Question-answer based on pdf

Question: How model Architecture works

Reply: The model architecture follows an encoder-decoder structure, where the encoder maps an input sequence of symbol representations to a sequence of continuous representations, and the decoder generates an output sequence of symbols one element at a time. The encoder and decoder both consist of a stack of identical layers, each of which has two sub-layers: a multi-head self-attention mechanism and a simple, position-wise fully connected feed-forward network. Residual connections are employed around each of the two sub-layers, followed by layer normalization. The self-attention mechanism allows the model to attend to different positions in the input or output sequence, while the feed-forward network provides non-linearity. The model also uses positional encoding to inject information about the relative or absolute position of the tokens in the sequence.

Question: how multi-head attention works

Reply: Instead of performing a single attention function with dmodel-dimensional keys, values and queries, we found it beneficial to linearly project the queries, keys and values htimes with different, learned linear projections to dk,dkanddvdimensions, respectively. On each of these projected versions of queries, keys and values we then perform the attention function in parallel, yielding dv-dimensional 4output values. These are concatenated and once again projected, resulting in the final values, as depicted in Figure 2. Multi-head attention allows the model to jointly attend to information from different representation subspaces at different positions. With a single attention head, averaging inhibits this. MultiHead(Q, K, V) = Concat(head 1, ...,head h)WO where head i= Attention(QWQ i, KWK i, V WV i) Where the projections are parameter matrices WQ i∈Rdmodel×dk,WK i∈Rdmodel×dk,WV i∈Rdmodel×dv andWO∈Rhdv×dmodel. In this work we employ h= 8 parallel attention layers, or heads.