Overview of transformer architecture

What is “transformer”?

Transformer is a type of artificial neural network architecture that is used to solve the problem of transduction or transformation of input sequences into output sequences in deep learning applications. A transformer is a deep learning model that adopts the mechanism of self-attention, differentially weighting the significance of each part of the input data. It is used primarily in the fields of natural language processing (NLP) and computer vision (CV).

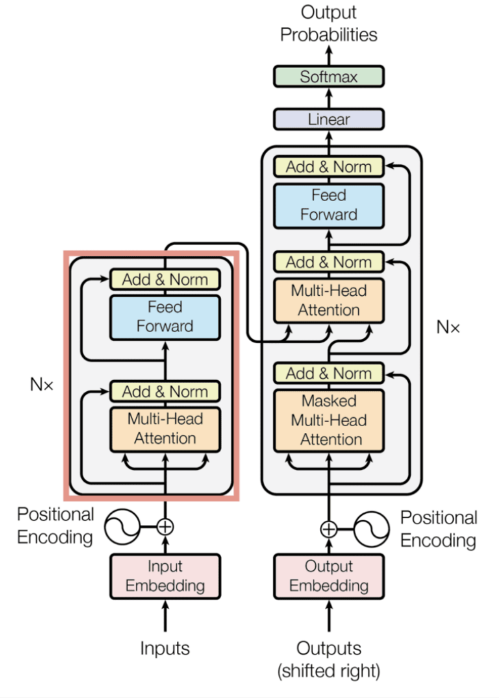
Why transformer?

Think of an application like english to chinese translation. Input and output are a sequence of words. However, it’s not enough to translate word by word, as it would likely give you a grammarly incorrect output. An approach to tackle this challenge is the use of recurrent neural networks (RNNs). Unlike feed forward neural nets, where inputs and outputs are considered to be independent of each other, the output of an RNN depends on the prior elements of a given sequence. However, there are limitations in their use: for example finding out the length of the output sequence may be challenging given the fixed-sized input/output vector architecture of RNNs. A better approach is the use of a transformer.

Transformer structure

Encoder

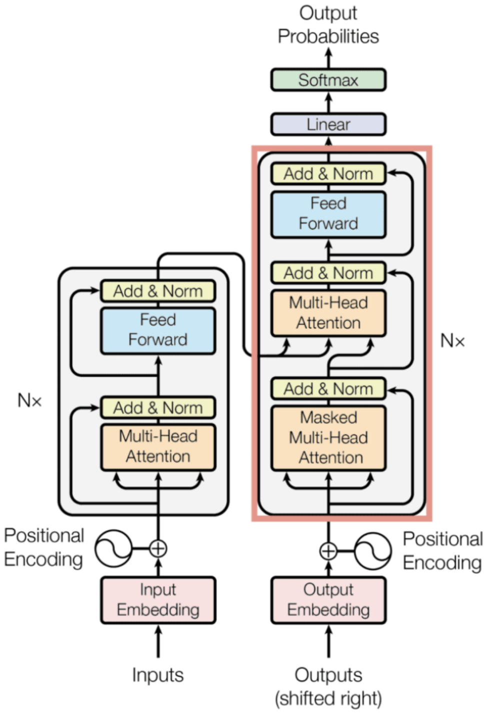
Transformer has an encoder and a decoder for its inputs and outputs. An encoder is the element that receives each word of the input sequence, encoding it into a vector carrying the context information about the whole sequence. As shown in the figure on the right (taken from “Attention is all you need”), an encoder is consist of N identical layers. Each layer is consist of two sublayers: multi-head self attention sublayer, and position-wise feed forward network (FFN) sublayer. The MHA sublayer implements the self



attention mechanism. The FFN sublayer consist of a linear layer, ReLU, and another linear layer, which process each embedding vector independently with identical weights consisting of two linear transformations with ReLU activation in between.

Decoder

A decoder is the element that understands the context (vector) and resolve the output meaningfully. The decoder block is similar to the encoder block, except it calculates the source-target attention. A decoder is consisted of 3 sublayers: masked multi-head attention layer, multi-head attention layer and a FFN layer. Masked multi-head attention means the multi-head attention receives inputs with masks so that the attention mechanism does not use information from the hidden positions. It receives the previous output of the decoder stack, augments it with positional information, and implements multi-head self-attention over it. While the encoder is designed to attend to all words in the input sequence regardless of their position in the sequence, the decoder is modified to attend only to the preceding words. The second and the third layer is similar to the one implemented in the encoder, where it implements a multi-head self-attention mechanism and passes to a fully connected feed-forward network.



Reference:

Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N.; Kaiser, Lukasz; Polosukhin, Illia (2017-06-12). "Attention Is All You Need". https://arxiv.org/abs/1706.03762

Stefania Cristina, “The Transformer Model”, https://machinelearningmastery.com/the-transformer-model/