Using the Transformer

BERT (Devlin et al., 2018)

Learning goals

- Understand the use of the transformer encoder in this model
- Understand the iarchitectural components

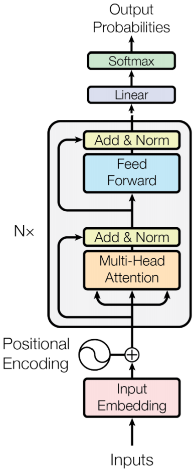
KEY FACTS ON BERT ▸ DEVLIN ET AL. (2018)

***B**idirectional **E**ncoder **R**epresentations from **T**ransformers:*

- Bidirectionally contextual model
- Introduces new self-supervised objective(s)
- Completely replaces recurrent architectures by Self-Attention
  $+$ simultaneously able to include bidirectionality
- Transformer *encoder* as backbone of the architecture
  - 12 (24) Transformer encoder blocks
  - Embedding size of $E = 768$ (1024)
  - Hidden layer size $H = E$
  - $A = H/64 = 12$ (16) attention heads
  - Feed-forward size is set to $4H$

  $\rightarrow$ 110M (340M) parameters in total for $BERT_{Base}$ ($BERT_{Large}$)

CORE OF BERT – THE TRANSFORMER ENCODER



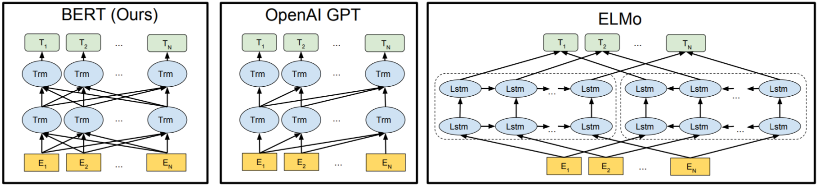
Source: Vaswani et al. (2017)

A REMARK ON "CAUSALITY"

Causality is an issue!

- Goal: Learn contextual representations for words/tokens
- *Self-Supervision:* Input and target sequence are the same
  $\rightarrow$ We modify the input to create a meaningful task
- Unconstrained Self-Attention makes using the LM objective infeasible
- Bidirectionality at a lower layer would allow a word to see itself at later hidden layers
  $\rightarrow$ The model would be allowed to cheat!
  $\rightarrow$ This would not lead to meaningful internal representations

ELMO VS. GPT VS. BERT



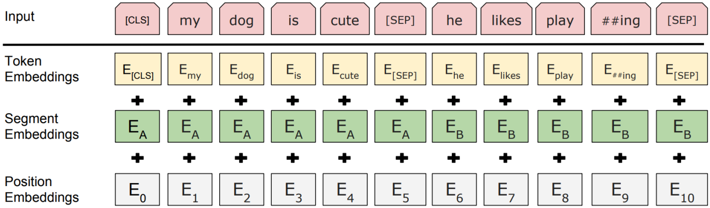
Source: Devlin et al. (2018)

Major architectural differences:

- ELMo uses two separate unidirectional models to achieve bidirectionality $\rightarrow$ Only "*shallow*" bidirectionality
- GPT is not bidirectional, thus no issues concerning causality
- BERT combines the best of both worlds:

$$\textit{Self-Attention} + (\textit{Deep})\ \textit{Bidirectionality}$$

INPUT EMBEDDINGS



Source: Devlin et al. (2018)

- Two concatenated sentences as input
- WordPiece tokenization (Wu et al., 2016) for the inputs
  $\rightarrow$ Vocabulary of 30.000 tokens
- Learned segment + position embeddings
- Special `[CLS]` and `[SEP]` tokens