Guest Lecture CS224n

Transformers (Ashish Vaswani)

author: luis0@stanford.edu
SUNet ID: 05794739

Ashish Vaswani's presentation particularly emphasized on Transformers.

Sequential computation inhibits parallelism and cannot explicitly model long and short-range dependencies, along with other limitations. CNNs can exploit local dependencies, but further away dependencies require linear or logarithmic operations. Self-attention can help solve these problems by allowing for constant-length dependencies in a single layer, as well as making layer operations parallelizable. The Transformer architecture is built on these self-attention mechanisms, which require explicit representations of the position. Interestingly enough, the decoder requires masking of the input in order to enforce casualty. At the end of the day, Attention is extremely cheap to compute in terms of FLOPs, especially when your dimensions is much larger than the sequence length. The other issue with attention mechanisms is that they smooth out all of the input. To solve this, one can make use of multiple attention mechanisms, and we can consider each to be a sort-of feature detector. Additionally, the residual connections are important, mainly to assist in pushing through the positional information for translation. The same architecture was also applied (with some minor modifications) to images, and achieves decent results in super-resolution and image generation tasks. However, it is still nowhere near the SOTA results produced by GANs. However, it seems that it might be possible to incorporate some of these self-attention mechanisms to GAN networks, and possibly achieve even better results.

Transformer models also helped in music generation, given the self-similarity that exists therein. However, it's important to try and learn repetitive patterns. Relative distance in this context is important, and self-attention (even multi-headed attention) is not good at doing this. Therefore, an additional term needs to be added to the attention mechanism, which is skewed, to allow the model to efficiently remember relative positions.

It's interesting to see how these ideas can be combined together, including CNNs with Transformers as well as using this for GANs.