8.7

A Large Language Model for Classification

Part 1: Bidirectional Pretraining with BERT

Sebastian Raschka and the Lightning AI Team

GPT Recap

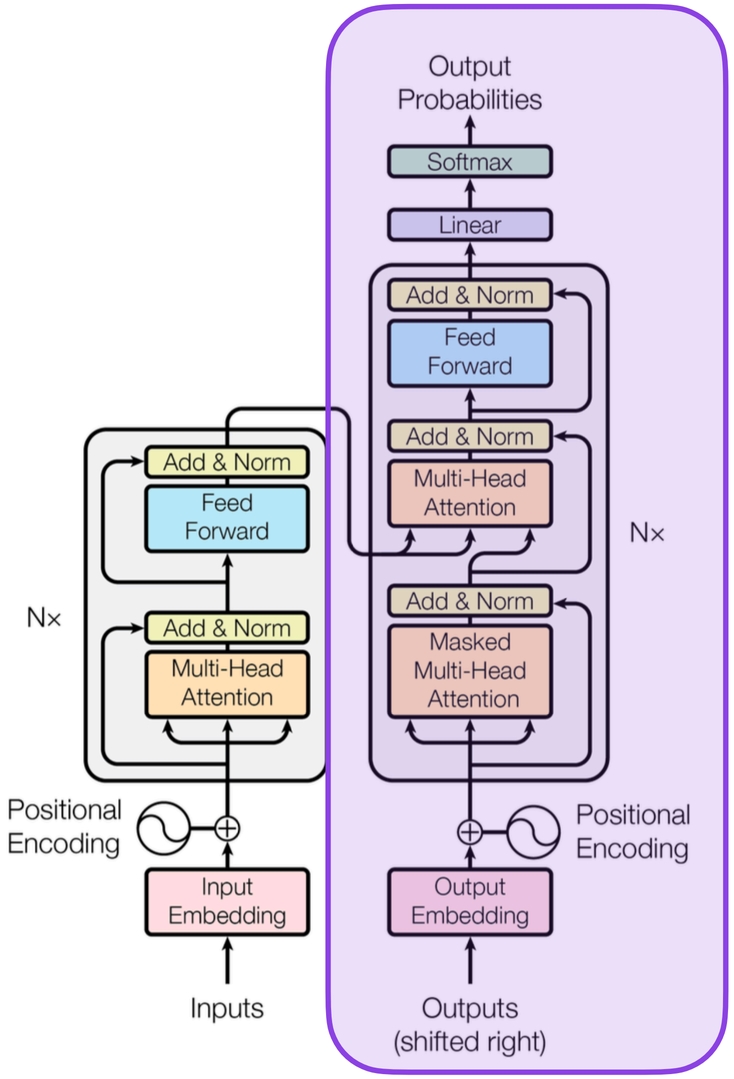
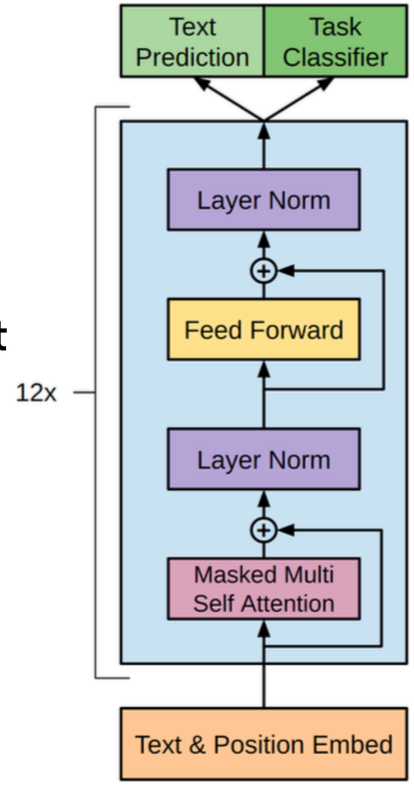


Figure 1: The Transformer - model architecture.

GPT is essentially the decoder part of the original transformer



https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

GPT Recap

Feed model text from left to right, and it learns to predict the next word.

GPT Recap

Self-supervised pretraining

Step 1: pretrain ⟶ Predict next word (unidirectional self-attention)

Step 2: fine-tune

BERT

Self-supervised pretraining

Step 1: pretrain → Predict next word (~~unidirectional~~ self-attention)

a) Predict randomly masked words (bidirectional / nondirectional)

b) Sentence-order prediction

Step 2: fine-tune

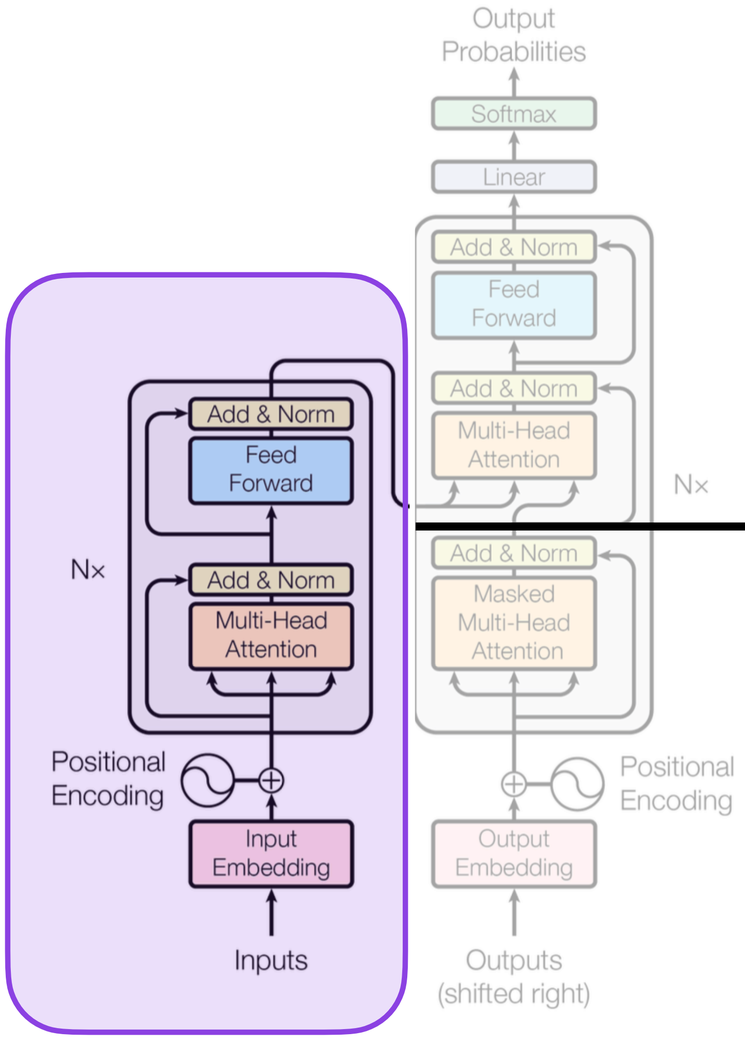
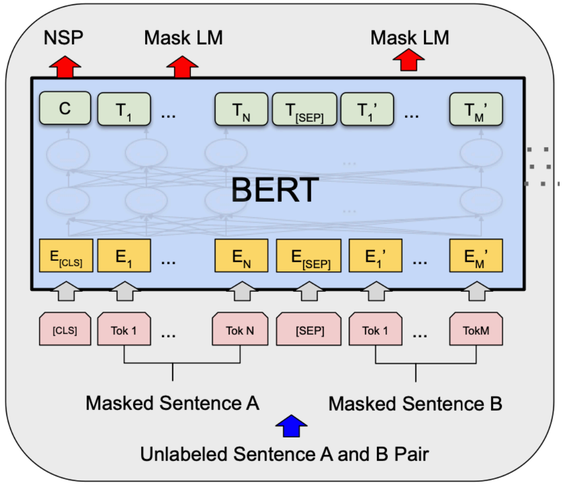


Figure 1: The Transformer - model architecture.

BERT is essentially the **encoder** part of the original transformer



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, https://arxiv.org/abs/1810.04805

Step 1: pretrain on large unlabeled dataset (learn a general language model)

a) Predict input sentence given randomly masked words

Input sentence: *The curious kitten deftly climbed the bookshelf*

Pick 15% of the words randomly

The curious kitten deftly climbed the bookshelf

Input sentence: *The curious kitten deftly climbed the bookshelf*

Pick 15% of the words randomly

The curious kitten deftly climbed the bookshelf

- **80% of the time, replace with [MASK] token**
- **10% of the time, replace with random token (e.g. ate)**
- **10% of the time, keep unchanged**

Step 1: pretrain on large unlabeled dataset (learn a general language model)

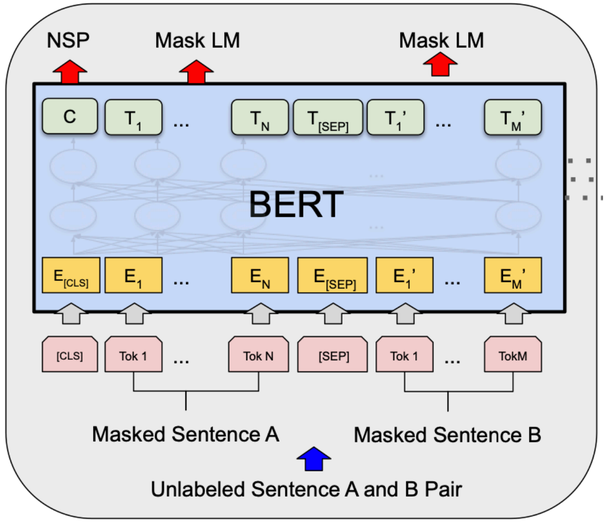
a) Predict input sentence given randomly masked words

b) Predict sentence order

b) Predict sentence order

[CLS] Sentence A [SEP] Sentence B

Placeholder for the IsNext=True / False label in the decoder output



b) Predict sentence order

[CLS] Toast is a simple yet delicious food [SEP] It's often served with butter, jam, or honey.

IsNext = True

[CLS] It's often served with butter, jam, or honey. [SEP] Toast is a simple yet delicious food.

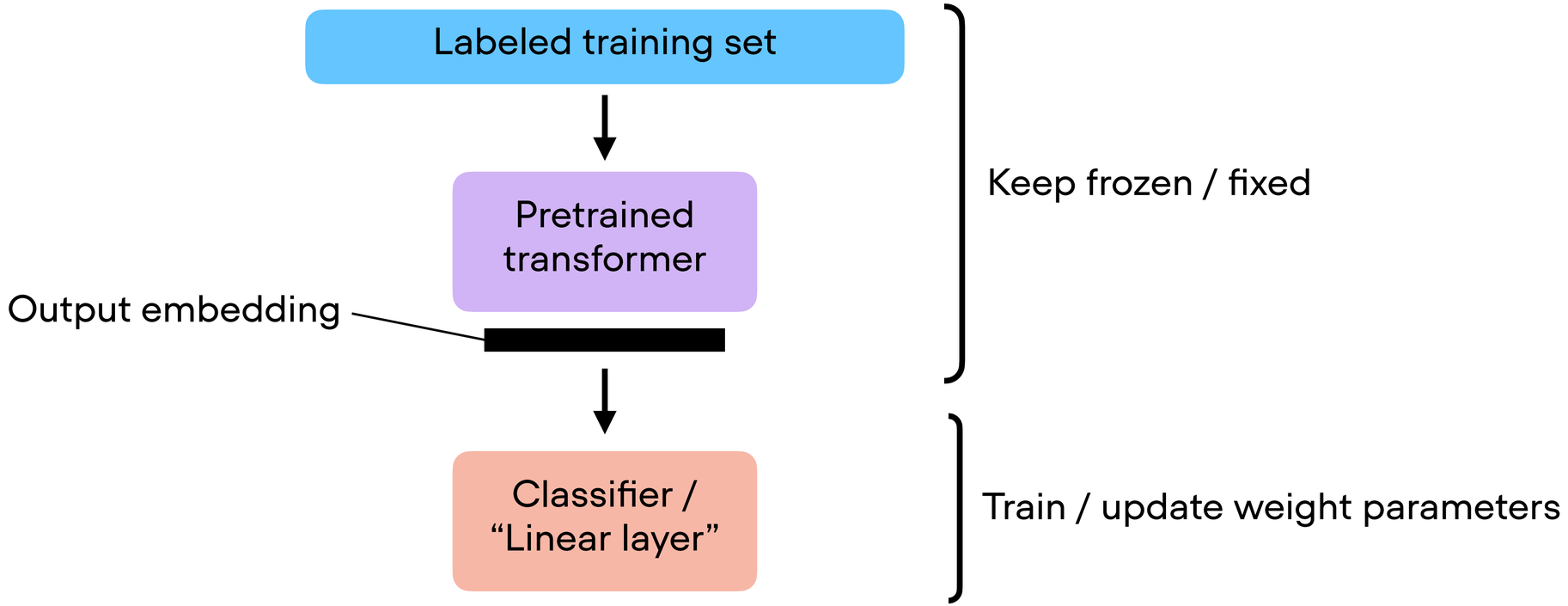
IsNext = False

2 ways of adopting a pretrained transformer

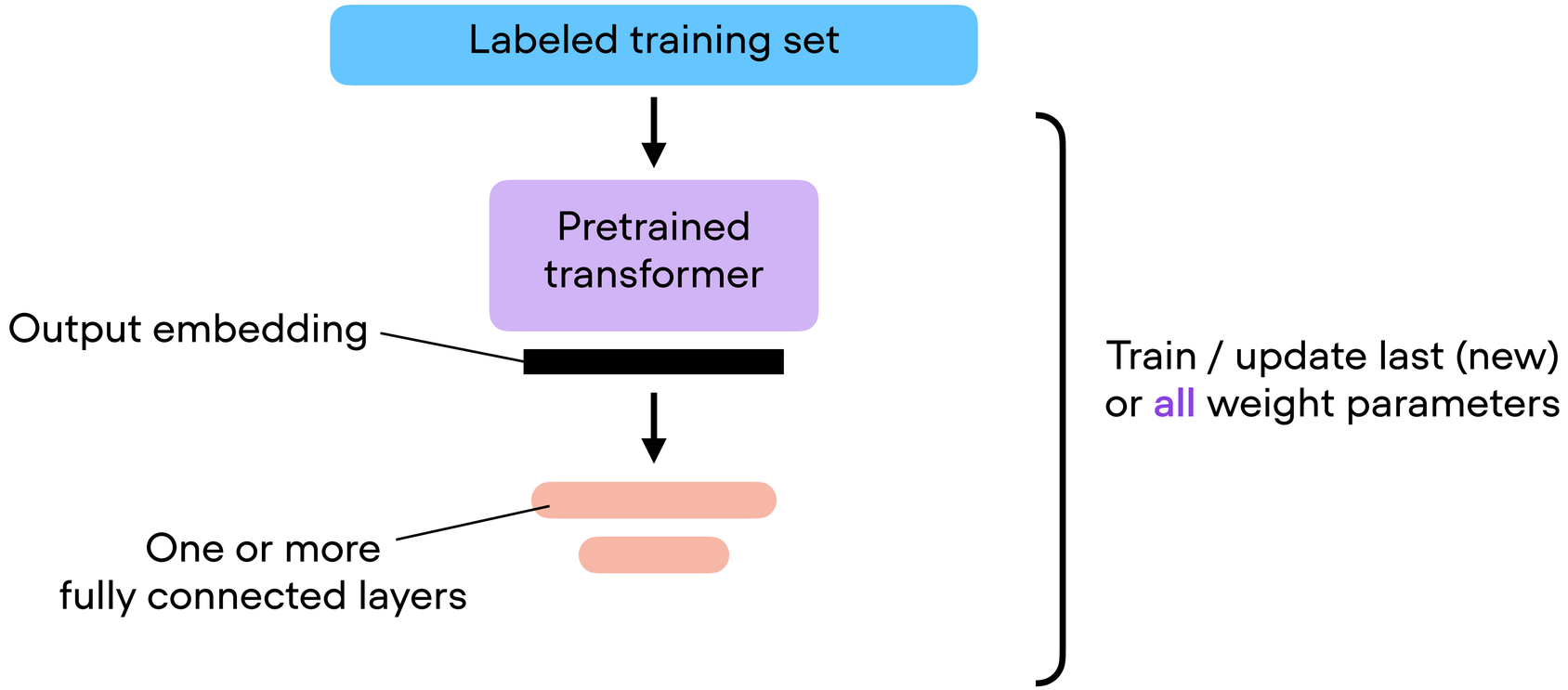
1) Feature-based approach

2) Fine-tuning approach

1) Feature-based approach



1) Fine-tuning approach



Next: Fine-tuning a BERT model in PyTorch