PAGE
DATE

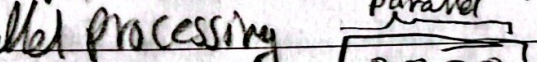
(week 4)

Transformers network intuition

- RNN → GRU → LSTM

The more increase complexity. & these model ingested the input word by word (or) one token at time

- In the transformer architecture, it allows us to run alot more of these computations for an entire seq. in parallel so we can ingest an entire seq all at the same time
- The major innovation of the transformer architecture is combining the use of Attention based representation & CNN Attention + CNN style. The Attention network is a way of computing very rich & useful representations of words but with something more akin to the CNN style of parallel processing



- Self attention The goal of it is if we have a sentence of 5 words $A^{<1>} \cdots A^{<5>}$ we will compute 5 representations for these 5 words. & these representations are the attension based way of computing representation for all the words in your seq. in parallel

- Multihead attention is for loop over self attention processes. So it end up with multiple versions of these representations

→ **Self Attenstion:**

• to use Attenstion with a style more like CNN we need to Calculate self attension where we create attension-based representations for each of the words in your input sentence
• after making the representations we call the five repres. $A^{<1>}$, $A^{<2>}$ -- $A^{<5>}$

sentence: Jane visite l'Afrique en Septembre
$X^{<1>}$ $X^{<2>}$ $X^{<3>}$ $X^{<5>}$

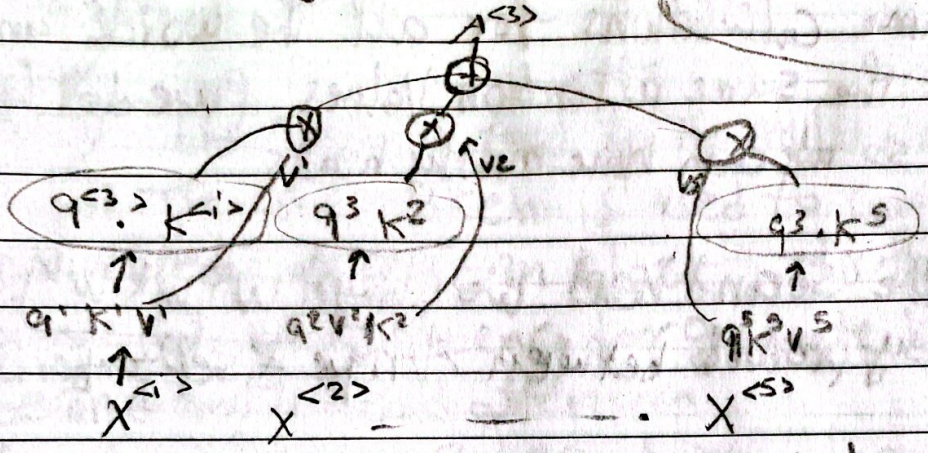
• if we take $X^{<3>}$ (How self attention allow computing $A^{<3>}$) depending on how we are thinking of $X^{<3>}$ we will choose to represent it differently by looking to the surrounding words.
we will compute these representations in parallel for all 5 words
• what is the computations needed to go from $A^{<3>}$ → self atten. representation of $X^{<3>}$. First we associate each of the words with three values query key & value pairs, so if $X^{<3>}$ is the word embedding for "L'Afrique" the way q_3 is computed is as a learned matrix $q^{<3>} = W^Q * X^{<3>}$ & similar with key & value
$'' = W^K * ''$
$'' = W^V * ''$

These matrices $W_Q \cdot W_K$ & W_V are the parameters of the learning algorithms

• q_3 is might be a question ex: what happening there? (l'Afrique) & what we going to do is compute the inner product bet q_3 & K^1, q^3 & K^2 ---- q^3 & K^5 & each one will tell us how good an answer of K^1 to the question q_3 & the goal of this operation is to pull up most info that's needed to compute most useful $A^{<3>}$

PAGE
DATE

- Then we take $q^{<3>} \cdot k^{<1>}$ ___ $q^{<3>} \cdot k^{<5>}$ and Compute Softmax over them $\Sigma_j \exp(q \cdot k^{<j>})$
- Then finally we take the softmax values & multiply them with $V^{<1>}$ which is value for word ① & then sum it up to give $A^{<3>}$ or $A(q,k,v) = \Sigma_i \frac{\exp(q \cdot k^{<i>})}{\Sigma_j \exp(q \cdot k^{<j>})} v_i$



- The key advantage of this representation is the word of "l'Afrique isn't some fixed embedding insted The selfatt. mechanism realize that l'Afriqu is the distination of a visit which is more useful representation
- The notation used is $Attention(Q,K,V) = soft\left(\frac{QK^T}{\sqrt{d_k}}\right)$

PAGE
DATE

→ multi-head Attension

- basically just a big for-loop over the self attenstion
- each time we calculate selfattenstion for a seq it is called a head

- After the same calculations for all the words and end up with the same attention values & we do this many times so we can have many heads

- for example in the second head we change $W^q W^k W^v$ so the key product between Afrique & septembre is the highest

- we can consider each of these heads as diffacnt features that could be passed to the NN

→ transformer Network

- The first step in the transformer is these embedd. is get fed into encoder block which has a multi head attention layer, where you feed in the values Q, K, V computed from the embeddings & the weight matrices W
- second the produced matrix is passed into feedforward NN the determine what intersting features there are in the sentence
- The encoder block is repeated N times then feed the encoder into a decoder block

PAGE
DATE

- In the decoder block which its job to output the english sentence the first output <sos> token, at every step the decoder block will input the first few words whatever we have already generated of the translation

- first) the sos is feed to the multi head attension block and just the <sos> token is used to compute Q k v for the multihead attenstion block. this first multihead generate Q while the encoder generate k, v for the second multihead attenstion
- then feedforward & repeat the decoder N times
- then predict the word that is going to enter with the <sos> as input for the decoder

---

- steps makes transformer network works even better

① positional encoding of input: we encode the position of elements in input by using combination of sin & cos equations

ex. if the word embedding is vector of four values. ▯ d=4

$x^{<1>} \dots x^{<4>}$ so we create positional embedding vector of the same dimensional ▯ $p^{<1>}$

② Add & norm ~ batch norm to speed learning.