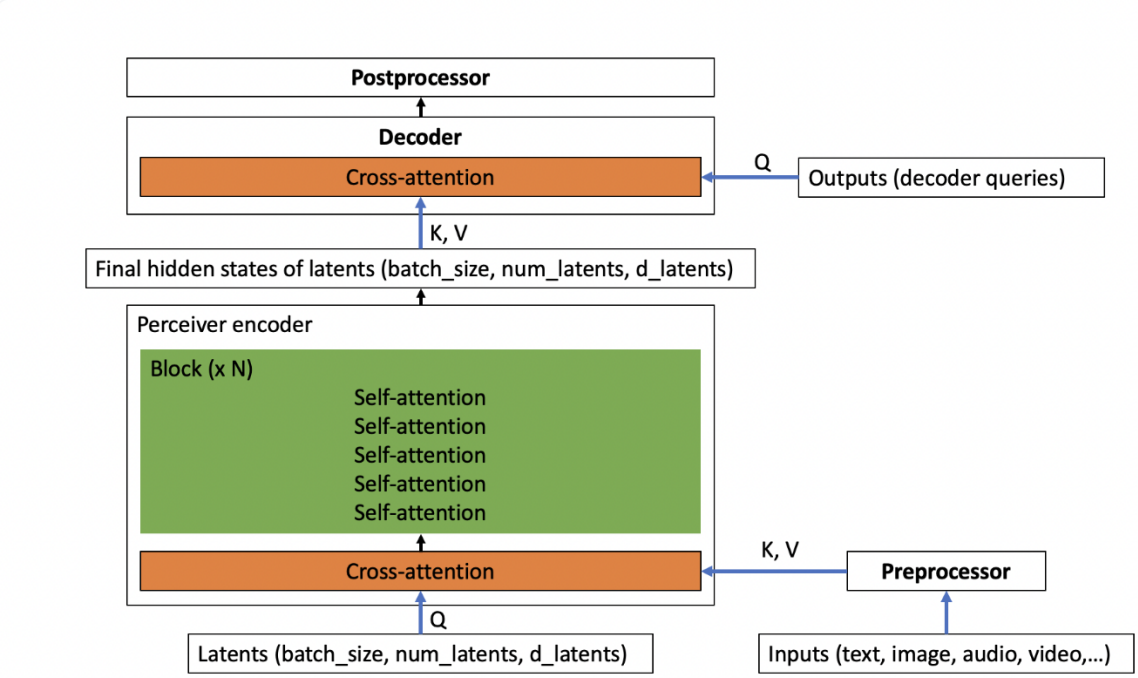
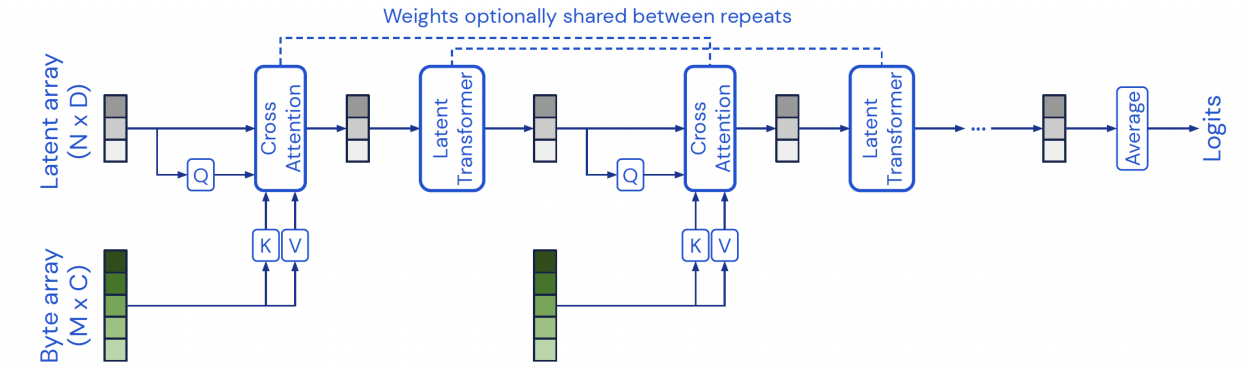
Perceiver: General Perception with Iterative Attention

Status	Finished
Author	Andrew Jaegle et al.
Publishing/Release Date	@June 23, 2021
Publisher	ICML 2021
Link	https://arxiv.org/abs/2103.03206
Summary	This paper introduces a new Transformer based NLP method, which can be used on any modalities including images, text, audio, and video.
Score /5	⭐⭐⭐⭐⭐
Column	
Type	Academic Journal
Original Git Repo	https://github.com/deepmind/deepmind-research/tree/master/perceiver
Git Repo	https://github.com/NielsRogge/Transformers-Tutorials/tree/master/Perceiver
Tutorial	https://huggingface.co/blog/perceiver



Perceiver Architecture

Note that the output of a QKV attention layer always has the same shape as the shape of the queries. To reduce the size, Input latents produce a Q which is smaller than KV from inputs, hence, we resolve the problem with Transformer. Then, in the last states, outputs decoder queries control the shape of the outputs.



The Perceiver is an architecture based on attentional principles that scales to high-dimensional inputs such as images, videos, audio, point-clouds, and multimodal combinations without making domain-specific assumptions. The Perceiver uses a cross-

attention module to project an high-dimensional input byte array to a fixed-dimensional latent bottleneck (the number of input indices M is much larger than the number of latent indices N) before processing it using a deep stack of Transformer-style self-attention blocks in the latent space. The Perceiver iteratively attends to the input byte array by alternating cross-attention and latent self-attention blocks.

Reason to use perceiver:

- The **Perceiver** aims to solve this limitation by employing the self-attention mechanism on a set of latent variables, rather than on the inputs. The `inputs` (which could be text, image, audio, video) are only used for doing cross-attention with the latents.
- Perceiver IO can do BERT-style masked language modeling directly using *bytes* instead of tokenized inputs.