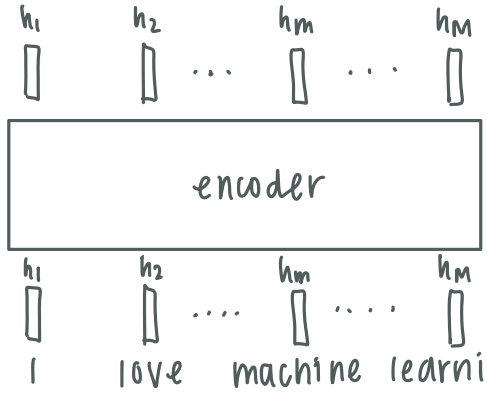
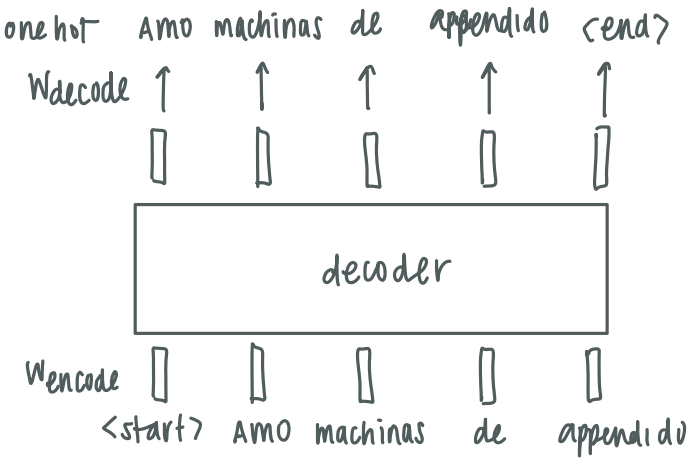
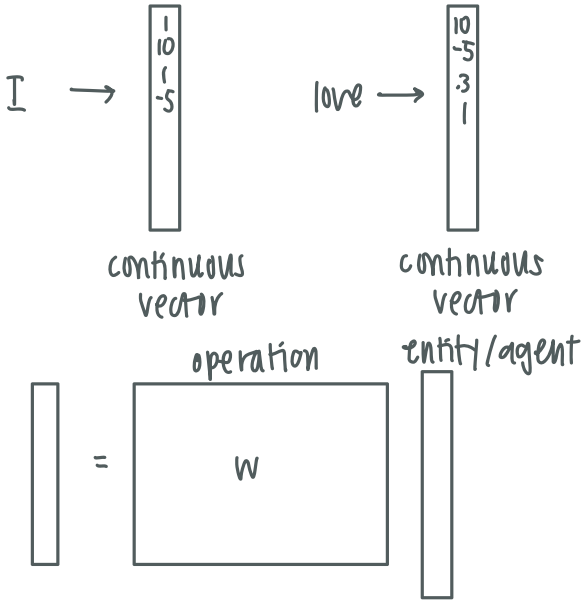
Transformer : self attention



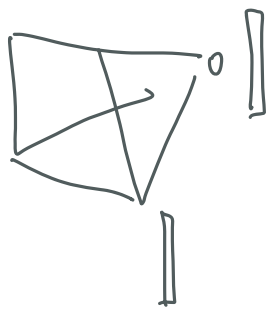




problem w/ LSTM: need to do in a sufficient way, slow

Embedding



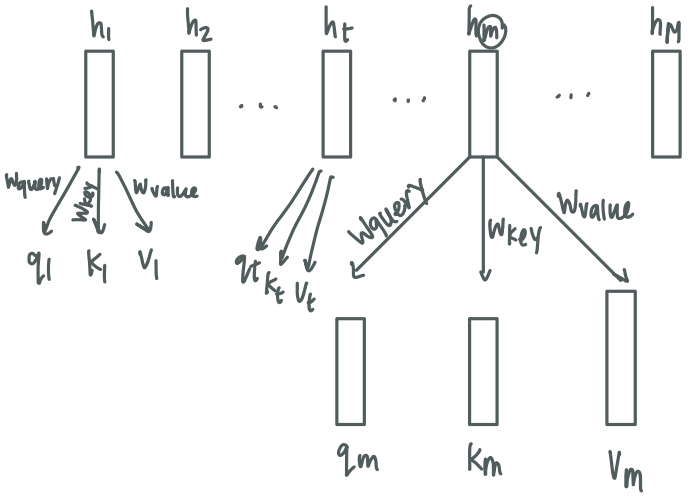

graph



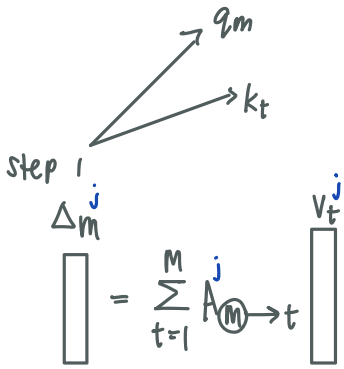

Attention softmax

$$A_{m \to t} = \frac{\exp(\text{affinity}(h_m, h_t))}{\sum_{t'=1}^{M} \exp(\text{affinity}(h_m, h_{t'}))}$$

$$\sum_{t=1}^{M} A_{m \to t} = 1$$

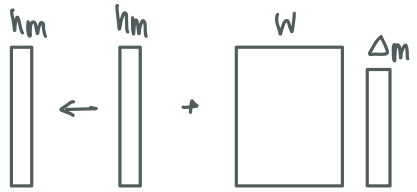



Affinity $(h_m, h_t) = \langle q_m, k_t \rangle / \sqrt{dim}$




- higher weight given to value vector w/ more attention
- yields weighted average

* j = multihead subscript, see 'multihead' below

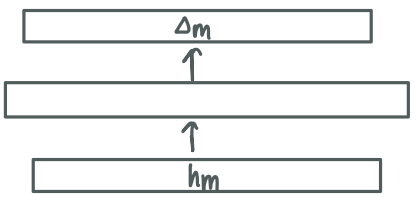



residual

contextualize → each vector has info from all other vectors

step 2

Each $h_m$ alone

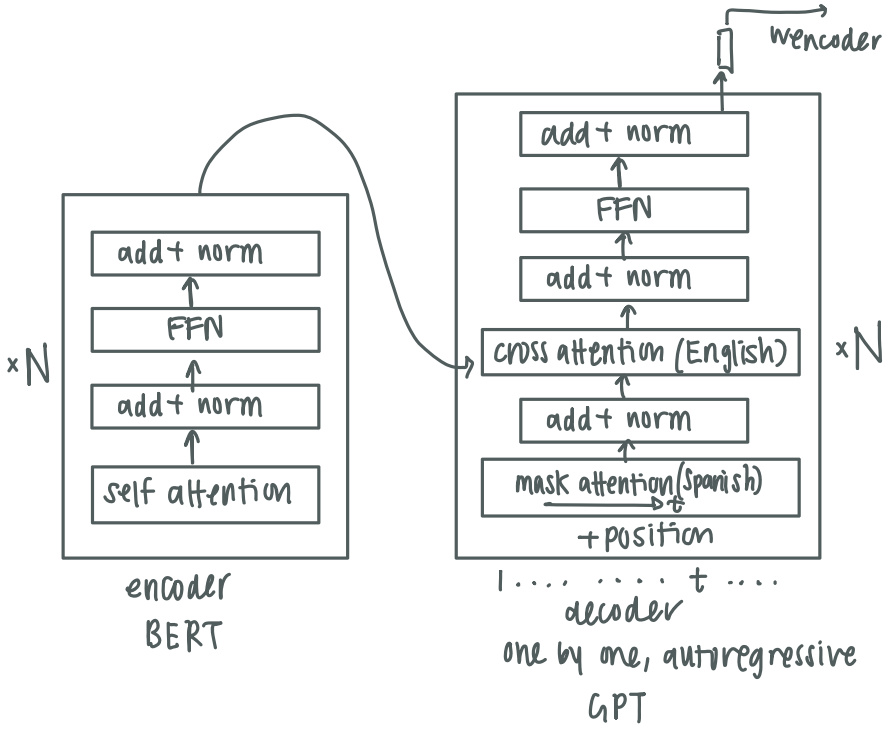



feed forward network (FFN)

$h_m \leftarrow h_m + \Delta_m$ residual

layer normalization after each residual (not batch normalization) [so, after step 1 & step 2 residuals)


normalization → to mean 0 variance 1




GPT-3 170 billion

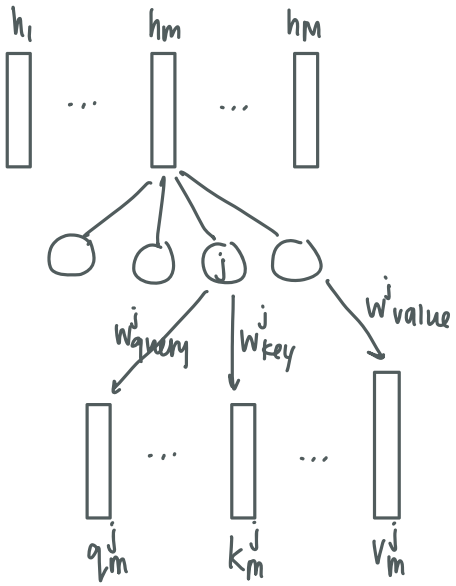
PaLM 540 billion

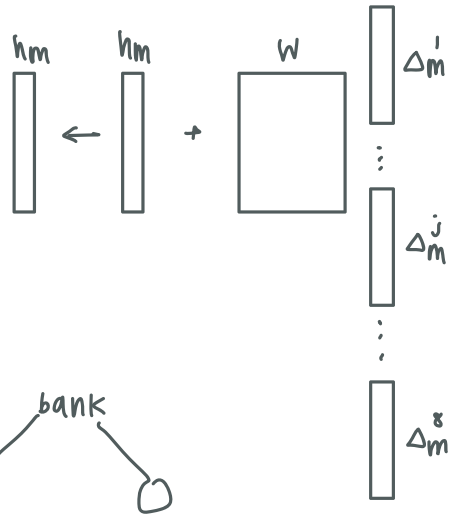
scaling law:
↑ size of model
↑ performance


parent

train AR model conditioned on the prompt.
doesn't need to be retrained due to AR nature

Multihead

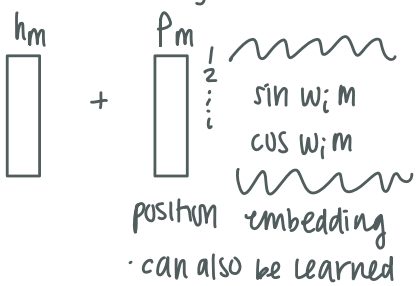



e.g.




(initial) embedding




Fourier basis

position embedding
· can also be learned

Decoding