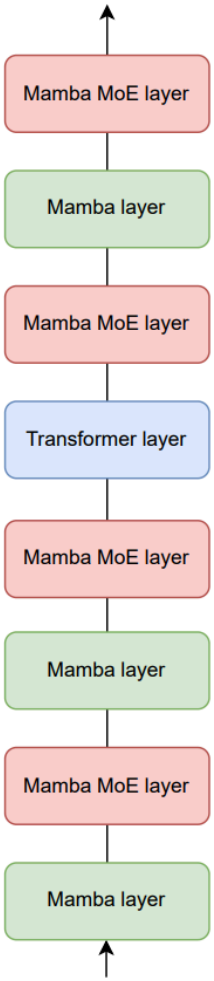
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

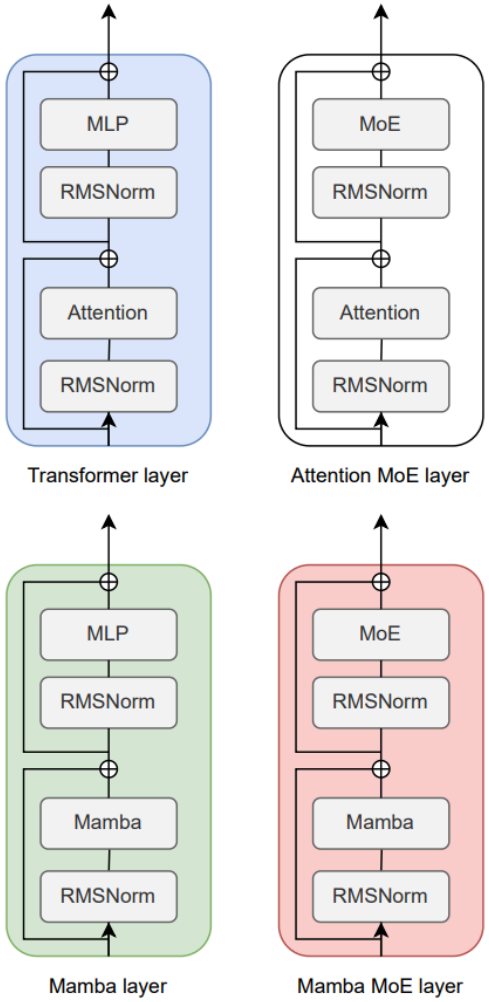
menti.com 4182 4438

What about models that don't use the transformer?

Jamba



(a) Jamba block



(b) Different types of layers

Lieber et al. (2024)

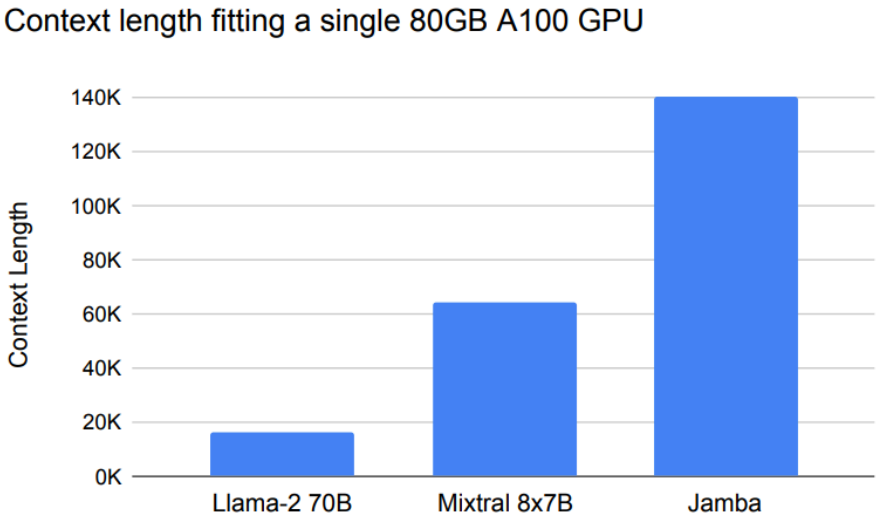
COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

What about models that don't use the transformer?

Jamba



Lieber et al. (2024)

COMP 4446 / 5046
Lecture 8, 2025

menti.com 4182 4438

Recap: Other Models

State space models: Sequential models, e.g., RNNs, may seem out of date, but there is still active research on variants. In this section we saw one, Mamba, but there is also RWKV, the xLSTM, and others. None of them are competitive with transformers on accuracy yet, but they can handle long contexts and be very fast.

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

Workshop Preview

COMP 4446 / 5046
Lecture 8, 2025

Language Models
Training LLMs
Using LLMs
Evaluating LLMs
Efficiency
Other Models
Workshop Preview

menti.com 4182 4438

HuggingFace

Research Study Survey

https://redcap.sydney.edu.au/surveys/?s=KENPXJLDHDF37PKE

Optional - see Ed post for details

Muddy Card

https://saipll.shinyapps.io/student-interface/

If you do not wish to participate in the study, use the Ed form instead

Go to Ed → Lessons → Muddy Cards Lecture 8