StatsQuest · Decoder Only Transformations

Masked self-Attention words contain input from all of the other words that came earlier → Context

{ Masked Attention cells }
Applied to positin encode value

↳ into residual connections

these residual numbers represent ~~words~~ A word

We want to use these numbers to predict the next word

Decode-only transformer

We want to build something that can both encode the prompt + generate output

word → embed → Mask self-Att → Residual → MLP → 🎯

↳ MLP Predicts the next word
- Possible outputs could be anything in the vocab

▶ in the orginal paper, the MLP is the word embed network x weights in reverse
(not all do this way)

Diff between tranforner & decode only

Decode only uses the same components to encode & generate output

Same cells, weights & biases

Masked self-Attention is ~~a~~ calculate to each current word & everything proceeding it

Self-Attention is applied equally to the input prompt & output generates

Mask Self-Att a Decoder-only to determin how words in the prompt are relate

& keep track of important input words when generating the output

- Basic transformers use:
  - encoder to learn prompt
  - Decoder to generate output
  - uses Self-Att, not mask Self-Att
    - ↳ learns how all words in prompt are related
    - ↳ Masked just uses words before

Normal transformer uses Masked in the Decoder only