**Transformer Architecture**

1. **Introduction**

RNN based encoder-decoder models struggle when processing large sequences. The encoder model is responsible for encoding this large sequence into a vector that represents the rich representation of the input sequence. Then, the decoder processes this vector to generate the output. However, the performance of this model decreases as the input sequence length increases. For instance, in a language translational task, the model struggles to translate a long paragraph. If a human were to translate this long sentence, they wouldn’t read the entire paragraph, memorize it, and then try to translate it. Instead, they would translate one sentence at a time. An attention network works similarly to a human, processing one part at a time. Transformer model utilizes these attention mechanisms to handle large sequences effectively.

Transformer network has revolutionized the field of Natural Language Processing (NLP). Many of the most effective algorithms for NLP today are based on the transformer architecture. Unlike RNN, which process the sequences sequentially, transformer allows for parallel processing of entire sequences. This allows the transformer to handle long range dependencies effectively. The key innovation in transformer architecture is attention mechanism combined with a Convolutional Neural Network (CNN) style of processing. This attention mechanism enables the model to focus on the most relevant parts of the input for each output. This allows the model to compute rich, useful representations of words in parallel, rather than sequentially. The transformer architecture is divided into two main sections: the Encoder and the Decoder. Both these sections include key components such as multi-head self-attention mechanism and a Feed Forward Neural Network.

1. **Self-Attention Mechanism**

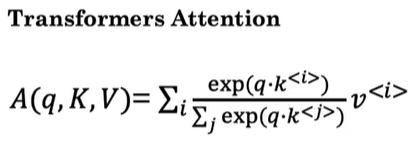
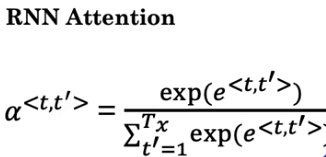
To understand the self-attention mechanism, let's consider an example from a language translation task. Take the French sentence “**Jane visite l’Afrique en septembre.**” This sentence needs to be translated into English. The main goal of the self-attention mechanism is to compute an attention-based vector representation for each word in the input sentence.



In our example, the sentence contains 5 words. Therefore, the self-attention mechanism will generate 5 vectors (A1 to A5), each corresponding to one word in the input sequence.

Let us explore how this mechanism computes A3 for the word l’Afrique.

* One way to represent the word "l’Afrique" is through word embeddings. However, this representation can vary depending on the context.
* Are we referring to "l’Afrique/Africa" as a site of historical interest, a holiday destination, or as the world's second-largest continent?
* Depending on the context, we need to represent "l’Afrique" differently.
* The A3 vector accomplishes this by examining the surrounding words to determine the context in which Africa is being discussed in the sentence and then finding the most appropriate representation for the word.



* Q represents query, K represents key, V represents Value. These vectors are the key inputs for computing the attention value for each word.
* Associate each of the words with three values called query, key, and value pairs. They are calculated as follows:
  + q<3> = WQ.x<3>
  + K<3> = WK.x<3>
  + V<3> = WV.x<3>

where WQ