Attention is all you need

The paper 'Attention Is All You Need' introduces transformers and the sequence-to-sequence architecture. This is a neural net that transforms a sequence of elements (for example, the sequence of words of a sentence) into another sequence, having the ability to model long-range dependencies without any convolutions (which are computationally expensive), using only the self-attention mechanism.

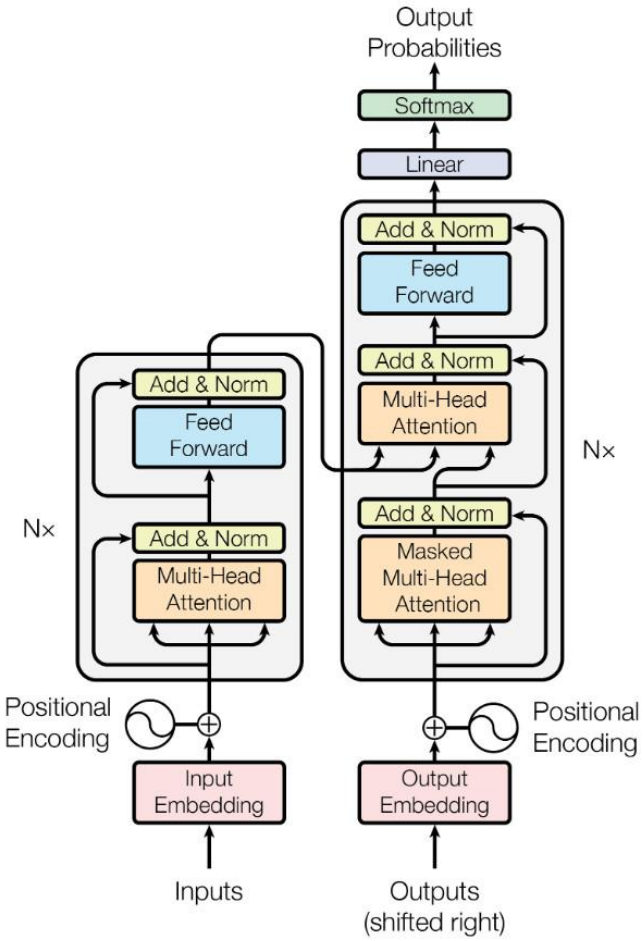

Fig. 1 The Transformer model architecture, with the encoder on the left half and the decoder on the right [1]

The attention function has as input a sequence of vectors that have three associate roles: a query and a set of key-value pairs. Intuitively, the query can be considered the current word, the key is uniquely indexing of values, and value vector is the information of the input word.

Dot products of each query and all keys (of dimension dim_k) are computed, followed by dividing

each one by $\sqrt{d_k}$ which is the dimension of each key. Afterwards, a softmax function is applied to get the weights on the values. Finally, the multiplication by value is performed, such that words not worth focusing-on have very small contribution.

The computation is done simultaneously for a series of queries; therefore, one can model queries, keys and values as learnable matrices Q, K and V:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

The Multi-Head attention represents of a series of attention layers than run in parallel, and whose outputs are concatenated.

Both the encoder and the decoder contain a fully connected feed-forward network that is separately applied to each position. It is created by applying a ReLU activation in between two linear transformations:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

where W_1 and W_2 are parameter matrices that differ from layer to layer, while linear transformations are the same everywhere.

Finally, after stacking N decoders, the softmax layer turns the score vector into probabilities (positive values that sum to 1). The cell with highest probability is selected and the word associated to this cell is the output at a particular time step.

The Transformer synthesizes features for each word in a sentence to estimate how important the other sentence words are for it, using the self-attention mechanism, which is computationally simple, efficient and parallelizable, being a series of weighted sums and activations.

Author: Gabriela Ghimpeteanu, Coronis Computing.

Bibliography

[1] Vaswani, Ashish, et al. “Attention is all you need.” Advances in neural information processing systems 30 (2017)