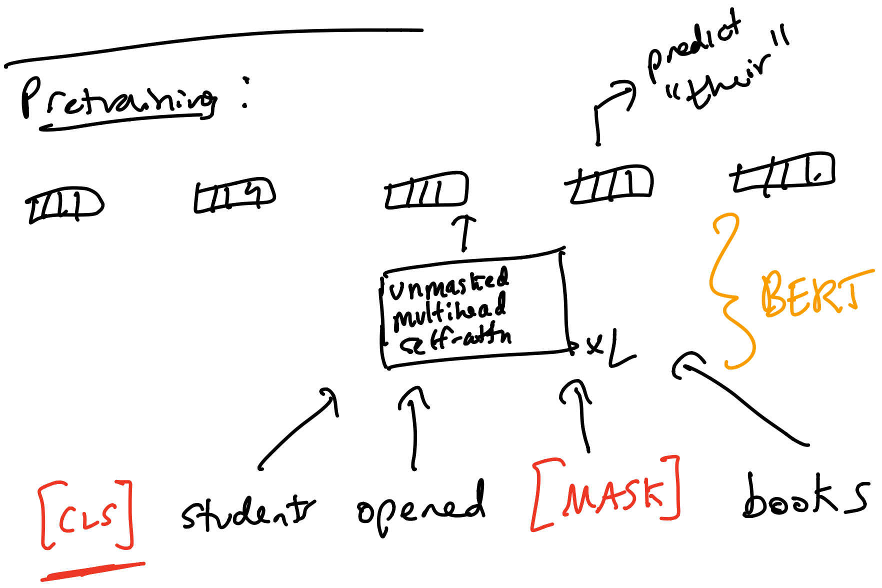
BERT:

↳ example of the encoder-only paradigm

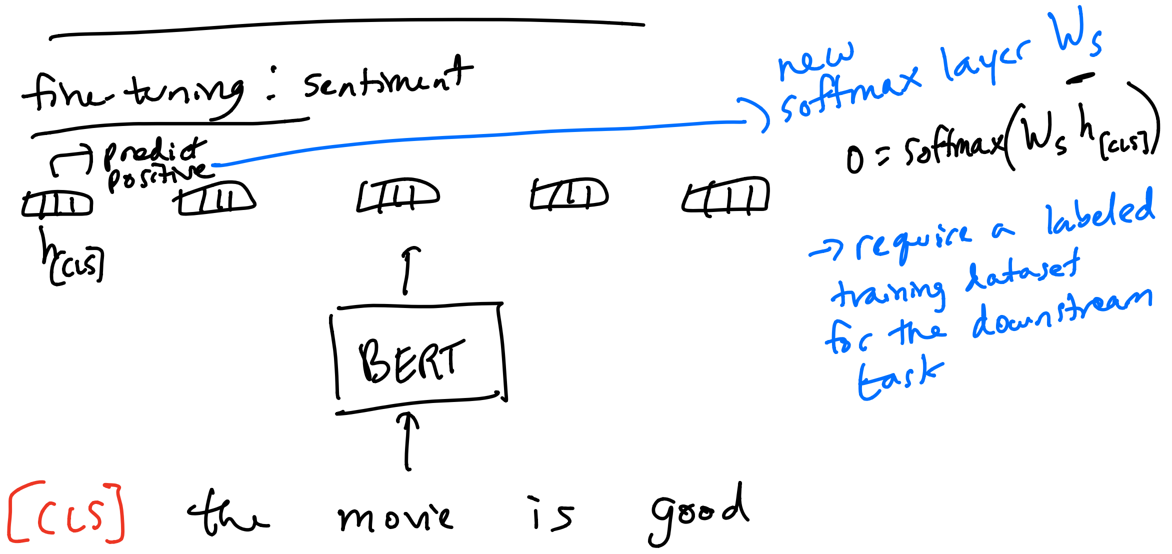
↳ Pretraining: train w/ self-supervised obj called "masked LM"

↳ Finetuning: process of adapting the pretrained model to a particular downstream task

Pretraining:



fine-tuning: sentiment



predict positive
$h_{[CLS]}$
BERT
[CLS]
the
movie
is
good
new softmax layer W_s
$o = \text{softmax}(W_s h_{[CLS]})$
→ require a labeled training dataset for the downstream task

extractive QA

↳ inputs: + a question
+ a paragraph that contains the answer

↳ output: a span of the paragraph that answers the question

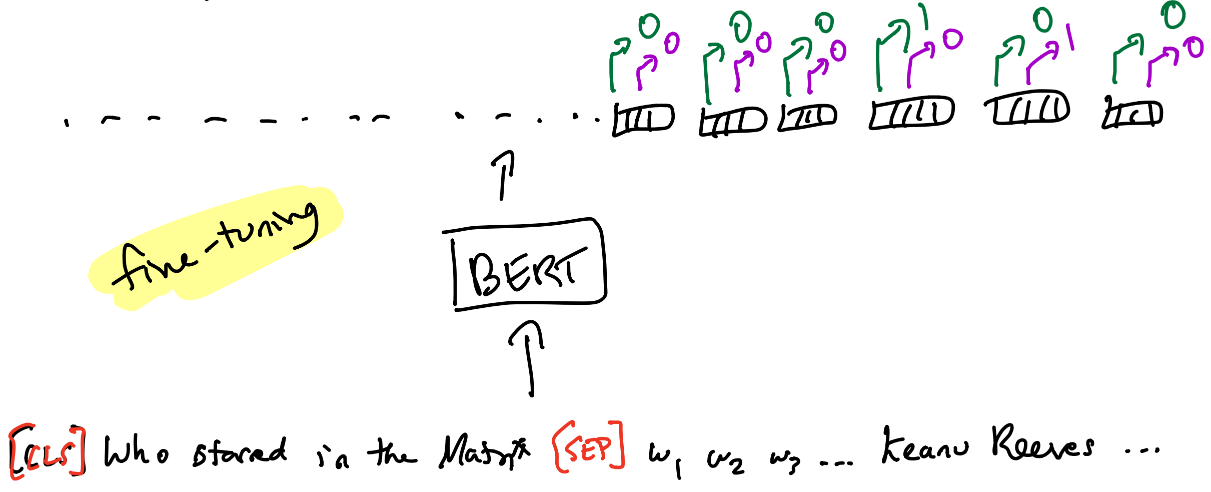
↳ datasets: SQuAD v1,2; QuAC/CoQa, HotpotQA, ...

Q: Who starred in the Matrix as Neo?

P: w_1 w_2 w_3 ..... Neo was played by actor $_i$Keanu Reeves$_j$ ... . .

A: (i, j)

How do we use BERT for extractive QA?
↳ 2 softmax layers on each token in passage
↳ predict beginning, end index of answer span



0 0
0 0
0 0
0 1
0 1
0 0
BERT
fine-tuning
[CLS] Who stared in the Matrix [SEP] w_1 w_2 w_3 ... Keanu Reeves ...

how do we select an answer span at test time?

→ find the span $w_{i \ldots j}$ that maximizes

$P_{START}(i) \cdot P_{END}(j)$

↳ exclude spans where $j < i$

↳ exclude spans longer than a threshold

---

advanced variants of BERT:

↳ pretraining improvements ⇒ RoBERTa
  ↳ more data

↳ longer sequences during pretraining
- BERT ⇒ 512 tokens max
- XLNet ⇒ 900 tokens

↳ more pretraining objectives

↳ ELECTRA

↳ → "Students opened [MASK] books"

↳ "students opened monster books"

$$\begin{array}{ccccc} \text{students} & & \text{opened} & \text{monster} & \text{books} \\ \swarrow \quad \searrow & & \downarrow & \downarrow & \downarrow \\ \text{real?, corrupted?} & & \text{real} & \text{corrupted} & \text{real} \end{array}$$

↳ smaller models

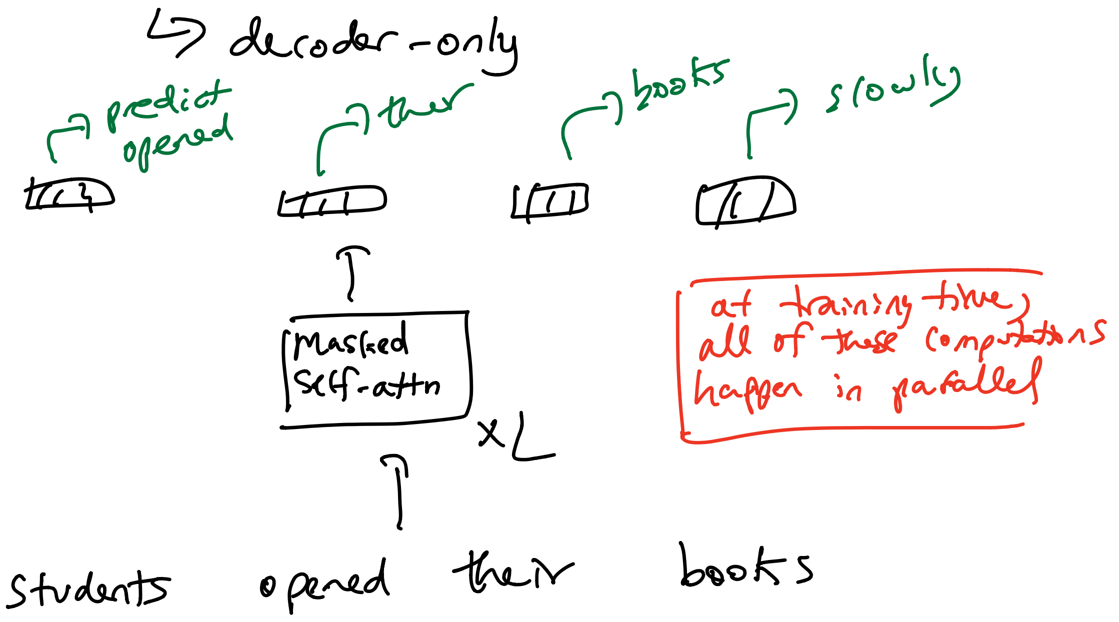
↳ tinyBERT, distilBERT, ALBERT

↳ distillation

↳ pruning

Transformer LMs at test-time:

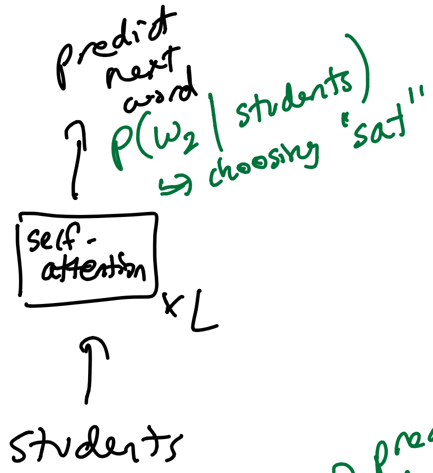
↳ decoder-only



⇒ those four predictions can happen simultaneously b/c we already know the identity of the gold next tokens
ground-truth

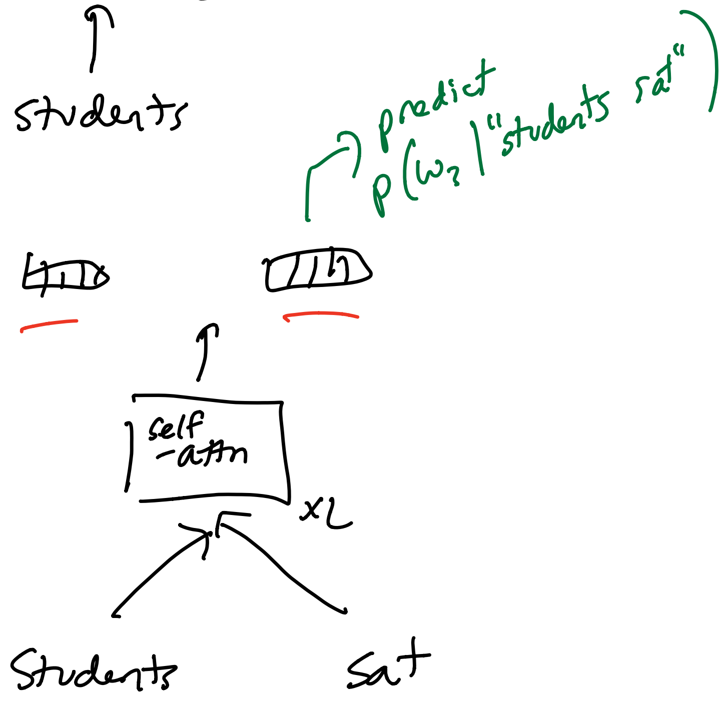
test-time:

timestep 1:



autoregressive decoding / LM

1. first have to choose a next word from $p(w_2 \mid \text{students})$
   - ↳ sample
   - ↳ argmax
2. concat this word to the input and then repeat step 1



at test-time, I have to <u>decode</u> the output word-by-word, because I don't have access to the gold next tokens

↳ decoding algorithms